



# **Apache Traffic Control Documentation**

*Release 7.0.0*

**Apache Traffic Control**

**Aug 09, 2022**



# CONTENTS

<b>1</b>	<b>CDN Basics</b>	<b>3</b>
1.1	CDN Basics . . . . .	3
<b>2</b>	<b>Traffic Control Overview</b>	<b>7</b>
2.1	Traffic Control Overview . . . . .	7
<b>3</b>	<b>Administrator's Guide</b>	<b>93</b>
3.1	Administrator's Guide . . . . .	93
<b>4</b>	<b>Developer's Guide</b>	<b>271</b>
4.1	Developer's Guide . . . . .	271
<b>5</b>	<b>APIs</b>	<b>391</b>
5.1	Traffic Ops API . . . . .	391
<b>6</b>	<b>Tools</b>	<b>1575</b>
6.1	Tools . . . . .	1575
<b>7</b>	<b>FAQ</b>	<b>1611</b>
7.1	FAQ . . . . .	1611
<b>8</b>	<b>Indices and Tables</b>	<b>1613</b>
8.1	Glossary . . . . .	1613
	<b>Python Module Index</b>	<b>1627</b>
	<b>Index</b>	<b>1629</b>





The vast majority of today's Internet traffic is media files being sent from a single source to many thousands or even millions of destinations. Content Delivery Networks make that one-to-many distribution possible in an economical way.

Traffic Control is an Open Source implementation of a Content Delivery Network.

The following documentation sections are available:



## CDN BASICS

A review of the basic functionality of a Content Delivery Network.

### 1.1 CDN Basics

Traffic Control is a CDN control plane, see the topics below to familiarize yourself with the basic concepts of a CDN.

#### 1.1.1 Content Delivery Networks

The vast majority of today's Internet traffic is media files (often video or audio) being sent from a single source (the *Content Provider*) to many thousands or even millions of destinations (the *Content Consumers*). CDN (Content Delivery Network)s are the technology that make that one-to-many distribution efficient. A CDN is a distributed system of servers for delivering content over HTTP(S). These servers are deployed in multiple locations with the goal of optimizing the delivery of content to the end users, while minimizing the traffic on the network. A CDN typically consists of the following:

**cache servers** The *cache server* is a server that both proxies the requests and caches the results for reuse. Traffic Control uses [Apache Traffic Server](#) to provide *cache servers*.

**Content Router** A *content router* ensures that the end user is connected to the optimal *cache server* for the location of the end user and content availability. Traffic Control uses [Traffic Router](#) as a *content router*.

**Health Protocol** The [Health Protocol](#) monitors the usage of the *cache servers* and tenants in the CDN.

**Configuration Management System** In many cases a CDN encompasses hundreds or even thousands of servers across a large geographic area. In such cases, manual configuration of servers becomes impractical, and so a central authority on configuration is used to automate the tasks as much as possible. [Traffic Ops](#) is the Traffic Control configuration management system, which is interacted with via [Traffic Portal](#).

**Log File Analysis System** Statistics and analysis are extremely important to the management and administration of a CDN. Transaction logs and usage statistics for a Traffic Control CDN are gathered into [Traffic Stats](#).

## 1.1.2 HTTP/1.1

For a comprehensive look at Traffic Control, it is important to understand basic HTTP/1.1 protocol operations and how *cache servers* function.

### See also:

For complete details on HTTP/1.1 see [RFC 2616](#).

What follows is a sequence of events that take place when a client requests content from an HTTP/1.1-compliant server.

1. The client sends a request to the LDNS (Local DNS) server to resolve the name `www.origin.com` to an IP address, then sends an HTTP request to that IP.

---

**Note:** A DNS response is accompanied by a TTL (Time To Live) which indicates for how long a name resolution should be considered valid. While longer DNS TTLs of a day (86400 seconds) or more are quite common in other use cases, in CDN use-cases DNS TTLs are often below a minute.

---

1: A Client Request for `/foo/bar/fun.html` from `www.origin.com`

```
GET /foo/bar/fun.html HTTP/1.1
Host: www.origin.com
```

2. The server at `www.origin.com` looks up the content of the path `/foo/bar/fun.html` and sends it in a response to the client.

2: Server Response

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Content-Length: 45

<!DOCTYPE html><html><body>This is a fun file</body></html>
```

## 1.1.3 Cache Control Headers and Revalidation

The HTTP/1.1 specification in [RFC 2616#section-14.9](#) allows for origin servers and clients to influence how caches treat their requests and responses. By default, the Traffic Control CDN will honor cache control headers. Most commonly, origin servers will tell the downstream caches how long a response can be cached.

---

**Note:** The terms “content revalidation” and “content invalidation” are often convoluted when referring to the same behavior. Within the context of Traffic Control, the two should be consid-

ered synonymous.

### 3: This Response may Only be Cached for 86400 Seconds

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Server: Apache/2.2.15 (Red Hat)
Last-Modified: Sun, 14 Dec 2014 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
Cache-Control: max-age=86400
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE html><html><body>This is a fun file</body></html>
```

The `max-age` directive in the `Cache-Control` header tells downstream caching systems that the maximum time for which they are allowed to cache this response is the specified number of seconds. The origin can also add an `Expires:` header, explicitly telling the cache the time this response is to be expired. When a response is expired it usually doesn't get deleted from the cache, but, when a request comes in that would have hit on this response if it was not expired, the cache *revalidates* the response. In particular, this is the way ATS (Apache Traffic Server) handles content revalidation. Instead of requesting the object again from the origin server, the cache will send a request to the origin indicating what version of the response it has, and asking if it has changed. If it changed, the server will send a `200 OK` response, with the new data. If it has not changed, the origin server will send back a `304 Not Modified` response indicating the response is still valid, and that the cache can reset the timer on the response expiration. To indicate what version the client (cache) has it will add an `If-Not-Modified-Since:` header, or an `If-None-Match:` header. For example, in the `If-None-Match:` case, the origin will have sent an `ETag` header that uniquely identifies the response. The client can then later use that in a revalidation request to check if the `ETag` of the requested content has changed.

### 4: The Cache Server Sends a Request with the Old ETag Value in the If-None-Match Header

```
GET /foo/bar/fun.html HTTP/1.1
If-None-Match: "1aa008f-2d-50a3559482cc0"
Host: www.origin.com
```

If the content has changed (meaning, the new response would not have had the same `ETag`) the server **MUST** respond with the up-to-date content, usually in the body of a `200 OK` response.

### 5: The Origin Responds with the Modified Content and a New ETag

```
HTTP/1.1 200 OK
Date: Sun, 18 Dec 2014 3:22:44 GMT
Server: Apache/2.2.15 (Red Hat)
```

(continues on next page)

(continued from previous page)

```
Last-Modified: Sun, 14 Dec 2014 23:18:51 GMT
ETag: "1aa008f-2d-50aa00feadd"
Cache-Control: max-age=604800
Content-Length: 49
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE html><html><body>This is NOT a fun file</body></html>
```

If the content did not change (meaning, the response would have had the same ETag) the server SHOULD respond with a 304 Not Modified. In most cases, the server will also send back an ETag header, since the client is allowed to send multiple ETag values in its If-None-Match header to check against multiple cached versions of the content, and the ETag will tell it which specifically is the current plugin version. This is a very rare use case, and ATS will not make use of this feature without a plugin to modify its behavior.

6: The Content has not been Modified so the Server Indicates  
the Cached Version is Up-To-Date

```
HTTP/1.1 304 Not Modified
Date: Sun, 18 Dec 2014 3:22:44 GMT
Server: Apache/2.2.15 (Red Hat)
Last-Modified: Sun, 14 Dec 2014 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
Cache-Control: max-age=604800
Connection: close
```

## TRAFFIC CONTROL OVERVIEW

An introduction to the Traffic Control architecture, components, and their integration.

### 2.1 Traffic Control Overview

Introduces the Traffic Control architecture, components, and their integration.

#### 2.1.1 Cache Groups

A *Cache Group* is - ostensibly - exactly what it sounds like it is: a group of *cache servers*. More specifically, every server in a Traffic Control CDN must be in a Cache Group (even if they are not actually *cache servers*). Typically a Cache Group is representative of the available *cache servers* within a specific geographical location. Despite that *cache servers* have their own *Physical Locations*, when *cache servers* are chosen to serve content to a client based on geographic location the geographic location actually used for comparisons is that for the Cache Group that contains it, not the geographic location of the *cache server* itself.

The most typical *Types* of Cache Groups are *EDGE\_LOC* which contain *Edge-tier cache servers* and *MID\_LOC* which contain *Mid-tier cache servers*. The latter are each designated as a *Parent* of one or more of the former to fill out the two-tiered caching hierarchy of an ATC (Apache Traffic Control) CDN.

Consider the example CDN in Fig. 1. Here some country/province/region has been divided into quarters: Northeast, Southeast, Northwest, and Southwest. The arrows in the diagram indicate the flow of requests. If a client in the Northwest, for example, were to make a request to the *Delivery Service*, it would first be directed to some *cache server* in the “Northwest” Edge-tier *Cache Group*. Should the requested content not be in cache, the Edge-tier server will select a parent from the “West” *Cache Group* and pass the request up, caching the result for future use. All Mid-tier *Cache Groups* (usually) answer to a single *Origin* that provides canonical content. If requested content is not in the Mid-tier cache, then the request will be passed up to the *Origin* and the result cached.

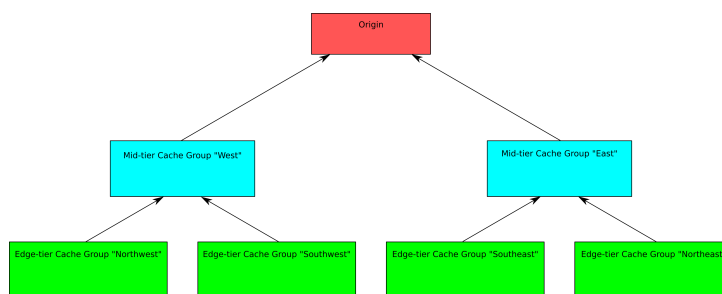


Fig. 1: An example CDN that shows the hierarchy between four Edge-tier *Cache Groups*, two Mid-tier *Cache Groups*, and one Origin

## Regions, Divisions, and Locations

In addition to being in a Cache Group, all servers have to have a *Physical Location*, which defines their geographic latitude and longitude. Each *Physical Location* is part of a *Region*, and each *Region* is part of a *Division*. For example, Denver could be the name of a *Physical Location* in the Mile High *Region* and that *Region* could be part of the West *Division*. The hierarchy between these terms is illustrated graphically in *Topography Hierarchy*.

To create these structures in Traffic Portal, first make at least one *Division* under *Topology* → *Divisions*. Next enter the desired *Region*(s) in *Topology* → *Regions*, referencing the earlier-entered *Division*(s). Finally, enter the desired *Physical Location*(s) in *Topology* → *Phys Locations*, referencing the earlier-entered *Region*(s).

A Cache Group is a logical grouping of cache servers, that don’t have to be in the same *Physical Location* (in fact, usually a Cache Group is spread across minimally two *Physical Locations* for redundancy purposes), but share geographical coordinates for content routing purposes. There is no strict requirement that *cache servers* in a Cache Group share a *Physical Location*, *Region*, or *Division*. This may be confusing at first as there are a few places in code, interfaces, or even documentation where Cache Groups are referred to as “Cache Locations” or even erroneously as “Physical Locations”.

## Properties

Cache Groups are modeled several times over, in the Traffic Ops database, in Traffic Portal forms and tables, and several times for various *Traffic Ops API* versions in the new Go Traffic Ops codebase. Go-specific data structures can be found in the project’s [GoDoc documentation](#). Rather than application-specific definitions, what follows is an attempt at consolidating all of the different properties and names of properties of Cache Group objects throughout the ATC suite. The names of these fields are typically chosen as the most human-readable and/or most commonly-used names for the fields, and when reading please note that in many cases these names will appear camelCased or snake\_cased to be machine-readable. Any aliases of these fields that are not merely case transformations of the indicated, canonical names will be noted in a table of aliases.

### See also:

The API reference for Cache Group-related endpoints such as *cachegroups* contains definitions of the Cache Group object(s) returned and/or accepted by those endpoints.



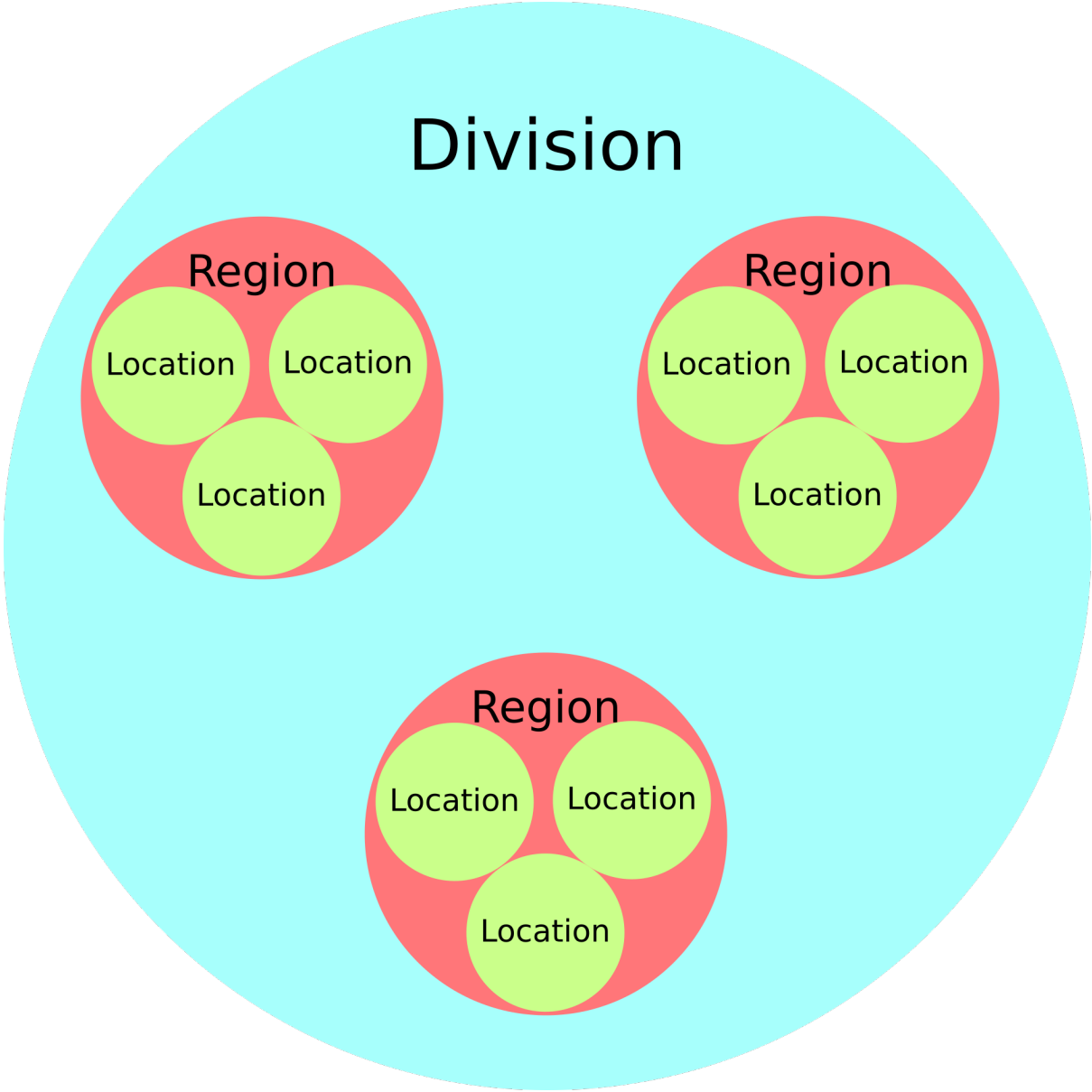


Fig. 2: Topography Hierarchy

### ASNs

A Cache group can have zero or more ASNs (Autonomous System Numbers) assigned to it, which is used to classify traffic that passes through a CDN. These are typically not represented on a Cache Group object itself, but rather as a separate object indicating the relationship, e.g. in the requests and responses of the *asns* endpoint.

#### See also:

The [Autonomous System Wikipedia](#) page for an explanation of what an ASN (Autonomous System Number) actually is.

### Coordinate

---

**Tip:** Normally, one need not interact with this. In most contexts, this property of a Cache Group is not even exposed, but instead the Cache Group's *Latitude* and *Longitude* are exposed and should be directly manipulated.

---

The *Coordinate* of a Cache Group defines the geographic coordinates of a Cache Group that is used for routing clients based on geographic location. It is also used to determine the “closest” Cache Group to another for the purposes of *Fallback to Closest*.

Typically, this is expressed as an integral, unique identifier for the “Coordinate” object bound to the Cache Group that defines its geographic location, but occasionally it may appear as the name of that “Coordinate” object.

---

**Note:** When a new Cache Group is created, it is not necessary to first create a “Coordinate” object where it may reside. Instead, “Coordinates” are created automatically to reflect the *Latitude* and *Longitude* given to the newly created Cache Group. The name of the generated “Coordinate” will conform to the pattern `from_cachegroup_Name` where Name is the Cache Group's *Name*. Because of this, creating new Cache Groups will fail if a “Coordinate” with a name matching that pattern already exists.

---

### Fallbacks

*Fallbacks* are a group of zero or more Cache Groups to be considered for routing when a Cache Group becomes unavailable due to high load or excessive maintenance. These are normally represented by an array of each Cache Group's *ID*, but may occasionally appear as the *Name* or *Short Name* of each Cache Group.

This set is consulted before *Fallback to Closest* is taken into consideration.

#### See also:

*Health Protocol*

Table 1: Aliases

Name	Use(s)	Type(s)
Failover Cache Groups	Traffic Portal forms - but <b>not</b> tables	List or array of <i>Names</i> as strings
backupLoca- tions	In CDN <i>Snapshots</i>	A sub-object called “list” which is a list or array of Cache Group <i>Names</i> as strings
Backup- CacheGroups	Traffic Router source code	A List of strings that are the <i>Names</i> of Cache Groups

## Fallback to Closest

This is a boolean field which, when `true` (`True`, `TRUE` etc.) causes routing to “fall back” on the nearest Cache Group - geographically - when this Cache Group becomes unavailable due to high load and/or excessive maintenance.

When this *is* a “true” value, the closest Cache Group will be chosen if and only if any set of *Fallbacks* configured on the Cache Group has already been exhausted and no available Cache Groups were found..

**See also:**

*Health Protocol*

Table 2: Aliases

Name	Use(s)	Type(s)
Fallback to Geo Failover	Traffic Portal forms	Unchanged ( <code>bool</code> , <code>Boolean</code> etc.)

## ID

All Cache Groups have an integral, unique identifier that is mainly used to reference it in the *Traffic Ops API*.

Despite that a Cache Group’s *Name* must be unique, this is the identifier most commonly used to represent a unique Cache Group in most contexts throughout ATC. One notable exception is in CDN *Snapshots* and in routing configuration used by Traffic Router.

## Latitude

The Cache Group’s geomagnetic latitude for use in routing and for the purposes of *Fallback to Closest*.

Table 3: Aliases

Name	Use(s)	Type(s)
Geo Magnetic Latitude	Traffic Portal forms	Unchanged ( <code>number</code> , <code>float</code> etc.)

## Localization Methods

The *Localization Methods* of a Cache Group define the methods by which Traffic Router is allowed to route clients to *cache servers* within this Cache Group. This is a collection of the allowed methods, and the values in the collection are restricted to the following.

- “Coverage Zone File” (alias CZ in source code, database entries, and *Traffic Ops API* requests/responses) allows Traffic Router to direct clients to this Cache Group if they were assigned a geographic location by looking up their IP address in the *Coverage Zone File*.
- “Deep Coverage Zone File” (alias DEEP\_CZ in source code, database entries, and *Traffic Ops API* requests/responses) was intended to allow Traffic Router to direct clients to this Cache Group if they were assigned a geographic location by looking up their IP addresses in the *Deep Coverage Zone File*. However, it **has no effect at all**. This option therefore will not appear in Traffic Portal forms.

**Warning:** In order to make use of “deep caching” for a *Delivery Service*, all that is required is that *Delivery Service* has *Deep Caching* enabled. If that is done and a *cache server* appears in the *Deep Coverage Zone File* then clients can and will be routed using that method. There is no way to disable this behavior on a Cache Group (or otherwise) basis, and the precence or absence of DEEP\_CZ in a Cache Group’s Localization Methods has no meaning.

- “Geo-IP Database” (alias GEO in source code, database entries, and *Traffic Ops API* requests/responses) allows Traffic Router direct clients to this Cache Group if the client’s IP was looked up in a provided IP address-to-geographic location mapping database to provide their geographic location.

If none of these localization methods are in the set of allowed methods on a Cache Group, it is assumed that *clients should be allowed to be routed to that Cache Group regardless of the method used to determine their geographic location*.

This property only has meaning for Cache Groups containing *Edge-tier cache servers*. Which is to say (one would hope) that it only has meaning for *EDGE\_LOC* Cache Groups.

Table 4: Aliases

Name	Use(s)	Type(s)
Enabled Localization Methods	Traffic Portal forms, Traffic Router source code	Unchanged (Set<String>, Array<string> etc.)

## Longitude

The Cache Group’s geomagnetic longitude for use in routing and for the purposes of *Fallback to Closest*.

Table 5: Aliases

Name	Use(s)	Type(s)
Geo Magnetic Longitude	Traffic Portal forms	Unchanged (number, float etc.)

## Name

A unique, human-friendly name for the Cache Group, with no special character restrictions or length limit.

Though this property must be unique for all Cache Groups, *ID* is more commonly used as a unique identifier for a Cache Group in most contexts.

Table 6: Aliases

Name	Use(s)	Type(s)
locationID	CDN <i>Snapshots</i>	Unchanged (str, String etc.)

## Parent

A Cache Group can have a *parent* Cache Group which has different meanings based on the Cache Group’s *Type*.

- An *EDGE\_LOC* Cache Group’s parent must be a *MID\_LOC* Cache Group. When configuration files are generated for *Edge-tier cache servers*, their *parents* (different from a Cache Group parent) will be selected from the *Mid-tier cache servers* contained within the Cache Group that is the parent of their containing Cache Group.
- A *MID\_LOC* Cache Group’s parent must be an *ORG\_LOC* Cache Group. However, if any given *MID\_LOC* *either doesn’t have a parent, or does and it’s an ORG\_LOC*, then **all *ORG\_LOC* Cache Groups - even across CDNs - will be considered the parents of that *MID\_LOC* Cache Group.** This parent relationship only has any meaning in the context of “multi-site-origins”, as they are unnecessary in other scenarios.

**See also:**

*Configure Multi-Site Origin*

- For all other Cache Group *Types*, parent relationships have no meaningful semantics.

**Danger:** There is no safeguard in the data model or *Traffic Ops API* that ensures these relationships hold. If they are violated, the resulting CDN behavior is undefined - and almost certainly undesirable.

*Parents* are typically represented by their *ID*, but may occasionally appear as their *Name*.

### See also:

The [Apache Traffic Server documentation](#) for the “parent.config” configuration file. ATC parentage relationships boil down to the `parent` field of that configuration file, which sets the *parents* of *cache servers*.

Table 7: Aliases

Name	Use(s)	Type(s)
Parent Cache Group	Traffic Portal forms and tables	Unchanged (str, String etc.)
Parent-CacheGroupID	Traffic Ops database, Traffic Ops Go code, <i>Traffic Ops API</i> requests and responses	Positive integer (int, bigint etc.)
Parent-CacheGroup-Name	Traffic Ops database, Traffic Ops Go code, <i>Traffic Ops API</i> requests and responses	Unchanged (str, String etc.)

## Parameters

For unknown reasons, it’s possible to assign *Parameters* to Cache Groups. This has no attached semantics respected by ATC, and exists today only for compatibility purposes.

These are nearly always represented as a collection of *Parameter IDs* but occasionally they can be expressed as full *Parameter* objects.

Table 8: Aliases

Name	Use(s)	Type(s)
cachegroup-Parameters	Certain <i>Traffic Ops API</i> responses, sometimes in internal Go code	Usually a tuple of associative information like a Cache Group ID and a Parameter ID

## Secondary Parent

A *secondary parent* of a Cache Group is used for fall-back purposes on a *cache server*-to-*cache server* basis after routing has already occurred (in contrast with *Fallbacks* and *Fallback to Closest* which operate at the routing step on a Cache Group-to-Cache Group basis).

For an explanation of what it means for one Cache Group to be the *Parent* of another, refer to the *Parent* section.

### See also:

The [Apache Traffic Server documentation](#) for the “parent.config” configuration file. ATC secondary parentage relationships boil down to the `secondary_parent` field of that configuration file, which sets the “secondary parents” of *cache servers*.

Table 9: Aliases

Name	Use(s)	Type(s)
Secondary Parent Cache Group	Traffic Portal forms and tables	Unchanged (str, String etc.)
SecondaryParent- CacheGroupID	Traffic Ops database, Traffic Ops Go code, <i>Traffic Ops API</i> requests and responses	Positive integer (int, bigint etc.)
SecondaryParent- CacheGroupName	Traffic Ops database, Traffic Ops Go code, <i>Traffic Ops API</i> requests and responses	Unchanged (str, String etc.)

## Servers

The primary purpose of a Cache Group is to contain servers. In most scenarios it is implied or assumed that the servers are *cache servers*, but this is not required and in fact it is not by any means uncommon for the contained servers to be of some other arbitrary *Type*.

A Cache Group can have zero or more assigned servers, and each server can belong to at most one Cache Group.

## Short Name

This is typically an abbreviation of the Cache Group's *Name*. The main difference is that it isn't required to be unique.

## Type

A Cache Group's *Type* determines what kind of servers it contains. Note that there's no real restriction on the kinds of servers that a Cache Group can contain, and this *Type* serves as more of a guide in certain contexts. The *Types* available by default are described in this section.

**Tip:** Because *Types* are mutable, the actual *Types* that can describe Cache Groups cannot be completely or precisely defined. However, there are no good reasons of which this author can think to modify or delete the default *Types* herein described, and honestly the good reasons to even merely add to their ranks are likely few.

Table 10: Aliases

Name	Use(s)	Type(s)
Type ID	Traffic Ops client and server Go code, <i>Traffic Ops API</i> requests and responses	positive integer (int, bigint, etc.)
Type Name	<i>Traffic Ops API</i> requests and responses, Traffic Ops database	unchanged (string, String etc.)

### EDGE\_LOC

This *Type* of Cache Group contains *Edge-tier cache servers*.

### MID\_LOC

This *Type* of Cache Group contains *Mid-tier cache servers*.

### ORG\_LOC

This *Type* of Cache Group contains *Origins*. The primary purpose of these is to group *Origins* for the purposes of “multi-site-origins”, and it’s suggested that if that doesn’t meet your use-case that these be mostly avoided. In general, it’s not strictly necessary to create *Origin servers* in ATC at all, unless you have to support “multi-site-origins”.

**See also:**

*Configure Multi-Site Origin*

### TC\_LOC

A catch-all *Type* of Cache Group that’s meant to house infrastructure servers that gain no special semantics based on the Cache Group containing them, e.g. Traffic Portal instances.

### TR\_LOC

This *Type* of Cache Group is meant specifically to contain Traffic Router instances.

## 2.1.2 Delivery Service Requests

A DSR (Delivery Service Request) is a request to create a new *Delivery Service*, delete an existing *Delivery Service*, or modify an existing *Delivery Service*. The model for a DSR is, therefore, somewhat nebulous as it depends on the model of a *Delivery Service*. This concept is not to be confused with *deliveryservices/request*.

**See also:**

*Delivery Service Requests* for information about how to use DSRs in

**See also:**

The API reference for Delivery Service-related endpoints such as *deliveryservice\_requests* contains definitions of the Delivery Service object(s) returned and/or accepted by those endpoints.



## See also:

The [github.com/apache/trafficcontrol/lib/go-tc.DeliveryServiceRequestV30](https://github.com/apache/trafficcontrol/lib/go-tc.DeliveryServiceRequestV30) Go structure documentation.

The model for a DSR in the most recent version of the *Traffic Ops API* is given in *Delivery Service Request as a Typescript Interface* as a Typescript interface.

### 7: Delivery Service Request as a Typescript Interface

```
interface DeliveryServiceRequest {
  assignee: string | null;
  author: string;
  changeType: 'create' | 'delete' | 'update';
  createdAt: Date; // RFC3339 string - response-only field
  id?: number; // response-only field
  lastEditedBy: string; // response-only field
  lastUpdated: Date; // RFC3339 string - response-only field
  original?: DeliveryService;
  requested?: DeliveryService;
  status: 'draft' | 'pending' | 'submitted' | 'rejected' | 'complete
  →';
}
// Specifically, every DSR will be one of the following more "concrete
  →" types.
interface CreatedSR extends DeliveryServiceRequest {
  changeType: 'create';
  original: undefined;
  requested: DeliveryService;
}
interface DeletedSR extends DeliveryServiceRequest {
  changeType: 'delete';
  original: DeliveryService;
  requested: undefined;
}
interface UpdatedSR extends DeliveryServiceRequest {
  changeType: 'update';
  original: DeliveryService;
  requested: DeliveryService;
}
```

### Assignee

Assignee is the username of the user to whom the DSR is assigned. It may be null-typed if there is no assignee for a given DSR.

Table 11: Aliases/Synonyms

Name	Use(s)	Type
assigneeId	older API versions, internally in Traffic Control code	unsigned integer

### Author

Author is the username of the user who created the DSR.

Table 12: Aliases/Synonyms

Name	Use(s)	Type
authorId	older API versions, internally in Traffic Control code	unsigned integer

### Change Type

This string indicates the action that will be taken in the event that the DSR is fulfilled. It can be one of the following values:

**create** A new *Delivery Service* will be created

**delete** An existing *Delivery Service* will be deleted

**update** An existing *Delivery Service* will be modified

### Created At

This is the date and time at which the DSR was created. In the context of the *Traffic Ops API*, it is formatted as an **RFC 3339** date string except where otherwise noted.

### ID

An integral, unique identifier for the DSR.

## Last Edited By

This is the username of the user by whom the DSR was last edited. Author is the username of the user who created the DSR.

Table 13: Aliases/Synonyms

Name	Use(s)	Type
lastEditedById	older API versions, internally in Traffic Control code	unsigned integer

## Original

If this property of a DSR exists, then it represents the original *Delivery Service* before the DSR was/would have been/is fulfilled. This property only exists on DSRs that have a *Change Type* of “update” or “delete”. This is a full representation of a *Delivery Service*, and so in the context of *Traffic Ops API* has the same structure as a request to or response from the *deliveryservices* endpoint, as appropriate for its *Change Type*.

Table 14: Aliases/Synonyms

Name	Use(s)	Type
deliv- eryser- vice	older API versions combined the concepts of Original and <i>Requested</i> into this single field	unchanged ( <i>Delivery Service</i> representation)

## Requested

If this property of a DSR exists, then it represents the *Delivery Service* the creator wishes to exist - possibly in place of an existing *Delivery Service* that shares its identifying properties. This property only exists on DSRs that have a *Change Type* of “update” or “create”. This is a full representation of a *Delivery Service*, and so in the context of *Traffic Ops API* has the same structure as a request to or response from the *deliveryservices* endpoint, as appropriate for its *Change Type*.

Table 15: Aliases/Synonyms

Name	Use(s)	Type
deliv- eryser- vice	older API versions combined the concepts of <i>Original</i> and Requested into this single field	unchanged ( <i>Delivery Service</i> representation)

### Status

Status is a string that indicates the point in the DSR workflow lifecycle at which a given DSR is. Generally a DSR may be either “open” - meaning that it is available to be modified, reviewed, and possibly either completed or rejected - or “closed” - meaning that it has been completed or rejected. More specifically, “open” DSRs have one of the following Statuses:

**draft** The DSR is not yet ready for completion or review that might result in rejection, as it is still being actively worked on.

**submitted** The DSR has been submitted for review, but has not yet been reviewed.

... while a “closed” DSR has one of these Statuses:

**complete** The DSR was approved and its declared action was taken.

**pending** The DSR was approved and the changes are applied, but the new configuration is not yet disseminated to other ATC components - usually meaning that it cannot be considered truly complete until a *Snapshot* is taken or a *Queue Updates* performed.

**rejected** The DSR was rejected and closed; it cannot be completed.

A “closed” DSR cannot be edited - except to change a “pending” Status to “complete” or “rejected”.

### 2.1.3 Delivery Services

“Delivery Services” are a very important construct in ATC. At their most basic, they are a source of content and a set of *cache servers* and configuration options used to distribute that content.

Delivery Services are modeled several times over, in the Traffic Ops database, in Traffic Portal forms and tables, and several times for various *Traffic Ops API* versions in the new Go Traffic Ops codebase. Go-specific data structures can be found in [the project’s GoDoc documentation](#). Rather than application-specific definitions, what follows is an attempt at consolidating all of the different properties and names of properties of Delivery Service objects throughout the ATC suite. The names of these fields are typically chosen as the most human-readable and/or most commonly-used names for the fields, and when reading please note that in many cases these names will appear camelCased or snake\_cased to be machine-readable. Any aliases of these fields that are not merely case transformations of the indicated, canonical names will be noted in a table of aliases.

#### See also:

The API reference for Delivery Service-related endpoints such as *deliverservices* contains definitions of the Delivery Service object(s) returned and/or accepted by those endpoints.

#### See also:

*Delivery Service Requests*

## Active

Whether or not this Delivery Service is active on the CDN and can be served. When a Delivery Service is not “active”, Traffic Router will not be made aware of its existence - i.e. it will not appear in CDN *Snapshots*. Setting a Delivery Service to be “active” (or “inactive”) will require that a new *Snapshot* be taken.

## Anonymous Blocking

Enables/Disables blocking of anonymized IP address - proxies, TOR (The Onion Ring) exit nodes, etc - for this Delivery Service. Set to true to enable blocking of anonymous IPs for this Delivery Service.

Table 16: Aliases

Name	Use(s)	Type(s)
anony-mous-Blockin-gEnabled	Traffic Ops client and server Go code, <i>Traffic Ops API</i> requests and responses	usually unchanged (boolean), but sometimes as a string containing a boolean e.g. in the response of a GET request to <i>cdns/{{name}}/snapshot</i>

**Note:** Anonymous Blocking requires an anonymous IP address database from the Delivery Service’s Geolocation Provider. E.g. [MaxMind’s Anonymous IP Database](#) when MaxMind is used as the Geolocation Provider.

### See also:

The *Configure Anonymous Blocking* “Quick-How-To” guide.

## Cache URL Expression

Deprecated since version 3.0: This feature is no longer supported by ATS and consequently it will be removed from Traffic Control in the future. Current plans are to remove after ATC 5.X is no longer supported.

Manipulates the cache key of the incoming requests. Normally, the cache key is the *Origin* domain. This can be changed so that multiple services can share a cache key, can also be used to preserve cached content if service origin is changed.

**Warning:** This field provides access to a feature that was only present in ATS 6.X and earlier. As *cache servers* must now use ATS 7.1.X, this field **must** be blank unless all *cache servers* can be guaranteed to use that older ATS version (**NOT** recommended).

## CDN

A CDN to which this Delivery Service belongs. Only *cache servers* within this CDN are available to route content for this Delivery Service. Additionally, only Traffic Routers assigned to this CDN will perform said routing. Most often `cdn/CDN` refers to the *name* of the CDN to which the Delivery Service belongs, but occasionally (most notably in the payloads and/or query parameters of certain *Traffic Ops API* endpoints) it actually refers to the *integral, unique identifier* of said CDN.

## Check Path

A request path on the *origin server* which is used to by certain *Traffic Ops Extensions* to indicate the “health” of the *Origin*.

## Consistent Hashing Regular Expression

When Traffic Router performs *Consistent Hashing* on a client request to find an *Edge-tier cache server* to which to redirect them, it can optionally first modify the request path by extracting the pieces that match this regular expression.

**See also:**

*Consistent Hashing Patterns*

Table 17: Aliases

Name	Use(s)	Type(s)
consisten-tHashRegex	In source code and <i>Traffic Ops API</i> requests and responses	unchanged (regular expression)
pattern-based consistent hashing	documentation and the Traffic Portal UI	unchanged (regular expression), but usually used when discussing the concept rather than the field

## Consistent Hashing Query Parameters

When Traffic Router performs *Consistent Hashing* on a client request to find an *Edge-tier cache server* to which to redirect them, it can optionally take into account any number of query parameters. This field defines them, formally as a Set but often represented as an Array/List due to encoding limitations. That is, if the Consistent Hashing Query Parameters on a Delivery Service are `{test}` and a client makes a request for `/?test=something` they will be directed to a different *cache server* than a different client that requests `/?test=somethingElse`, but the *same cache server* as a client that requests `/?test=something&quest=somethingToo`.

Table 18: Aliases

Name	Use(s)	Type(s)
consistentHash-Query-Params	In source code, Traffic Portal, and <i>Traffic Ops API</i> requests and responses	unchanged (Array of strings - should ALWAYS be unique, thus treated as a Set in most contexts)

## Deep Caching

Controls the *Deep Caching* feature of Traffic Router when serving content for this Delivery Service. This should always be represented by one of two values:

**ALWAYS** This Delivery Service will always use *Deep Caching*

**NEVER** This Delivery Service will never use *Deep Caching*

---

## Implementation Detail

Traffic Ops and Traffic Ops client Go code use an empty string as the name of the enumeration member that represents “NEVER”.

---

## Display Name

The “name” of the Delivery Service. Since nearly any use of a string-based identification method for Delivery Services (e.g. in Traffic Portal tables) uses *xml\_id*, this is of limited use. For that reason and for consistency’s sake it is suggested that this be the same as the *xml\_id*. However, unlike the *xml\_id*, this can contain any UTF-8 characters without restriction.

## DNS Bypass CNAME

When the limits placed on this Delivery Service by the *Global Max Mbps* and/or *Global Max Tps* are exceeded, a DNS-*Routed* Delivery Service will direct excess traffic to the host referred to by this CNAME (Canonical Name) record.

---

**Note:** IPv6 traffic will be redirected if and only if *IPv6 Routing Enabled* is “true” for this Delivery Service.

---

## DNS Bypass IP

When the limits placed on this Delivery Service by the *Global Max Mbps* and/or *Global Max Tps* are exceeded, a DNS-*Routed* Delivery Service will direct excess IPv4 traffic to this IPv4 address.

## DNS Bypass IPv6

When the limits placed on this Delivery Service by the *Global Max Mbps* and/or *Global Max Tps* are exceeded, a DNS-*Routed* Delivery Service will direct excess IPv6 traffic to this IPv6 address.

---

**Note:** This requires an accompanying configuration of *IPv6 Routing Enabled* such that IPv6 traffic is allowed at all.

---

## DNS Bypass TTL

When the limits placed on this Delivery Service by the *Global Max Mbps* and/or *Global Max Tps* are exceeded, a DNS-*Routed* Delivery Service will direct excess traffic to their *DNS Bypass IP*, *DNS Bypass IPv6*, or *DNS Bypass CNAME*.

## DNS TTL

The TTL on the DNS record for the Traffic Router A and AAAA records. DNS-*Routed* Delivery Services will send this TTL along with their record responses to clients requesting access to this Delivery Service. Setting too high or too low will result in poor caching performance.

Table 19: Aliases

Name	Use(s)	Type(s)
CCR DNS TTL	In Delivery Service objects returned by the <i>Traffic Ops API</i>	unchanged (int, integer etc.)
CCR TTL	Legacy Traffic Ops UI, documentation for older Traffic Control versions	unchanged (int, integer etc.)
ttl	In CDN <i>Snapshot</i> structures, where it is displayed on a per-record-type-basis	map of record type names to integral values



## DSCP

The DSCP (Differentiated Services Code Point) which will be used to mark IP packets as they are sent out of the CDN to the client.

### See also:

[The Differentiated Services Wikipedia article.](#)

**Warning:** The DSCP setting in Traffic Portal is *only* for setting traffic towards the client, and gets applied *after* the initial TCP handshake is complete and the HTTP request has been received. Before that the cache can't determine what Delivery Service is being requested, and consequently can't know what DSCP to apply. Therefore, the DSCP feature can not be used for security settings; the IP packets that form the TCP handshake are not going to be DSCP-marked.

---

## Implementation Detail

DSCP settings only apply on *cache servers* that run `APACHE TRAFFIC SERVER`. The implementation uses the [ATS Header Rewrite Plugin](#) to create a rule that will mark traffic bound outward from the CDN to the client.

---

## Edge Header Rewrite Rules

This field in general contains the contents of the a configuration file used by the [ATS Header Rewrite Plugin](#) when serving content for this Delivery Service - on *Edge-tier cache servers*.

---

**Tip:** Because this ultimately is the contents of an ATS configuration file, it can make use of the *Strings with Special Meaning to t3c*.

---

---

**Note:** This field cannot be used if the Delivery Service is assigned to a *Topology*.

---

## EDNS0 Client Subnet Enabled

A boolean value that controls whether or not EDNS0 client subnet is enabled on this Delivery Service by Traffic Router. When creating a Delivery Service in Traffic Portal, this will default to “false”.

## Example URLs

The Example URLs of a Delivery Service are the scheme/host specifications that clients can use to request content through it. These are determined by Traffic Ops from the Delivery Service's configuration, and are read-only in virtually every context. The only reason a Delivery Service should ever have no Example URLs is if it is an ANY\_MAP-*Type* Delivery Service (since they are not routed). For example, a Delivery Service that can deliver HTTP and HTTPS content, has a *Routing Name* of “cdn”, an *xml\_id* of “demo1”, and belonging to a *CDN* that is authoritative for the *mycdn.ciab.test* domain would have two Example URLs:

- `https://cdn.demo1.mycdn.ciab.test`
- `http://cdn.demo1.mycdn.ciab.test`

Note that these are irrespective of request path; meaning a client can request e.g. `https://cdn.demo1.mycdn.ciab.test/index.html` through this Delivery Service.

**Warning:** This list does not consider any *Static DNS Entries* configured on the Delivery Service, those are

Table 20: Aliases

Name	Use(s)	Type(s)
Delivery Service URLs	Traffic Portal forms	unchanged (list of strings)

## Fair-Queuing Pacing Rate Bps

The maximum bytes per second a *cache server* will deliver on any single TCP connection. This uses the Linux kernel's Fair-Queuing `setsockopt(2)` (`SO_MAX_PACING_RATE`) to limit the rate of delivery. Traffic exceeding this speed will only be rate-limited and not diverted. This option requires extra configuration on all *cache servers* assigned to this Delivery Service - specifically, the line `net.core.default_qdisc = fq` must exist in `/etc/sysctl.conf`.

See also:

`tc-fq_codel(8)`

See also:

This is implemented using the *ATS fq\_pacing plgn*.

Table 21: Aliases

Name	Use(s)	Type(s)
FQPacingRate	Traffic Ops source code, Delivery Service objects returned by the <i>Traffic Ops API</i>	unchanged (int, integer etc.)

## First Header Rewrite Rules

This field in general contains the contents of the a configuration file used by the [ATS Header Rewrite Plugin](#) when serving content for this Delivery Service - on *First-tier cache servers*.

---

**Tip:** Because this ultimately is the contents of an ATS configuration file, it can make use of the *Strings with Special Meaning to t3c*.

---



---

**Note:** This field can only be used if the Delivery Service is assigned to a *Topology*.

---

## Geo Limit

Limits access to a Delivery Service by geographic location. The only practical difference between this and *Regional Geoblocking* is the configuration method; as opposed to *Regional Geoblocking*, GeoLimit configuration is handled by country-wide codes and the *Coverage Zone File*. When a client is denied access to a requested resource on an HTTP-*Routed* Delivery Service, they will receive a 503 Service Unavailable instead of the usual 302 Found response - unless *Geo Limit Redirect URL* is defined, in which case a 302 Found response pointing to that URL will be returned by Traffic Router. If the Delivery Service is a DNS-*Routed* Delivery Service, the IP address of the *resolver* for the client DNS request is what is checked. If the IP address of this resolver is found to be in a restricted location, the Traffic Router will respond with an NXDOMAIN response, causing the name resolution to fail. This is nearly always an integral, unique identifier for a behavior set to be followed by Traffic Router. The defined values are:

- 0 Geographic access limiting is not enabled, and content served by this Delivery Service will be accessible regardless of the clients geographic location. (Aliased as “0 - None” in Traffic Portal forms)
- 1 A client will be allowed to request content if and only if their IP address is found by Traffic Router within the *Coverage Zone File*. Otherwise, access will be denied. (Aliased as “1 - CZF Only” in Traffic Portal forms)
- 2 A client will be allowed to request content if their IP address is found by Traffic Router within the *Coverage Zone File*, or if looking up the client’s IP address in the Geographic IP mapping database provided by *Geolocation Provider* indicates the client resides in a country that is found in the *Geo Limit Countries* array. (Aliased as “2 - CZF + Country Code(s)” in Traffic Portal forms - formerly was known as “CZF + US” when only the US country code was supported)

**Warning:** The definitions of each integral, unique identifier are hidden in implementations in each ATC component. Different components will handle invalid values differently, and there’s no actual enforcement that the stored integral, unique identifier actually be within the representable range.

Table 22: Aliases

Name	Use(s)	Type(s)
coverageZoneOnly	In CDN <i>Snapshot</i> structures, especially in <i>Traffic Ops API</i> responses	A boolean which, if <code>true</code> , tells Traffic Router to only service requests when the client IP address is found in the <i>Coverage Zone File</i>

**Danger:** Geographic access limiting is **not** sufficient to guarantee access is properly restricted. The limiting is implemented by Traffic Router, which means that direct requests to *Edge-tier cache servers* will bypass it entirely.

## Geo Limit Countries

When *Geo Limit* is being used with this Delivery Service (and is set to exactly 2), this is optionally a list of country codes to which access to content provided by the Delivery Service will be restricted. Normally, this is a comma-delimited string of said country codes, or an array of strings representing the country codes. When creating a Delivery Service with this field or modifying the Geo Limit Countries field on an existing Delivery Service, any amount of whitespace between country codes is permissible, as it will be removed on submission, but responses from the *Traffic Ops API* should never include such whitespace.

Table 23: Aliases

Name	Use(s)	Type(s)
geoEnabled	In CDN <i>Snapshot</i> structures, especially in <i>Traffic Ops API</i> responses	An array of objects each having the key “countryCode” that is a string containing an allowed country code - one should exist for each allowed country code

## Geo Limit Redirect URL

If *Geo Limit* is being used with this Delivery Service, this is optionally a URL to which clients will be redirected when Traffic Router determines that they are not in a geographic zone that permits their access to the Delivery Service content. This changes the response from Traffic Router from 503 Service Unavailable to 302 Found with a provided location that will be this URL. There is no restriction on the provided URL; it may even be the path to a resource served by this Delivery Service. In fact, this field need not even be a full URL, it can be a relative path. Both of these cases are handled specially by Traffic Router.

- If the provided URL is a resource served by the Delivery Service (e.g. if the client requests `http://cdn.dsXMLID.somedomain.example.com/index.html` but are denied access by *Geo Limit* and the Geo Limit Redirect URL is something like `http://cdn.dsXMLID.somedomain.example.com/help.php`), Traffic Router will find an appropriate *Edge-tier cache server* and redirect the client, ignoring Geo Limit restrictions *for this request only*.

- If the provided “URL” is actually a relative path, it will be considered *relative to the requested Delivery Service FQDN* (Fully Qualified Domain Name). This means that e.g. if the client requests `http://cdn.dsXMLID.somedomain.example.com/index.html` but are denied access by *Geo Limit* and the Geo Limit Redirect URL is something like `/help.php`, Traffic Router will find an appropriate *Edge-tier cache server* and redirect the client to it as though they had requested `http://cdn.dsXMLID.somedomain.example.com/help.php`, ignoring *Geo Limit* restrictions *for this request only*.

Table 24: Aliases

Name	Use(s)	Type(s)
NGB (National GeoBlock)	Older documentation, in Traffic Router comments and error logs	unchanged (string, String etc.)
geoRedirectURLType	Internally in Traffic Router	<p>A <code>String</code> that describes whether or not the actual Geo Limit Redirect URL is relative to the Delivery Service base FQDN. Should be one of:</p> <p><b>INVALID_URL</b> The Geo Limit Redirect URL has not yet been parsed, or an error occurred during parsing</p> <p><b>DS_URL</b> The Geo Limit Redirect URL is served by this Delivery Service</p> <p><b>NOT_DS_URL</b> The Geo Limit Redirect URL is external to this Delivery Service</p>

**Note:** The use of a redirect URL relies on the ability of Traffic Router to redirect the client using HTTP 302 Found responses. As such, this field has no effect on DNS-*Routed* Delivery Services.

## Geolocation Provider

This is nearly always the integral, unique identifier of a provider for a database that maps IP addresses to geographic locations. Less frequently, this may be accompanied by the actual name of the provider. Only two values are possible at the time of this writing:

**0: MaxMind** IP address to geographic location mapping will be provided by a [MaxMind GeoIP2 database](#).

**1: Neustar** IP address to geographic location mapping will be provided by a [Neustar GeoPoint IP address database](#).

**Warning:** It's not clear whether Neustar databases are actually supported; this is an old option and compatibility may have been broken over time.

Table 25: Aliases

Name	Use(s)	Type(s)
geo-Provider	Traffic Ops and Traffic Ops client code, <i>Traffic Ops API</i> requests and responses	unchanged (integral, unique identifier)

## Geo Miss Default Latitude

Default Latitude for this Delivery Service. When the geographic location of the client cannot be determined, they will be routed as if they were at this latitude.

Table 26: Aliases

Name	Use(s)	Type(s)
missLat	In <i>Traffic Ops API</i> responses and Traffic Ops source code	unchanged (numeric)

## Geo Miss Default Longitude

Default Longitude for this Delivery Service. When the geographic location of the client cannot be determined, they will be routed as if they were at this longitude.

Table 27: Aliases

Name	Use(s)	Type(s)
miss-Long	In <i>Traffic Ops API</i> responses and Traffic Ops source code	unchanged (numeric)

## Global Max Mbps

The maximum Mbps (Megabits per second) this Delivery Service can serve across all *Edge-tier cache servers* before traffic will be diverted to the bypass destination. For a DNS-*Routed* Delivery Service, the *DNS Bypass IP* or *DNS Bypass IPv6* will be used (depending on whether this was a A or AAAA request), and for HTTP-*Routed* Delivery Services the *HTTP Bypass FQDN* will be used.

Table 28: Aliases

Name	Use(s)	Type(s)
totalKbpsThreshold	In <i>Traffic Ops API</i> re- sponses - most notably <i>cdns/{{name}}/configs/monitoring</i>	unchanged (numeric), but converted from MBPS to KBPS (kilobits per sec- ond)

## Global Max TPS

The maximum TPS (Transactions per Second) this Delivery Service can serve across all *Edge-tier cache servers* before traffic will be diverted to the bypass destination. For a DNS-*Routed* Delivery Service, the *DNS Bypass IP* or *DNS Bypass IPv6* will be used (depending on whether this was a A or AAAA request), and for HTTP-*Routed* Delivery Services the *HTTP Bypass FQDN* will be used.

Table 29: Aliases

Name	Use(s)	Type(s)
totalTpsThreshold	In <i>Traffic Ops API</i> responses - most notably <i>cdns/{{name}}/configs/monitoring</i>	unchanged (nu- meric)

## HTTP Bypass FQDN

When the limits placed on this Delivery Service by the *Global Max Mbps* and/or *Global Max Tps* are exceeded, an HTTP-*Routed* Delivery Service will direct excess traffic to this FULLY QUALIFIED DOMAIN NAME.

## IPv6 Routing Enabled

A boolean value that controls whether or not clients using IPv6 can be routed to this Delivery Service by Traffic Router. When creating a Delivery Service in Traffic Portal, this will default to “true”.

### Info URL

This should be a URL (though neither the *Traffic Ops API* nor the Traffic Ops Database in any way enforce the validity of said URL) to which administrators or others may refer for further information regarding a Delivery Service - e.g. a related JIRA ticket.

### Initial Dispersion

The number of *Edge-tier cache servers* across which a particular asset will be distributed within each *Cache Group*. For most use-cases, this should be 1, meaning that all clients requesting a particular asset will be directed to 1 *cache server* per *Cache Group*. Depending on the popularity and size of assets, consider increasing this number in order to spread the request load across more than 1 *cache server*. The larger this number, the more copies of a particular asset are stored in a *Cache Group*, which can “pollute” caches (if load distribution is unnecessary) and decreases caching efficiency (due to cache misses if the asset is not requested enough to stay “fresh” in all the caches).

### Inner Header Rewrite Rules

This field in general contains the contents of the a configuration file used by the *ATS Header Rewrite Plugin* when serving content for this Delivery Service - on *Inner-tier cache servers*.

---

**Tip:** Because this ultimately is the contents of an ATS configuration file, it can make use of the *Strings with Special Meaning to t3c*.

---

---

**Note:** This field can only be used if the Delivery Service is assigned to a *Topology*.

---

### Last Header Rewrite Rules

This field in general contains the contents of the a configuration file used by the *ATS Header Rewrite Plugin* when serving content for this Delivery Service - on *Last-tier cache servers*.

---

**Tip:** Because this ultimately is the contents of an ATS configuration file, it can make use of the *Strings with Special Meaning to t3c*.

---

---

**Note:** This field can only be used if the Delivery Service is assigned to a *Topology*.

---



## Logs Enabled

A boolean switch that can be toggled to enable/disable logging for a Delivery Service.

---

**Note:** This doesn't actually do anything. It was part of the functionality for a planned Traffic Control component named "Traffic Logs" - which was never created.

---

## Long Description

Free text field that has no strictly defined purpose, but it is suggested that it contain a short description of the Delivery Service and its purpose.

Name	Use(s)	Type(s)
longDesc	Traffic Control source code and <i>Traffic Ops API</i> responses	unchanged (string, String etc.)

## Match List

A Match List is a set of regular expressions used by Traffic Router to determine whether a given request from a client should be served by this Delivery Service. Under normal circumstances this field should only ever be read-only as its contents should be generated by Traffic Ops based on the Delivery Service's configuration. These regular expressions can each be one of the following types:

**HEADER\_REGEX** This Delivery Service will be used if an HTTP Header/Value pair can be found in the clients request matching this regular expression.<sup>3</sup>

**HOST\_REGEX** This Delivery Service will be used if the requested host matches this regular expression. The host can be found using the Host HTTP Header, or as the requested name in a DNS request, depending on the *Type* of the Delivery Service.

**PATH\_REGEX** This Delivery Service will be used if the request path matches this regular expression.<sup>?</sup>

**STEERING\_REGEX** This Delivery Service will be used if this regular expression matches the *xml\_id* of one of this Delivery Service's "targets"

---

**Note:** This regular expression type can only exist in the Match List of STEERING-*Type* Delivery Services - and **not** CLIENT\_STEERING.

---



---

<sup>3</sup> These regular expression types can only appear in the Match List of HTTP-*Routed* Delivery Services.

Table 30: Aliases

Name	Use(s)	Type(s)
deliveryservice_regex	Traffic database	Ops unique, integral identifier for a regular expression

## Max DNS Answers

**DNS-routed Delivery Service** The maximum number of *Edge-tier cache server* IP addresses that the Traffic Router will include in responses to DNS requests. When provided, the *cache server* IP addresses included are rotated in each response to spread traffic evenly. This number should scale according to the amount of traffic the Delivery Service is expected to serve.

**HTTP-routed Delivery Service** If the Traffic Router profile parameter “edge.http.limit” is set, setting this to a non-zero value will override that parameter for this delivery service, limiting the number of Traffic Router IP addresses (A records) that are included in responses to DNS requests for this delivery service.

## Max Origin Connections

The maximum number of TCP connections individual *Mid-tier cache servers* are allowed to make to the *Origin Server Base URL*. A value of 0 in this field indicates that there is no maximum.

## Max Request Header Bytes

The maximum size(in bytes) of the request header that is allowed for this Delivery Service.

## Mid Header Rewrite Rules

This field in general contains the contents of the a configuration file used by the *ATS Header Rewrite Plugin* when serving content for this Delivery Service - on *Mid-tier cache servers*.

---

**Tip:** Because this ultimately is the contents of an ATS configuration file, it can make use of the *Strings with Special Meaning to t3c*.

---



---

**Note:** This field cannot be used if the Delivery Service is assigned to a *Topology*.

---

## Origin Server Base URL

The Origin Server’s base URL which includes the protocol (http or https). Example: `http://movies.origin.com`. Must not include paths, query parameters, document fragment identifiers, or username/password URL fields.

Table 31: Aliases

Name	Use(s)	Type(s)
orgServer-Fqdn	<i>Traffic Ops API</i> responses and in Traffic Control source code	unchanged (usually <code>str</code> , <code>string</code> etc.)

## Origin Shield

An experimental feature that allows administrators to list additional forward proxies that sit between the *Mid-tier* and the *Origin*. In most scenarios, this is represented (and required to be input) as a pipe (|)-delimited string.

## Profile

Either the *Name* of a *Profile* used by this Delivery Service, or the *ID* of said *Profile*.

Table 32: Aliases

Name	Use(s)	Type(s)
pro-fileId	In Traffic Control source code and some <i>Traffic Ops API</i> responses dealing with Delivery Services	Unlike the more general “Profile”, this is <i>always</i> an integral, unique identifier
pro-file-Name	In Traffic Control source code and some <i>Traffic Ops API</i> responses dealing with Delivery Services	Unlike the more general “Profile”, this is <i>always</i> a name ( <code>str</code> , <code>string</code> , etc.)

## Protocol

The protocol with which to serve content from this Delivery Service. This defines the way the Delivery Service will handle client requests that are either HTTP or HTTPS, which is distinct from what protocols are used to direct traffic. For example, this can be used to direct clients to only request content using HTTP, or to allow clients to use either HTTP or HTTPS, etc. Normally, this will be the name of the protocol handling, but occasionally this will appear as the integral, unique identifier of the protocol handling instead. The integral, unique identifiers and their associated names and meanings are:

- 0: HTTP** This Delivery Service will only accept unsecured HTTP requests. Requests made with HTTPS will fail.
- 1: HTTPS** This Delivery Service will only accept secured HTTPS requests. Requests made with HTTP will fail.

**2: HTTP AND HTTPS** This Delivery Service will accept both unsecured HTTP requests and secured HTTPS requests.

**3: HTTP TO HTTPS** When this Delivery Service is using HTTP *Content Routing* unsecured HTTP requests will be met with a response that indicates to the client that further requests must use HTTPS.

**Note:** If any other type of *Content Routing* is used, this functionality cannot be used. In those cases, a protocol setting of 3/"HTTP TO HTTPS" will result in the same behavior as 1/"HTTPS". This behavior is tracked by [GitHub Issue #3221](#)

**Warning:** The definitions of each integral, unique identifier are hidden in implementations in each ATC component. Different components will handle invalid values differently, and there's no actual enforcement that the stored integral, unique identifier actually be within the representable range.

Table 33: Aliases

Name	Use(s)	Type(s)
Protocol	<p><b>CDN <i>Snapshots</i>   An object containing the key "acceptHttps" that is a string containing a boolean which expresses whether or not Traffic Router should redirect HTTP requests to HTTPS URLs. Optionally, the key "acceptHttp" may also appear, once again a string containing a boolean that expresses whether or not Traffic Router should accept unsecured HTTP requests - this is implicitly treated as "true" by Traffic Router when it is not present.</b></p>	

## Query String Handling

Describes how query strings should be handled by the *Edge-tier cache servers* when serving content for this Delivery Service. This is nearly always expressed as an integral, unique identifier for each behavior, though in Traffic Portal a more descriptive value is typically used, or at least provided in addition to the integral, unique identifier. The allowed values and their meanings are:

- 0** For the purposes of caching, *Edge-tier cache servers* will consider URLs unique if and only if they are unique up to and including any and all query parameters. They will also pass the query parameters in their own requests to *Mid-tier cache servers* (which in turn will exhibit the same caching behavior and pass the query parameters in requests to the *Origin*).

(Aliased as “USE” in Traffic Portal tables, and “0 - use qstring in cache key, and pass up” in Traffic Portal forms)

- 1 For the purposes of caching, neither *Edge-tier* nor *Mid-tier cache servers* will consider the query parameter string when determining if a URL is stored in cache. However, the query string will still be passed in upstream requests to *Mid-tier cache servers* and in turn the *Origin*. (Aliased as “IGNORE” in Traffic Portal tables and “1 - ignore in cache key, and pass up” in Traffic Portal forms)
- 2 The query parameter string will be stripped from URLs immediately when the request is received by an *Edge-tier cache server*. This means it is never considered for the purposes of caching unique URLs and will not be passed in upstream requests. (Aliased as “DROP” in Traffic Portal tables and “2 - drop at edge” in Traffic Portal forms)

**Warning:** The implementation of dropping query parameter strings at the *Edge-tier* uses a *Regex Remap Expression* and thus Delivery Services with this type of query string handling cannot make use of *Regex Remap Expressions*.

Table 34: Aliases

Name	Use(s)	Type(s)
Qstring Handling	Traffic Portal tables	One of the Traffic Portal value aliases “USE” (0), “IGNORE” (1), “DROP” (2)
qstringIgnore	Traffic Ops code, <i>Traffic Ops API</i> requests/responses	unchanged (integral, unique identifier)

The Delivery Service’s Query String Handling can be set directly as a field on the Delivery Service object itself, or it can be overridden by a *Parameter* on a *Profile* used by this Delivery Service. The special *Parameter* named `psel.qstring_handling` and configuration file `parent.config` will have it’s contents directly inserted into the `parent.config` file on all *cache servers* assigned to this Delivery Service.

**Danger:** Using the `psel.qstring_handling` *Parameter* is **strongly** discouraged for several reasons. Firstly, at a Delivery Service level it will **NOT** change the configuration of that Delivery Service’s own Query String Handling - which will cause it to appear in Traffic Portal and in *Traffic Ops API* responses as though it were configured one way while actually behaving a different way altogether. Also, no validation is performed on the value given to it. Because it’s inserted verbatim into the `qstring` field of a line in ATS `parent.config configuration file`, a typo or an ignorant user can easily cause ATS instances on all *cache servers* assigned to that Delivery Service to fail to reload their configuration, possibly grinding entire CDNs to a halt.

#### See also:

When implemented as a *Parameter* (`psel.qstring_handling`), its value must be a valid value for the `qstring` field of a line in the ATS `parent.config` configuration file. For a description of valid values, see the [documentation for parent.config](#)

### Range Request Handling

Describes how HTTP “Range Requests” should be handled by the Delivery Service at the *Edge-tier*. This is nearly always an integral, unique identifier for the behavior set required of the *Edge-tier cache servers*. The valid values and their respective meanings are:

**0** Do not cache Range Requests at all. (Aliased as “0 - Don’t cache” in Traffic Portal forms)

---

**Note:** This is not retroactive - when modifying an existing Delivery Services to have this value for “Range Request Handling”, ranges requested from files that are already cached due to a non-range request will be served out of cache for as long as the Cache-Control headers allow.

---

- 1 Use the `background_fetch` plugin to service the range request while caching the whole object. (Aliased as “1 - Use background\_fetch plugin” in Traffic Portal forms)
- 2 Use the `cache_range_requests` plugin to cache ranges as unique objects. (Aliased as “2 - Use cache\_range\_requests plugin” in Traffic Portal forms)
- 3 Use the `slice` plugin to slice range based requests into deterministic chunks. (Aliased as “3 - Use slice plugin” in Traffic Portal forms)

New in version ATCv4.1.

---

**Note:** Range Request Handling can only be implemented on *cache servers* using ATS because of its dependence on ATS plugins. The value may be set on any Delivery Service, but will have no effect when the *cache servers* that ultimately end up serving the content are e.g. Grove, Nginx, etc.

---

**Warning:** The definitions of each integral, unique identifier are hidden in implementations in each ATC component. Different components will handle invalid values differently, and there’s no actual enforcement that the stored integral, unique identifier actually be within the representable range.

### Range Slice Request Block Size

The block size in bytes that is used for `slice` plugin.

This can only and must be set if the *Range Request Handling* is set to 3.

## Raw Remap Text

For HTTP and DNS-*Routed* Delivery Services, this will be added to the end of a line in the `remap.config` ATS configuration file line on the cache verbatim. For ANY\_MAP-*Type* Delivery Services this must be defined.

**Tip:** Because this ultimately is a raw line of content in a configuration file, it can make use of the *Strings with Special Meaning to t3c*. Of particular note is the *Remap Override* template string.

**Note:** This field **must** be defined on ANY\_MAP-*Type* Delivery Services, but is otherwise optional.

### See also:

The Apache Trafficserver documentation for the Regex Remap plugin

Table 35: Aliases

Name	Use(s)	Type(s)
remap-Text	In Traffic Ops source code and <i>Traffic Ops API</i> requests/responses	unchanged (text, string etc.)

### Directives “”

The Raw Remap text is ordinarily added at the end of the line, after everything else. However, it may be necessary to add Range Request Handling after the Raw Remap. For example, if you have a plugin which manipulates the Range header. In this case, you can insert the text `__RANGE_DIRECTIVE__` in the Raw Remap text, and the range request handling directives will be added at that point.

For example, if you have an Apache Traffic Server lua plugin which manipulates the range, and are using Slice Range Request Handling which needs to run after your plugin, you can set a Raw Remap, `@plugin=tslua.so @pparam=range.lua __RANGE_DIRECTIVE__`, and the `@plugin=slice.so` range directive will be inserted after your plugin.

## Regex Remap Expression

Allows remapping of incoming requests URL using regular expressions to search and replace text. In a more literal sense, this is the raw contents of a configuration file used by the *ATS regex\_remap plugin*. At its most basic, the contents of this field should consist of `map` followed by a regular expression and then a “template URL” - all space-separated. The regular expression matches a client’s request *path* (i.e. not a full URL - `/path/to/content` **not** `https://origin.example.com/path/to/content`) and when such a match occurs, the request is transformed into a request for the template URL. The most basic usage of the template URL is to use `$1-$9` to insert the corresponding regular expression capture group. For example, a regular expression of `^/a/(.*)` and a template URL of `https://origin.example.com/b/$1`

maps requests for *Origin* content under path /a/ to the same sub-paths under path b. Note that since it's a full URL, this mapping can be made to another server entirely.

### See also:

The [documentation for the regex\\_remap plugin](#) for ATS

**Caution:** This field is not validated by Traffic Ops to be correct syntactically, and can cause Traffic Server to not start if invalid. Please use with caution.

**Warning:** Regex remap expressions are incompatible with *Query String Handling* being set to 2. The behavior of a *cache server* under that configuration is undefined.

**Tip:** It is, of course, entirely possible to write a Regex Remap Expression that reproduces the desired *Query String Handling* as well as any other desired behavior.

### See also:

The [Apache Trafficserver documentation for the Regex Remap plugin](#)

Table 36: Aliases

Name	Use(s)	Type(s)
regexRemap	Traffic Ops source code and database, and <i>Traffic Ops API</i> requests/responses	unchanged (string etc.)

## Regional Geoblocking

A boolean value that defines whether or not *Regional Geoblocking* is active on this Delivery Service. The actual configuration of *Regional Geoblocking* is done in the *Profile* used by the Traffic Router serving the Delivery Service. Rules for this Delivery Service may exist, but they will not actually be used unless this field is `true`.

**Tip:** *Regional Geoblocking* is configured primarily with respect to Canadian postal codes, so unless specifically Canadian regions should be allowed/disallowed to access content, *Geo Limit* is probably a better setting for controlling access to content according to geographic location.



## Required Capabilities

New in version ATCv4.

A Delivery Service can be associated with *Server Capabilities* that it requires *cache servers* serving its content to have. When one or more *Server Capability* is required by a Delivery Service, it will block the assignment of *cache servers* to it that do not have those *Server Capabilities*. Additionally, the *Edge-tier cache servers* assigned to a Delivery Service that requires a *Server Capability* will only request content they do not have cached from *Mid-tier cache servers* which also have this *Server Capability*.

Typically, a required *Server Capability* is represented merely by the name of said *Server Capability*. In fact, there's nothing more to a *Server Capability* than its name; it's the responsibility of CDN operators to ensure that they are assigned and required properly. There is no mechanism to detect whether or not a *cache server* has a given *Server Capability*, it must be assigned manually.

## Routing Name

A DNS label in the Delivery Service's domain that forms the FQDN that is used by clients to request content. All together, the constructed FQDN looks like: *Delivery Service xml\_id.CDN Subdomain.CDN Domain.Top-Level Domain*<sup>1</sup>.

## Servers

Servers can be assigned to Delivery Services using the *Servers* and *Delivery Services* Traffic Portal sections, or by directly using the *deliveryserviceserver* endpoint. Only *Edge-tier cache servers* can be assigned to a Delivery Service, and once they are so assigned they will begin to serve content for the Delivery Service (after updates are queued and then applied). Any servers assigned to a Delivery Service must also belong to the same *CDN* as the Delivery Service itself. At least one server must be assigned to a Delivery Service in order for it to serve any content.

## Service Category

A service category is a tag that describes the type of content being delivered by the Delivery Service. Some example values are: "Linear" and "VOD"

---

<sup>1</sup> Some things to consider when choosing an *xml\_id* and routing name: the name should be descriptive and unique, but as brief as possible to avoid creating a monstrous FQDN. Also, because these are combined to form an FQDN, they should not contain any characters that are illegal for a DNS subdomain, e.g. . (period/dot). Finally, the restrictions on what characters are allowable (especially in *xml\_id*) are, in general, **NOT** enforced by the *Traffic Ops API*, so take care that the name is appropriate. See [RFC 1035](#) for exact guidelines.

## Signing Algorithm

URLs/URIs may be signed using one of two algorithms before a request for the content to which they refer is sent to the *Origin* (which in practice can be any upstream network). At the time of this writing, this field is restricted within the Traffic Ops Database to one of two values (or NULL/"None", to indicate no signing should be done).

### See also:

The `url_sig` [README](#).

### See also:

The draft RFC for `uri_signing` - note, however that the current implementation of `uri_signing` uses Draft 12 of that RFC document, **NOT** the latest.

**url\_sig** URL signing will be implemented in this Delivery Service using the `url_sig` [Apache Traffic Server plugin](#). (Aliased as "URL Signature Keys" in Traffic Portal forms)

**uri\_signing** URL signing will be implemented in this Delivery Service using an algorithm based on a work-in-progress RFC specification draft. (Aliased as "URI Signing Keys" in Traffic Portal forms)

Table 37: Aliases

Name	Use(s)	Type(s)
Signed	In all components prior to Traffic Control v2.2. Some endpoints in early versions of the <i>Traffic Ops API</i> will still return this field instead of "signingAlgorithm".	A boolean value where <code>true</code> was the same as "url_sig" in current versions, and <code>false</code> indicated URL signing would not be done for the Delivery Service.

Keys for either algorithm can be generated within *Traffic Portal*.

## SSL Key Version

An integer that describes the version of the SSL key(s) - if any - used by this Delivery Service. This is incremented whenever Traffic Portal generates new SSL keys for the Delivery Service.

**Warning:** This number will not be correct if keys are manually replaced using the API, as the key generation API does not increment it!

## Static DNS Entries

Static DNS Entries can be added *under* a Delivery Service’s domain. These DNS records can be configured in the [Delivery Services](#) section of Traffic Portal, and can be any valid CNAME, A or AAAA DNS record - provided the associated hostname falls within the DNS domain for the Delivery Service. For example, a Delivery Service with [xml\\_id](#) “demo1” and belonging to a [CDN](#) with domain “mycdn.ciab.test” could have Static DNS Entries for hostnames “foo.demo1.mycdn.ciab.test” or “foo.bar.demo1.mycdn.ciab.test” but not “foo.bar.mycdn.ciab.test” or “foo.bar.test”.

---

**Note:** The [Routing Name](#) of a Delivery Service is not part of the SOA (Start of Authority) record for the Delivery Service’s domain, and so there is no need to place Static DNS Entries below a domain containing it.

---

## Tenant

The [Tenant](#) who owns this Delivery Service. They (and their parents, if any) are the only ones allowed to make changes to this Delivery Service. Typically, `tenant/Tenant` refers to the *name* of the owning [Tenant](#), but occasionally (most notably in the payloads and/or query parameters of certain [Traffic Ops API](#) requests) it actually refers to the *integral, unique identifier* of said [Tenant](#).

Table 38: Aliases

Name	Use(s)	Type(s)
TenantID	Go code and <a href="#">Traffic Ops API</a> requests/responses	Integral, unique identifier (bigint, int etc.)

## TLS Versions

The versions of TLS that can be used in HTTP requests to [Edge-tier cache servers](#) for this Delivery Service’s content can be limited using this property. When a Delivery Service has this property set to anything other than a `null` value, it lists the versions that will be allowed. Any versions can be added to the supported set, so long as they are of the form *Major.Minor*, e.g. 1.1 or 42.0. When this is a `null` value, no restrictions are placed on the TLS versions that may be used for retrieving Delivery Service content.

---

## Implementation Detail

Traffic Ops will accept empty arrays as a synonym for `null` in requests, but will always represent them as `null` in responses. Note that this means it’s impossible to create a Delivery Service that explicitly supports no TLS versions - the proper way to disable HTTPS for a Delivery Service is to set its [Protocol](#) accordingly.

---

A Delivery Service that has a *Type* of STEERING or CLIENT\_STEERING may not legally be set to have a TLS Versions property that is non-null.

**Warning:** Using this setting may cause old clients that only support archaic TLS versions to break suddenly. Be sure that the security increase is worth this risk.

## Topology

A structure composed of *Cache Groups* and parent relationships, which is assignable to one or more *Delivery Services*.

## Traffic Router Additional Response Headers

List of HTTP header `{{name}}:{{value}}` pairs separated by `__RETURN__` or simply on separate lines. Listed pairs will be included in all HTTP responses from Traffic Router for HTTP-*Routed* Delivery Services.

Deprecated since version 4.0: The use of `__RETURN__` as a substitute for a real newline is unnecessary and the ability to do so will be removed in the future.

Table 39: Aliases

Name	Use(s)	Type(s)
trResponse- Headers	Traffic Control source code and Delivery Service objects returned by the <i>Traffic Ops API</i>	unchanged (string etc.)

## Traffic Router Log Request Headers

List of HTTP header names separated by `__RETURN__` or simply on separate lines. Listed pairs will be logged for all HTTP requests to Traffic Router for HTTP-*Routed* Delivery Services.

Deprecated since version 4.0: The use of `__RETURN__` as a substitute for a real newline is unnecessary and the ability to do so will be removed in the future.

Table 40: Aliases

Name	Use(s)	Type(s)
trRequest- Headers	Traffic Control source code and Delivery Service objects returned by the <i>Traffic Ops API</i>	unchanged (string etc.)

## Type

Defines the content routing method used by the Delivery Service. In most cases this is an integral, unique identifier that corresponds to an enumeration of the Delivery Service Types. In other cases, this is the actual name of said type.

The “Type” of a Delivery Service can mean several things. First, it can be used to refer to the “routing type” of Delivery Service. This is one of:

---

**Tip:** The only way to get the integral, unique identifier of a *Type* of Delivery Service is to look at the database after it has been generated; these are non-deterministic and cannot be guaranteed to have any particular value, or even consistent values. This can be done directly or, preferably, using the *types* endpoint. Unfortunately, knowing the name of the *Type* is rarely enough for many applications. The `useInColumn` values of these *Types* will be `deliveryservice`.

---

**DNS** Delivery Services of this routing type are routed by Traffic Router by providing DNS records that provide the IP addresses of *cache servers* when clients look up the full Delivery Service FQDN.

**HTTP** The Traffic Router(s) responsible for routing this Delivery Service will still answer DNS requests for the Delivery Service FQDN, but will provide its own IP address. The client then directs its HTTP request to the Traffic Router, which will use an *HTTP redirection response* to direct the client to a *cache server*.

More generally, though, Delivery Services have a Type that defines not only how traffic is routed, but also how content is cached and semantically defines what “content” means in the context of a given Delivery Service.

**ANY\_MAP** This is a special kind of Delivery Service that should only be used when control over the clients is guaranteed, and very fine control over the ATS `remap.config` line for this Delivery Service is required. ANY\_MAP is not known to Traffic Router. It is not routed in any way. For Delivery Services of this type, the “Raw Remap Text” field **must** be defined, as it is the only configuration generated by Traffic Control. The only way for a client to utilize delivery through an ANY\_MAP service is by knowing in advance the IP address of one or more *Edge-tier cache servers* and make the appropriate request(s).

**DNS** Uses DNS content routing. Delivers content normally. This is the recommended Type for delivering smaller objects like web page images.

**DNS\_LIVE<sup>2</sup>** Uses DNS Content routing, but optimizes caching for live video streaming. Specifically, the configuration generated for *cache servers* responsible for serving content for this Delivery Service will not cache that content on storage disks. Instead, they will make use of RAM block devices dedicated to ATS - as specified by the special `RAM_Drive_Prefix` and `RAM_Drive_Letters` *Parameters*. Also, any *Mid-tier* of caching is bypassed.

**DNS\_LIVE\_NATNL** Works exactly the same as DNS\_LIVE, but is optimized for delivery of live video content across a wide physical area. What this means is that the *Mid-tier*

---

<sup>2</sup> These Delivery Services Types are vulnerable to what this writer likes to call the “Duplicate Origin Problem”. This problem is tracked by [Issue #3537](#).

of caching is **not** bypassed, unlike DNS\_LIVE. The *Mid-tier* will also use block RAM devices.

**HTTP** Uses HTTP content routing, delivers content normally. This is the recommended Type for delivering larger objects like video streams.

**HTTP\_LIVE**<sup>Page 45, 2</sup> Uses HTTP Content routing, but optimizes caching for live video streaming. Specifically, the configuration generated for *cache servers* responsible for serving content for this Delivery Service will not cache that content on storage disks. Instead, they will make use of RAM block devices dedicated to ATS - as specified by the special `RAM_Drive_Prefix` and `RAM_Drive_Letters` *Parameters*. Also, any *Mid-tier* of caching is bypassed.

**HTTP\_LIVE\_NATNL** Works exactly the same as HTTP\_LIVE, but is optimized for delivery of live video content across a wide physical area. What this means is that the *Mid-tier* of caching is **not** bypassed, unlike HTTP\_LIVE. The *Mid-tier* will also use block RAM devices.

**HTTP\_NO\_CACHE**<sup>Page 45, 2</sup> Uses HTTP Content Routing, but *cache servers* will not actually cache the delivered content - they act as just proxies. This will bypass any existing *Mid-tier* entirely (as it's totally useless when content is not being cached).

**STEERING** This is a sort of “meta” Delivery Service. It is used for directing clients to one of a set of Delivery Services, rather than delivering content directly itself. The Delivery Services to which a STEERING Delivery Service routes clients are referred to as “targets”. Targets in general have an associated “value” and can be of several *Types* that define the meaning of the value - these being:

**STEERING\_ORDER** The value of a STEERING\_ORDER target sets a strict order of preference. In cases where a response to a client contains multiple Delivery Services, those targets with a lower “value” appear earlier than those with a higher “value”. In cases where two or more targets share the same value, they each have an equal chance of being presented to the client - effectively spreading traffic evenly across them.

**STEERING\_WEIGHT** The values of STEERING\_WEIGHT targets are interpreted as “weights”, which define how likely it is that any given client will be routed to a specific Delivery Service - effectively this determines the spread of traffic across each target.

The targets of a Delivery Service may be set using *the appropriate section of Traffic Portal* or via the `steering/{ID}/targets` and `steering/{ID}/targets/{targetID}` *Traffic Ops API* endpoints.

**See also:**

For more information on setting up a STEERING (or CLIENT\_STEERING) Delivery Service, see *Configure Delivery Service Steering*.

**See also:**

For implementation details about how Traffic Router routes STEERING (and CLIENT\_STEERING) Delivery Services, see *Steering Feature*.

**CLIENT\_STEERING** A CLIENT\_STEERING Delivery Service is exactly like STEERING except that it provides clients with methods of bypassing the weights, orders, and localizations of targets in order to choose any arbitrary target at will. When utilizing these methods, the client will either directly choose a target immediately or request a list of all available targets from Traffic Router and then choose one to which to send a subsequent request for actual content. CLIENT\_STEERING also supports two additional target types:

**STEERING\_GEO\_ORDER** These targets behave exactly like STEERING\_ORDER targets, but Delivery Services are grouped according to the “locations” of their *Origins*. Before choosing a Delivery Service to which to direct the client, Traffic Router will first create subsets of choices according to these groupings, and order them by physical distance from the client (closest to farthest). Within these subsets, the values of the targets establish a strict precedence ordering, just like STEERING\_ORDER targets.

**STEERING\_GEO\_WEIGHT** These targets behave exactly like STEERING\_WEIGHT targets, but Delivery Services are grouped according to the “locations” of their *Origins*. Before choosing a Delivery Service to which to direct the client, Traffic Router will first create subsets of choices according to these groupings, and order them by physical distance from the client (closest to farthest). Within these subsets, the values of the targets establish the likelihood that any given target within the subset will be chosen for the client - effectively determining the spread of traffic across targets within that subset.

---

**Important:** To make use of the STEERING\_GEO\_ORDER and/or STEERING\_GEO\_WEIGHT target types, it is first necessary to ensure that at least the “primary” *Origin* of the *Delivery Service* has an associated geographic coordinate pair. This can be done either from the *Origins* page in Traffic Portal, or using the *origins Traffic Ops API* endpoint.

---



---

**Note:** “Steering” is also commonly used to collectively refer to either of the kinds of Delivery Services that can participate in steering behavior (STEERING and CLIENT\_STEERING).

---

Table 41: Aliases

Name	Use(s)	Type(s)
Content Routing Type	Traffic Portal forms	The name of any of the Delivery Service <i>Types</i> (string)
TypeID	In Go code and <i>Traffic Ops API</i> requests/responses	Integral, unique identifier (bigint, int etc.)



## Use Multi-Site Origin Feature

A boolean value that indicates whether or not this Delivery Service serves content for an *Origin* that provides content from two or more redundant servers. There are very few good reasons for this to not be false. When true, Traffic Ops will configure *Mid-tier cache servers* to perform load-balancing and other optimizations for redundant *origin servers*.

Naturally, this assumes that each redundant server is exactly identical, from request paths to actual content. If Multi-Site Origin is configured for servers that are *not* identical, the client's experience is undefined. Furthermore, the *origin servers* may have differing IP addresses, but **must** serve content for a single FQDN - as defined by the Delivery Service's *Origin Server Base URL*. These redundant servers **must** be configured as servers (server *Type* ORG) in Traffic Ops - either using the *appropriate section of Traffic Portal* or the *servers* endpoint.

---

**Important:** In order for a given *Mid-tier cache server* to support Multi-Site Origins, the value of a *Parameter* named `http.parent_proxy_routing_enable` in configuration file `records.config` must be set to 1 on that server's *Profile*. If using an optional secondary grouping of Multi-Site Origins, the *Parameter* named `url_remap.remap_required` in configuration file `records.config` must also be set to 1 on that *Profile*. These settings must be applied to all *Mid-tier cache servers* that are the *parents* of any *Edge-tier cache server* assigned to this Delivery Service.

### See also:

These parameters are described in the ATS documentation sections for [Parent Proxy Configuration](#) and [URL Remap Rules](#), respectively.

---

Table 42: Aliases

Name	Use(s)	Type(s)
multiSiteOrigin	In Go code and <i>Traffic Ops API</i> requests/responses	unchanged (bool, boolean etc.)
MSO (Multi-Site Origin)	In documentation and used heavily in discussion in Slack, mailing list etc.	unchanged (usually only used where implicitly true)

A Delivery Service *Profile* can have *Parameters* that affect Multi-Site Origin configuration. These are detailed in *Parameters that Affect Multi-Site Origin*.

### See also:

A quick guide on setting up Multi-Site Origins is given in *Configure Multi-Site Origin*.



## xml\_id

A text-based unique identifier for a Delivery Service. Many *Traffic Ops API* endpoints and internal ATC functions use this to uniquely identify a Delivery Service as opposed to the historically favored “ID”. This string will become a part of the CDN service domain, which all together looks like: *Delivery Service Routing Name.Delivery Service xml\_id.CDN Subdomain.CDN Domain.Top-Level Domain*. Must be all lowercase, no spaces or special characters, but may contain dashes/hyphens<sup>?</sup>.

Table 43: Aliases

Name	Use(s)	Type(s)
Key	Traffic Portal tables and forms	unchanged (string)

## Delivery Service Parameters

Features which are new, experimental, or not significant enough to be first-class Delivery Service fields are often added as *Parameters*. To use these, add a *Profile* to the Delivery Service, with the given *Parameter* assigned.

## parent.config

The following *Parameters* must have the *Config File* `parent.config` to take effect - even if, strictly speaking, they aren’t used to modify the contents of the ATS `parent.config` configuration file.

### See also:

See the [Apache Traffic Server documentation for parent.config](#) and their documentation for [strategies.yaml](#) for more information on its implementation of parent selection (and in particular Multi-Site Origins).

- `try_all primaries before secondary` - on a Delivery Service *Profile*, if this exists, try all “primary parents” before “failing over” to “secondary parents”, which may be ideal if objects are unlikely to be in cache. The default behavior is to immediately fail to a secondary, which is ideal if objects are likely to be in cache, as the first consistent-hashed “secondary parent” will be the “primary parent” in its own *Cache Group* and therefore receive requests for that object from clients near its own *Cache Group*.

**Caution:** The *Value* of this *Parameter* is ignored. It is considered implicitly “truthy” if the *Parameter* is present at all on the *Profile*. This means that the *Values* `false`, `0`, and `"no"` will all result in the behavior described being adopted, contrary to what might be intuitively expected.

- `enable_h2` - On a Delivery Service Profile, if the *Value* of this *Parameter* begins with `t` or `y` (case-insensitive), HTTP/2 is enabled for client requests. ATS must also be listening for HTTP/2 - configured in `records.config` - or this will have no effect.

### Implementation Detail

This *Parameter* does not affect the contents of `parent.config`, but instead either `ssl_server_name.yaml` in ATS 8 or `sni.yaml` in ATS 9. It has the `parent.config` *Config File* value for consistency.

---

**Warning:** Interpretation of the *Value* of this *Parameter* is extremely permissive. For example, the *Values* `t`, `Y`, `True`, `yes`, `talse`, `yno`, `Yeah`, `don't do this`, `You should never under any circumstances allow HTTP/2`, and `totally horrible idea to enable this` all equally mean “true”. No part of ATC checks or warns about typos or strange *Values* for this *Parameter*, so take care to prevent typos, misspellings, and the like to avoid confusing situations.

- `tls_versions` - on a Delivery Service *Profile*, if this exists, enable the given comma-delimited<sup>4</sup> TLS versions for client requests e.g. `1.1,1.2,1.3`. ATS must also be accepting those TLS versions - configured in `records.config` - or this will have no effect.

### Implementation Detail

This *Parameter* does not affect the contents of `parent.config`, but instead either `ssl_server_name.yaml` in ATS 8 or `sni.yaml` in ATS 9. It has the `parent.config` *Config File* value for consistency.

---

**Caution:** The actual permitted TLS versions are the union of those laid out in this *Parameter* and those configured as the *TLS Versions* property of the Delivery Service.

- `use_peering` - on a Deliver Service *Profile*, if this exists and is true, the strategy `ring_mode` will be set to `peering_ring` for large library DNS support.

### Implementation Detail

This *Parameter* does not affect the contents of `parent.config`, but instead `strategies.yaml` in ATS 9. It has the `parent.config` *Config File* value for consistency.

---

Deprecated since version ATCv6.2: In *Traffic Ops API* version 4 (unstable at the time of this writing), TLS versions should be configured using the *TLS Versions* property of the Delivery Service, and support for this *Parameter* will be removed at some point after the stabilization of *Traffic Ops API* version 4.

---

<sup>4</sup> The list may also be separated by spaces or semicolons, or spread across lines.

## Parameters that Affect Multi-Site Origin

Each *Parameter* directly corresponds to a field in a line of the ATS `parent.config` file (usually by almost the same name), and documentation for these fields is provided in the form of links to their entries in the ATS documentation.

Table 44: Parameters of a Delivery Service Profile that Affect MSO Configuration

Name	ATS <code>parent.config</code> field	Effect
<code>mso.algorithm</code>	<code>round_robin</code>	Sets the algorithm used to determine from which <i>origin server</i> content will be requested.
<code>mso.max_simple_retries</code>	<code>simple_retries</code>	Sets a strict limit on the number of “simple retries” allowed before giving up
<code>mso.max_unavailable_server_retries</code>	<code>unavailable_server_retries</code>	Sets a strict limit on the number of times the <i>cache server</i> will attempt to request content from an <i>origin server</i> that has previously been considered “unavailable”.
<code>mso.parent_retry</code>	<code>parent_retry</code>	Sets whether the <i>cache servers</i> will use “simple retries”, “unavailable server retries”, or both.
<code>mso.simple_retry_response_codes</code>	<code>simple_retry_response_codes</code> <b>UNKNOWN</b>	<b>UNKNOWN</b> - supposedly defines HTTP response codes from an <i>origin server</i> that necessitate a “simple retry”.
<code>mso.unavailable_server_retry_response_codes</code>	<code>unavailable_server_retry_response_codes</code>	Defines HTTP response codes from an <i>origin server</i> that indicate it is currently “unavailable”.
<code>merge_parent_groups</code>	<code>parent</code> and <code>secondary-parent</code>	Moves each of the space-separated <i>Cache Groups</i> named in the <i>Value</i> from the secondary parent list into the primary parent list. This can be used to combine all parents into a single consistent hash ring.

**Warning:** The `mso.simple_retry_response_codes` *Parameter* has no apparent, possible use according to the ATS `parent.config` documentation. Whether or not it has any effect - let alone the *intended* effect - is not known, and its use is therefore strongly discouraged.

### See also:

To see how the *Values* of these Parameters are interpreted, refer to the [Apache Traffic Server documentation on the parent.config configuration file](#)

## 2.1.4 Introduction

Traffic Control is a CDN control plane. It is made up of a suite of applications which are used to configure, manage, and direct client traffic to a tiered system of HTTP caching proxy servers (herein referred to as *cache servers*). In principle, a CDN may be implemented with any HTTP caching proxy. The caching software chosen for Traffic Control is [Apache Traffic Server](#). Although the current release supports only ATS as a *cache server* implementation, this may change with future releases.

Traffic Control was first developed at Comcast for internal use and released to Open Source in April of 2015. Traffic Control moved into the Apache Incubator in August of 2016.

Traffic Control implements the elements illustrated in green in the diagram below.

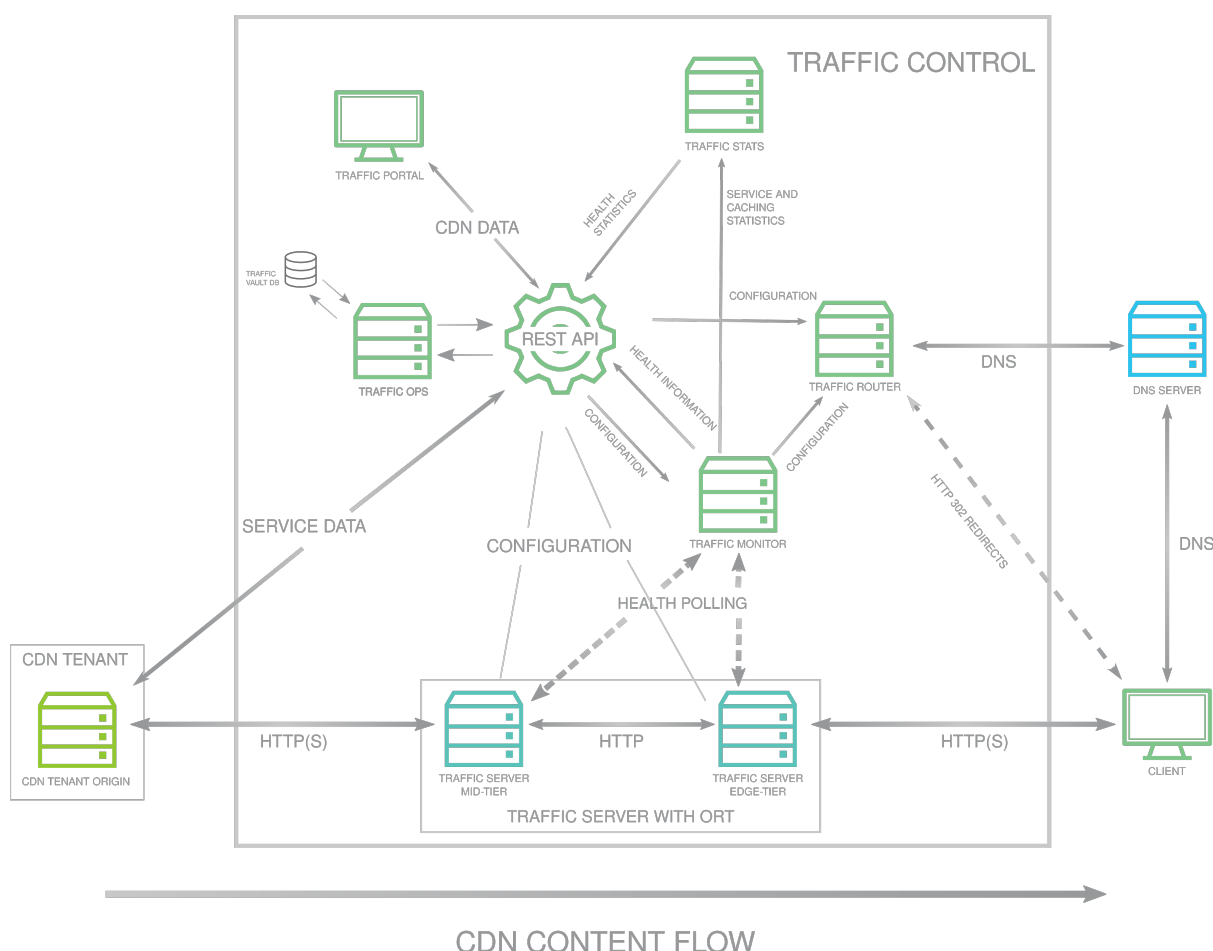


Fig. 3: Apache Traffic Control Hierarchical Diagram

**Traffic Ops** *Traffic Ops* stores the configuration of *cache servers* and *CDN Delivery Services*. It also serves the *Traffic Ops API* which can be used by tools, scripts, and programs to access and manipulate CDN data.

**Traffic Router** *Traffic Router* is used to route client requests to the closest healthy *cache server* by analyzing the health, capacity, and state of the *cache servers* according to the *Health Protocol* and relative geographic distance between each *Cache Group* and the client.

**Traffic Monitor** *Traffic Monitor* does health polling of the *cache servers* on a very short interval

to keep track of which servers should be kept in rotation.

**See also:**

*Health Protocol*

**Traffic Stats** *Traffic Stats* collects and stores real-time traffic statistics aggregated from each of the *cache servers*. This data is used by the *Traffic Router* to assess the available capacity of each *cache server* which it uses to balance traffic load and prevent overload.

**Traffic Portal** *Traffic Portal* is a web interface which uses the *Traffic Ops API* to present CDN data and the controls to manipulate it in a user-friendly interface.

New in version 2.2: As of Traffic Control 2.2, this is the recommended, official UI for the Traffic Control platform. In Traffic Control 3.x, the Traffic Ops UI has been deprecated and disabled by default, and it will be removed with the release of Traffic Control 4.0.

**Traffic Vault** *Traffic Vault* is used as a secure key/value store for SSL private keys used by other Traffic Control components.

## 2.1.5 Content Invalidation Jobs

*Content Invalidation Jobs*, or simply “jobs” as they are sometimes known, are ways of forcing *cache servers* to treat content as no longer valid, bypassing their normal caching policies.

In general, this *should* be unnecessary, because a well-behaved *Origin* *should* be setting its HTTP caching headers properly, so that content is only considered valid for some appropriate time intervals. Occasionally, however, an *Origin* will be too optimistic with its caching instructions, and when content needs to be updated, *cache servers* need to be informed that they must check back with the *Origin*. Content Invalidation Jobs allow this to be done for specific patterns of assets, so that *cache servers* will check back in with the *Origin* and verify that the content they have cached is still valid.

The model for Content Invalidation Job as API objects is given in *Content Invalidation Job as a Typescript interface*.

8: Content Invalidation Job as a Typescript interface.

```
/** This is the form used to create a new Content Invalidation Job */
interface ContentInvalidationJobCreationRequest {
  deliveryService: string;
  invalidationType: "REFRESH" | "REFETCH";
  regex: `/${string}` | `\\/${string}`; // must also be a valid_
  ↳RegExp
  startTime: Date; // RFC3339 string
  ttlHours: number;
}

/**
 * This is the form used to return representations of Content_
  ↳Invalidation
```

(continues on next page)

(continued from previous page)

```
* Requests to clients.
*/
interface ContentInvalidationJob {
    assetUrl: string;
    createdBy: string;
    deliveryService: string;
    id: number;
    invalidationType: "REFRESH" | "REFETCH";
    startTime: Date; // RFC3339 string
    ttlHours: number;
}
```

## Asset URL

This property only appears in responses from the *Traffic Ops API* (and in the bodies of PUT requests to *jobs*, where the scheme and host/authority sections of the URL is held immutable). The *Asset URL* is constructed from the *Regular Expression* used in the creation of a Content Invalidation Job and the *Origin Server Base URL* of the *Delivery Service* for which it was created. It is a URL that has a valid regular expression as its path (and may not be “percent-encoded” where a normal URL typically would be). Requests from CDN clients for content that matches this pattern will trigger Content Invalidation behavior.

## Created By

The username of the user who created the Content Invalidation Job is stored as the *Created By* property of the Content Invalidation Job.

## Delivery Service

A Content Invalidation Job can only act on content for a single *Delivery Service* - invalidating content for multiple *Delivery Services* requires multiple Content Invalidation Jobs. The *Delivery Service* property of a Content Invalidation Job holds the *xml\_id* of the *Delivery Service* on which it operates.

Changed in version 4.0: In earlier API versions, this property was allowed to be either the integral, unique identifier of the target *Delivery Service*, or its *xml\_id* - this is no longer the case, but it should always be safe to use the *xml\_id* in any case.

## ID

The integral, unique identifier for the Content Invalidation Job, assigned to it upon its creation.

## Invalidation Type

*Invalidation Type* defines how a *cache server* should go about ensuring that its cache is valid.

The normal operating mode for a Content Invalidation Job is to force the *cache server* to send a request to the *Origin* to verify that its cache is valid. If that is the case, no extra work is done and business as usual resumes. However, some *Origins* are misconfigured and do not respond as required by HTTP specification. In this case, it is strongly advised to fix the *Origin* so that it properly implements HTTP. However, if an *Origin* is sending cache-able responses to requests, and cannot be trusted to verify the validity of cached content based on cache-controlling HTTP headers (e.g. *If-Modified-Since*) instead returning responses like 304 Not Modified *even when the content has in fact been modified*, **and** if correcting this behavior is not an option, then the *cache server* may be forced to pretend that the content it has was actually invalidated by the *Origin* and must be completely re-fetched.

The two values allowed for a Content Invalidation Job's Invalidation Type are:

**REFRESH** A *REFRESH* Content Invalidation Job instructs *cache servers* to behave normally - when matching content is requested, send an upstream request to (eventually) the *Origin* with cache-controlling HTTP headers, and trust the *Origin*'s response. The vast majority of all Content Invalidation Jobs should most likely use this Invalidation Type.

**REFETCH** Rather than treating the cached content as “stale”, the *cache servers* processing a *REFETCH* Content Invalidation Job should fetch the cached content again, regardless of what the *Origin* has to say about the validity of their caches. These types of Content Invalidation Jobs cannot be created without a proper “semi-global” *refetch\_enabled Parameter*.

**Caution:** A “REFETCH” Content Invalidation Job should be used **only** when the *Origin* is not properly configured to support HTTP caching, and will return invalid or incorrect responses to conditional requests as described in section 4.3.2 of **RFC 7234**. In any other case, this will cause undo load on both the *Origin* and the requesting *cache servers*, and “REFRESH” should be used instead.

## Regular Expression

The *Regular Expression* of a Content Invalidation Job defines the content on which it acts. It is used to match URL *paths* (including the query string - but **not** including document fragments, which are not sent in HTTP requests) of content to be invalidated, and is combined with the *Origin Server Base URL* of the *Delivery Service* for which the Content Invalidation Job was created to obtain a final pattern that is made available as the *Asset URL*.

**Note:** While the *Traffic Ops API* and *Traffic Portal* both require the Regular Expression to



begin with / (so that it matches URL paths), the *Traffic Ops API* allows optionally escaping this leading character with a “backslash” \, while *Traffic Portal* does not. As / is not syntactically important to regular expressions, the use of a leading \ should be avoided where possible, and is only allowed for legacy compatibility reasons.

Table 45: Aliases/Synonyms

Name	Use(s)	Type
Path Regex	In Traffic Portal forms	unchanged (String, str, etc.)
regex	In raw <i>Traffic Ops API</i> requests and responses, internally in multiple components	unchanged (String, str, etc.)

## Start Time

Content Invalidation Jobs are planned in advance, by setting their *Start Time* to some point in the future (the *Traffic Ops API* will refuse to create Content Invalidation Jobs with a Start Time in the past). Content Invalidation Jobs will have no effect until their Start Time.

## TTL

The *TTL* of a Content Invalidation Job defines how long a Content Invalidation Job should remain in effect. This is generally expressed as an integer number of hours.

Table 46: Aliases/Synonyms

Name	Use(s)	Type
parameters	In legacy <i>Traffic Ops API</i> versions	A string, containing the TTL in the format <i>TTL:Actual TTLh</i>
ttlHours	In <i>Traffic Ops API</i> requests and responses	Unchanged (unsigned integer number of hours)

## 2.1.6 Profiles and Parameters

*Profiles* are a collection of configuration options, defined partially by the Profile’s *Type* (not to be confused with the more general “*Type*” used by many other things in Traffic Control) and partially by the *Parameters* set on them. Mainly, Profiles and Parameters are used to configure *cache servers*, but they can also be used to configure parts of (nearly) any Traffic Control component, and can even be linked with more abstract concepts like *Delivery Services* and *Cache Groups*. The vast majority of configuration done within a Traffic Control CDN must be done through *Profiles* and *Parameters*, which can be achieved either through the *Traffic Ops API* or in the *Profiles* view of Traffic Portal. For ease of use, Traffic Portal allows for duplication, comparison, import and export of *Profiles* including all of their associated *Parameters*.



**See also:**

For Delivery Service Profile Parameters, see *Delivery Service Parameters*.

**Profiles****Properties**

Profile objects as represented in e.g. the *Traffic Ops API* or in the *Traffic Portal Profiles view* have several properties that describe their general operation. In certain contexts, the *Parameters* assigned to a Profile (and/or the integral, unique identifiers thereof) may appear as properties of the Profile, but that will not appear in this section as a description of *Parameters* is provided in the section of that name.

**CDN**

A Profile is restricted to operate within a single CDN. Often, “CDN” (or “cdn”) refers to the integral, unique identifier of the CDN, but occasionally it refers to the *name* of said CDN. It may also appear as e.g. `cdnId` or `cdnName` in *Traffic Ops API* payloads and responses. A Profile may only be assigned to a server, *Delivery Service*, or *Cache Group* within the same CDN as the Profile itself.

**Description**

Profiles may have a description provided by the creating user (or Traffic Control itself in the case of the *Default Profiles*). The *Traffic Ops API* does not enforce length requirements on the description (though Traffic Portal does), and so it’s possible for Profiles to have empty descriptions, though it is strongly recommended that Profiles have meaningful descriptions.

**ID**

An integral, unique identifier for the Profile.

**Name**

Ostensibly this is simply the Profile’s name. However, the name of a Profile has drastic consequences for how Traffic Control treats it. Particularly, the name of a Profile is heavily conflated with its *Type*. These relationships are discussed further in the *Type* section, on a Type-by-Type basis.

The Name of a Profile may not contain spaces.

Changed in version ATCv6: In older versions of ATC, Profile Names were allowed to contain spaces. The *Traffic Ops API* will reject creation or update of Profiles that have spaces in their

Names as of ATC version 6, so legacy Profiles will need to be updated to meet this constraint before they can be modified.

### Routing Disabled

This property can - and in fact *must* - exist on a Profile of any *Type*, but it only has any meaning on a Profile that has a name matching the constraints placed on the names of `ATS_PROFILE-Type` Profiles. This means that it will also have meaning on Profiles of *Type* `UNK_PROFILE` that for whatever reason have names beginning with `EDGE` or `MID`. When this field is defined as 1 (may be displayed as `true` in e.g. Traffic Portal), Traffic Router will not be informed of any *Delivery Services* to which the *cache server* using this Profile may be assigned. Effectively, this means that client traffic cannot be routed to them, although existing connections would be uninterrupted.

### Type

A Profile's *Type* determines how its configured *Parameters* are treated by various components, and often even determine how the object using the Profile is treated (particularly when it is a server). Unlike the more general "*Type*" employed by Traffic Control, the allowed Types of Profiles are set in stone, and they are as follows.

**Danger:** Some of these Profile Types have strict naming requirements, and it may be noted that some of said requirements are prefixes ending with `_`, while others are either not prefixes or do not end with `_`. This is exactly true; some requirements **need** that `_` and some may or may not have it. It is our suggestion, therefore, that for the time being all prefixes use the `_` notation to separate words, so as to avoid causing headaches remembering when that matters and when it does not.

**ATS\_PROFILE** A Profile that can be used with either an Edge-tier or Mid-tier *cache server* (but not both, in general). This is the only Profile type that will ultimately pass its *Parameters* on to *ORT* in the form of generated configuration files. For this reason, it can make use of the *Strings with Special Meaning to t3c* in the values of some of its *Parameters*.

**Warning:** For legacy reasons, the names of Profiles of this type *must* begin with `EDGE` or `MID`. This is **not** enforced by the *Traffic Ops API* or Traffic Portal, but certain Traffic Control operations/components expect this and will fail to work otherwise! This includes *caches/stats*.

**DS\_PROFILE** A Profile that, rather than applying to a server, is instead *used by a Delivery Service*.

**ES\_PROFILE** A Profile for *ElasticSearch* servers. This has no known special meaning to any component of Traffic Control, but if ElasticSearch is in use the use of this Profile Type is suggested regardless.

**GROVE\_PROFILE** A Profile for use with the experimental Grove HTTP caching proxy.

**INFLUXDB\_PROFILE** A Profile used with [InfluxDB](#), which is used by Traffic Stats.

**KAFKA\_PROFILE** A Profile for [Kafka](#) servers. This has no known special meaning to any component of Traffic Control, but if Kafka is in use the use of this Profile Type is suggested regardless.

**LOGSTASH\_PROFILE** A Profile for [Logstash](#) servers. This has no known special meaning to any component of Traffic Control, but if Logstash is in use the use of this Profile Type is suggested regardless.

**ORG\_PROFILE** A Profile that may be used by either *origin servers* or *Origins* (no, they aren't the same thing).

**RIAK\_PROFILE** A Profile used for a Traffic Vault server.

---

### Implementation Detail

This Profile Type gets its name from the legacy implementation of Traffic Vault: Riak KV.

---

**SPLUNK\_PROFILE** A Profile meant to be used with [Splunk](#) servers. This has no known special meaning to any component of Traffic Control, but if Splunk is in use the use of this Profile Type is suggested regardless.

**TM\_PROFILE** A Traffic Monitor Profile.

**Warning:** For legacy reasons, the names of Profiles of this type *must* begin with `RASCAL_`. This is **not** enforced by the [Traffic Ops API](#) or Traffic Portal, but certain Traffic Control operations/components expect this and will fail to work otherwise!

**TP\_PROFILE** A Traffic Portal Profile. This has no known special meaning to any Traffic Control component(s) (not even Traffic Portal itself), but its use is suggested for the profiles used by any and all Traffic Portal servers anyway.

**TR\_PROFILE** A Traffic Router Profile.

**See also:**

[The Traffic Router Profile](#)

**TS\_PROFILE** A Traffic Stats Profile.

**UNK\_PROFILE** A catch-all type that can be assigned to anything without imbuing it with any special meaning or behavior.

---

**Tip:** A Profile of the wrong type assigned to a Traffic Control component *will* (in general) cause it to function incorrectly, regardless of the [Parameters](#) assigned to it.

---

### Default Profiles

Traffic Control comes with some pre-installed Profiles for its basic components, but users are free to define their own as needed. Additionally, these default Profiles can be modified or even removed completely. One of these Profiles is *The GLOBAL Profile*, which has a dedicated section.

**INFLUXDB** A Profile used by InfluxDB servers that store Traffic Stats information. It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

**RIAK\_ALL** This Profile is used by Traffic Vault, which is, generally speaking, the only instance in Traffic Control as it can store keys for an arbitrary number of CDNs. It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

**TRAFFIC\_ANALYTICS** A default Profile that was intended for use with the now-unplanned “Traffic Analytics” ATC component. It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

**TRAFFIC\_OPS** A Profile used by the Traffic Ops server itself. It’s suggested that any and all “mirrors” of Traffic Ops for a given Traffic Control instance be recorded separately and all assigned to this Profile for record-keeping purposes. It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

**TRAFFIC\_OPS\_DB** A Profile used by the PostgreSQL database server that stores all of the data needed by Traffic Ops. It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

**TRAFFIC\_PORTAL** A Profile used by Traffic Portal servers. This profile name has no known special meaning to any Traffic Control components (not even Traffic Portal itself), but its use is suggested for Traffic Portal servers anyway. It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

**TRAFFIC\_STATS** This is the **only** Profile used by Traffic Stats (though InfluxDB servers have their own Profile(s)). It has a *Type* of UNK\_PROFILE and is assigned to the special “ALL” *CDN*.

In addition to these Profiles, each release of Apache Traffic Control is accompanied by a set of suggested Profiles suitable for import in the *Profiles* view of Traffic Portal. They may be found on [the Profiles Downloads Index page](#). These Profiles are typically built from production Profiles by a company using Traffic Control, and as such are typically highly specific to the hardware and network infrastructure available to them. **None of the Profiles bundled with a release are suitable for immediate use without modification**, and in fact many of them cannot actually be imported directly into a new Traffic Control environment, because Profiles with the same *Names* already exist (as above).

Administrators may alternatively wish to consult the Profiles and *Parameters* available in the *CDN in a Box* environment, as they might be more familiar with them. Furthermore, those Profiles are built with a minimum running Traffic Control system in mind, and thus may be easier to look through. The Profiles and their associated *Parameters* may be found within the [infrastructure/cdn-in-a-box/traffic\\_ops\\_data/profiles/](#) directory.

## The GLOBAL Profile

There is a special Profile of *Type* UNK\_PROFILE that holds global configuration information - its *Name* is “GLOBAL”, its *Type* is UNK\_PROFILE and it is assigned to the special “ALL” *CDN*. The *Parameters* that may be configured on this Profile are laid out in the *Global Profile Parameters* Table.

Table 47: Global Profile Parameters

Name	Config File	Value
tm.url	global	The URL at which this Traffic Ops instance services requests.
tm.reverseproxy	global	Not required. The URL where a caching proxy for configuration files generated by Traffic Ops may be found. Requires a minimum <i>ORT</i> version of 2.1. When configured, <i>ORT</i> will request configuration files via this FQDN, which should be set up as a reverse proxy to the Traffic Ops server(s). The suggested cache lifetime for these files is 3 minutes or less. This setting allows for greater scalability of a CDN maintained by Traffic Ops by caching configuration files of profile and CDN scope, as generating these is a very computationally expensive process.
tm.tool	global	The name of the Traffic Ops tool. Usually “Traffic Ops” - this will appear in the comment headers of generated configuration files.
tm.info	global	This is the “for more information go here” URL, which used to be visible in the “About” page of the now-deprecated Traffic Ops UI.
tm.instance	global	The name of the Traffic Ops instance - typically to distinguish instances when multiple are active.
tm.forwardproxy	global	When collecting stats from Traffic Monitor, Traffic Ops will use this forward proxy instead of the actual Traffic Monitor host. Setting this <i>Parameter</i> can significantly lighten the load on the Traffic Monitor system and it is therefore recommended that this be set on a production system.
use_regexp	global	When this Parameter is present and its <i>Value</i> is exactly “1”, Traffic Ops will separately keep track of <i>cache servers</i> ’ updates and pending <i>Content Invalidation Jobs</i> . This behavior should be enabled by default, and disabling it, while still possible is <b>EXTREMELY DISCOURAGED</b> .
tm_query_online	global	When this Parameter is present, its <i>Value</i> will set which status of Traffic Monitors that Traffic Ops will query for endpoints that require querying Traffic Monitors. If not present, Traffic Ops will default to querying ONLINE Traffic Monitors.
geolocation.polling.url	CR-Conf-fig.json	The location of a geographic IP mapping database for Traffic Router instances to use.
geolocation6.polling.url	CR-Conf-fig.json	The location of a geographic IPv6 mapping database for Traffic Router instances to use.
maxmind.defaultgeographicid	CR-Conf-fig.json	The destination geographic coordinates to use for client location when the geographic IP mapping database returns a default location that matches the country code. This parameter can be specified multiple times with different values to support default overrides for multiple countries. The reason for the name “maxmind” is because the default geographic IP mapping database used by Traffic Control is MaxMind’s GeoIP2 database. The format of this <i>Parameter</i> ’s <i>Value</i> is: <i>Country Code</i> ; <i>Latitude</i> , <i>Longitude</i> , e.g. US;37.751,-97.822
maxmind.regex_revalidate.duration	CR-Conf-fig.json	Gets the maximum duration, in days, for which a <i>Content Invalidation Job</i> may run. Furthermore, while there is no restriction placed on creating multiple <i>Parameters</i> with this <i>Name</i> and <i>Config File</i> - potentially with differing <i>Values</i> - this is <b>EXTREMELY DISCOURAGED</b> as any <i>Parameter</i> that has both that <i>Name</i> and <i>Config File</i> might be used when generating any given <i>regex_revalidate.config</i> file for any given <i>cache server</i> and whenever such <i>Parameters</i> exist, the actual maximum duration for <i>Content Invalidation Jobs</i> is undefined, and CAN and WILL differ from server to server, and configura-

Some of these *Parameters* have the *Config File* value *global*, while others have *CRConfig.json*. This is not a typo, and the distinction is that those that use *global* are typically configuration options relating to Traffic Control as a whole or to Traffic Ops itself, whereas *CRConfig.json* is used by configuration options that are set globally, but pertain mainly to routing and are thus communicated to Traffic Routers through *CDN Snapshots* (which historically were called “CRConfig Snapshots” or simply “the CRConfig”). When a *Parameter* has a *Config File* value that *isn’t* one of *global* or *CRConfig.json*, it refers to the global configuration of said *Config File* across all servers that use it across all CDNs configured in Traffic Control. This can be used to easily apply extremely common configuration to a great many servers in one place.

## Parameters

A *Parameter* is usually a way to set a line in a configuration file that will appear on the servers using *Profiles* that have said Parameter. More generally, though, a Parameter merely describes some kind of configuration for some aspect of some thing. There are many Parameters that *must* exist for Traffic Control to work properly, such as those on *The GLOBAL Profile* or the *Default Profiles*. Some Traffic Control components can be associated with *Profiles* that only have a few allowed (or actually just meaningful - others are ignored and don’t cause problems) but some can have any number of Parameters to describe custom configuration of things of which Traffic Control itself may not even be aware (most notably *cache servers*). For most Parameters, the meaning of each Parameter’s various properties are very heavily tied to the allowed contents of [Apache Traffic Server configuration files](#).

## Properties

When represented in Traffic Portal (in the *Parameters* view) or in *Traffic Ops API* request and/or response payloads, a Parameter has several properties that define it. In some of these contexts, the *Profiles* to which a Parameter is assigned (and/or the integral, unique identifiers thereof) are represented as a property of the Parameter. However, an explanation of this “property” is not provided here, as the *Profiles* section exists for the purpose of explaining those.

## Config File

This (usually) names the configuration file to which the Parameter belongs. Note that it is only the *name of* the file and **not** the *full path* to the file - e.g. `remap.config` not `/opt/trafficserver/etc/trafficserver/remap.config`. To define the full path to any given configuration file, Traffic Ops relies on a reserved *Name* value: “*location*”.

### See also:

This section is only meant to cover the special handling of Parameters assigned to specific Config File values. It is **not** meant to be a primer on Apache Traffic Server configuration files, nor is it intended to be exhaustive of the manners in which said files may be manipulated by Traffic Control. For more information, consult the documentation for [Apache Traffic Server configuration files](#).



Certain Config Files are handled specially by Traffic Ops's configuration file generation. Specifically, the format of the configuration is tailored to be correct when the syntax of a configuration file is known. However, these configuration files **must** have “*location*” Parameters on the *Profile* of servers, or they will not be generated. The Config File values that are special in this way are detailed within this section. When a *Config File* is none of these special values, each Parameter assigned to given server's *Profile* with the same *Config File* value will create a single line in the resulting configuration file (with the possible exception being when the *Name* is “header”)

### 12M\_facts

This legacy file is generated entirely from a *Profile*'s metadata, and cannot be affected by Parameters.

---

**Tip:** This Config File serves an unknown and likely historical purpose, so most users/administrators/developers don't need to worry about it.

---

### 50-ats.rules

Parameters have no meaning when assigned to this Config File (except “*location*”), but it is affected by Parameters that are on the same *Profile* with the Config File `storage.config` - **NOT this Config File**. For each letter in the special “Drive Letters” Parameter, a line will be added of the form `KERNEL=="PrefixLetter", OWNER="ats"` where *Prefix* is the *Value* of the Parameter with the *Name* “Drive Prefix” and the Config File `storage.config` - but with the first instance of `/dev/` removed -, and *Letter* is the drive letter. Also, if the Parameter with the *Name* “RAM Drive Prefix” exists on the same Profile assigned to the server, a line will be inserted for each letter in the special “RAM Drive Letters” Parameter of the form `KERNEL=="PrefixLetter", OWNER="ats"` where *Prefix* is the *Value* of the “RAM Drive Prefix” Parameter - but with the first instance of `/dev/` removed -, and *Letter* is the drive letter.

---

**Tip:** This Config File serves an unknown and likely historical purpose, so most users/administrators/developers don't need to worry about it.

---

### astats.config

This configuration file will be generated with a line for each Parameter with this Config File value on the *cache server*'s *Profile* in the form `Name=Value` where *Name* is the Parameter's *Name* with trailing characters that match `__\d+$` stripped, and *Value* is its *Value*.



## bg\_fetch.config

This configuration file always generates static contents besides the header, and cannot be affected by any Parameters (besides its “*location*” Parameter).

### See also:

For an explanation of the contents of this file, consult [the Background Fetch Apache Traffic Server plugin’s official documentation](#).

## cache.config

This configuration is built entirely from *Delivery Service* configuration, and cannot be affected by Parameters.

### See also:

[The Apache Traffic Server cache.config documentation](#)

## cacheurlanything.config

Config Files that match this pattern - where *anything* is a string of zero or more characters - can only be generated by providing a *location* and their contents will be fully determined by properties of *Delivery Services*.

### See also:

[The official documentation for the Cache URL Apache Traffic Server plugin](#).

Deprecated since version ATCv3.0: This configuration file is only used by Apache Traffic Server version 6.x, whose use is deprecated both by that project and Traffic Control. These Config Files will have no special meaning at some point in the future.

## chkconfig

This actually isn’t a configuration file at all, kind of. Specifically, it is a valid configuration file for the legacy [chkconfig utility](#) - but it is never written to disk on any *cache server*. Though all Traffic Control-supported systems are now using *systemd(8)*, *ORT* still uses *chkconfig*-style configuration to set the status of services on its host system(s). This means that any Parameter with this Config File value should have a *Name* that is the name of a service on the *cache servers* using the *Profile* to which the Parameter is assigned, and it’s *Value* should be a valid *chkconfig* configuration line for that service.

### CRConfig.json

In general, the term “CRConfig” refers to *CDN Snapshots*, which historically were called “CR-Config Snapshots” or simply “the CRConfig”. Parameters with this Config File should be only be on either *The GLOBAL Profile* where they will affect global routing configuration, or on a Traffic Router’s *Profile* where they will affect routing configuration for that Traffic Router only.

#### See also:

For the available configuration Parameters for a Traffic Router Profile, see *The Traffic Router Profile*.

### drop\_qstring.config

This configuration file will be generated with a single line that is exactly: `/([^\?]+) $s://$t/$1n` **unless** a Parameter exists on the *Profile* with this Config File value, and the *Name* “content”. In the latter case, the contents of the file will be exactly the Parameter’s *Value* (with terminating newline appended).

### global

In general, this Config File isn’t actually handled specially by Traffic Ops when generating server configuration files. However, this is the Config File value typically used for Parameters assigned to *The GLOBAL Profile* for truly “global” configuration options, and it is suggested that this precedent be maintained - i.e. don’t create Parameters with this Config File.

### hdr\_rw\_anything.config

Config Files that match this pattern - where *anything* is zero or more characters - are written specially by Traffic Ops to accommodate the *DSCP* setting of *Delivery Services*.

---

**Tip:** The *anything* in those file names is typically a *Delivery Service*’s *xml\_id* - though the inability to affect the file’s contents is utterly independent of whether or not a *Delivery Service* with that *xml\_id* actually exists.

---

#### See also:

For information on the contents of files like this, consult [the Header Rewrite Apache Traffic Server plugin’s documentation](#)

## hosting.config

This configuration file is mainly generated based on the assignments of *cache servers* to *Delivery Services* and the *Cache Group* hierarchy, but there are a couple of Parameter *Names* that can affect it when assigned to this Config File. When a Parameter assigned to the `storage.config` Config File - **NOT this Config File** - with the *Name* “RAM\_Drive\_Prefix” exists, it will cause lines to be generated in this configuration file for each *Delivery Service* that is of one of the *Types* DNS\_LIVE (only if the server is an *Edge-tier cache server*), HTTP\_LIVE (only if the server is an *Edge-tier cache server*), DNS\_LIVE\_NATNL, or HTTP\_LIVE\_NATNL to which the *cache server* to which the *Profile* containing that Parameter belongs is assigned. Specifically, it will cause each of them to use `volume=1` **UNLESS** the Parameter with the *Name* “Drive\_Prefix” associated with Config File `storage.config` - again, **NOT this Config File** - also exists, in which case they will use `volume=2`.

**Caution:** If a Parameter with Config File `storage.config` and *Name* “RAM\_Drive\_Prefix” does *not* exist on a *Profile*, then the *cache servers* using that *Profile* will be incapable of serving traffic for *Delivery Services* of the aforementioned *Types*, even when a “*location*” Parameter exists.

### See also:

For an explanation of the syntax of this configuration file, refer to the [Apache Traffic Server hosting.config documentation](#).

## ip\_allow.config

This configuration file is mostly generated from various server data, but can be affected by a Parameter that has a *Name* of “purge\_allow\_ip”, which will cause the insertion of a line with `src_ip=VALUE action=ip_allow method=ALL` where VALUE is the Parameter’s *Value*. Additionally, Parameters with *Names* like `coalesce_masklen/number_v4/6` cause Traffic Ops to generate coalesced IP ranges in different ways. In the case that `number` was used, the Parameter’s *Value* sets the the maximum number of IP address that may be coalesced into a single range. If `masklen` was used, the lines that are generated are coalesced into CIDR (Classless Inter-Domain Routing) ranges using mask lengths determined by the *Value* of the parameter (using ‘4’ sets the mask length of IPv4 address coalescing while using ‘6’ sets the mask length to use when coalescing IPv6 addresses). This is not recommended, as the default mask lengths allow for maximum coalescence. Furthermore, if two Parameters on the same *Profile* assigned to a server having Config File values of `ip_allow.config` and *Names* that are both “`coalesce_masklen_v4`” but each has a different *Value*, then the actual mask length used to coalesce IPv4 addresses is undefined (but will be one of the two). All forms of the “coalescence Parameters” have this problem.

---

### Implementation Detail

At the time of this writing, coalescence is implemented through the [the NetAddr::IP Perl library](#).

---

### See also:

The Apache Traffic Server `ip_allow.config` documentation explains the syntax and meaning of lines in that file.

## logging.config

This configuration file can only be affected by Parameters with specific *Names*. Specifically, for each Parameter assigned to this Config File on the *Profile* used by the *cache server* with the name `LogFormatN.Name` where N is either the empty string or a natural number on the interval [1,9] the text in *Log Format Snippet* will be inserted. In that snippet, NAME is the *Value* of the Parameter with the *Name* `LogFormatN.Name`, and FORMAT is the *Value* of the Parameter with the *Name* `LogFormatN.Format` for the same value of N<sup>2</sup>.

### 9: Log Format Snippet

```
NAME = format {  
    Format = 'FORMAT '  
}
```

---

**Tip:** The order in which these Parameters are considered is exactly the numerical ordering implied by N (starting with it being empty). However, each section is generated for all values of N before moving on to the next.

---

Furthermore, for a given value of N - as before restricted to either the empty string or a natural number on the interval [1,9] -, if a Parameter exists on the *cache server*'s *Profile* having this Config File value with the *Name* `LogFilterN.Name`, a line of the format `NAME = filter.TYPE.('FILTER')` will be inserted, where NAME is the *Value* of the Parameter with the *Name* `LogFilterN.Name`, TYPE is the *Value* of the Parameter with the *Name* `LogFilterN.Type`, and FILTER is the *Value* of the Parameter with the name `LogFilterN.Filter`<sup>3</sup>.

---

**Note:** When, for a given value of N, a Parameter with the *Name* `LogFilterN.Name` exists, but a Parameter with the *Name* `LogFilterN.Type` does *not* exist, the value of TYPE will be accept.

---

Finally, for a given value of N, if a Parameter exists on the *cache server*'s *Profile* having this Config File value with the *Name* `LogObjectN.Filename`, the text in *Log Object Snippet* will be inserted. In that snippet, TYPE is the *Value* of the Parameter with the *Name* `LogObjectN.Type`

---

<sup>2</sup> This *Value* may safely contain double quotes (") as they will be backslash-escaped in the generated output.

<sup>3</sup> This *Value* may safely contain backslashes (\) and single quotes ('), as they will be backslash-escaped in the generated output.

#### 10: Log Object Snippet

```
log.TYPE {
    Format = FORMAT,
    Filename = 'FILENAME',
```

---

**Note:** When, for a given value of N a Parameter with the *Name* LogObjectN.Filename exists, but a Parameter with the *Name* LogObjectN.Type does *not* exist, the value of TYPE in *Log Object Snippet* will be `ascii`.

---

At this point, if the *Value* of the Parameter with the *Name* LogObjectN.Type is **exactly** `pipe`, a line of the format `Filters = FILTERS` will be inserted where FILTERS is the *Value* of the Parameter with the *Name* LogObjectN.Filters, followed by a line containing only a closing “curly brace” (`}`) - *if and only if said Parameter is not empty*. If, however, the *Value* of the Parameter with the *Name* LogObjectN.Type is **not** exactly `pipe`, then the text in *Log Object (not a “pipe”) Snippet* is inserted.

#### 11: Log Object (not a “pipe”) Snippet

```
RollingEnabled = ROLLING,
RollingIntervalSec = INTERVAL,
RollingOffsetHr = OFFSET,
RollingSizeMb = SIZE
}
```

In this snippet, ROLLING is the *Value* of the Parameter with the *Name* LogObjectN.RollingEnabled, INTERVAL is the *Value* of the Parameter with the *Name* LogObjectN.RollingIntervalSec, OFFSET is the *Value* of the Parameter with the *Name* LogObjectN.RollingOffsetHr, and SIZE is the *Value* of the Parameter with the *Name* LogObjectN.SizeMb - all still having the same value of N, and the Config File value `logging.config`, of course.

**Warning:** The contents of these fields are not validated by Traffic Control - handle with care!

#### See also:

The Apache Traffic Server documentation for the `logging.config` configuration file

### logging.yaml

This is a YAML-format configuration file used by *cache servers* that use Apache Traffic Server version 8 or higher - for lower versions, users/administrators/developers should instead be configuring `logging.config`. This configuration always starts with (after the header) the single line: `format:.` Afterward, for every Parameter assigned to this Config File with a *Name* like `LogFormatN.Name` where N is either the empty string or a natural number on the interval [1,9], the YAML fragment shown in *Log Format Snippet* will be inserted. In this snippet, *NAME* is the *Value* of the Parameter with the *Name* `LogFormatN.Name`, and for the same value of N *FORMAT* is the *Value* of the Parameter with the *Name* `LogFormatN.Format`.

#### 12: Log Format Snippet

```
- name: NAME
  format: 'FORMAT'
```

---

**Tip:** The order in which these Parameters are considered is exactly the numerical ordering implied by N (starting with it being empty). However, each section is generated for all values of N before moving on to the next.

---

After this, a single line containing only `filters:` is inserted. Then, for each Parameter on the *cache server's Profile* with a *Name* like `LogFilterN.Name` where N is either the empty string or a natural number on the interval [1,9], the YAML fragment in *Log Filter Snippet* will be inserted. In that snippet, *NAME* is the *Value* of the Parameter with the *Name* `LogFilterN.Name`, *TYPE* is the *Value* of the Parameter with the *Name* `LogFilterN.Type` for the same value of N, and *FILTER* is the *Value* of the Parameter with the *Name* `LogFilterN.Filter` for the same value of N.

#### 13: Log Filter Snippet

```
- name: NAME
  action: TYPE
  condition: FILTER
```

---

**Note:** When, for a given value of N, a Parameter with the *Name* `LogFilterN.Name` exists, but a Parameter with the *Name* `LogFilterN.Type` does *not* exist, the value of *TYPE* in *Log Filter Snippet* will be accept.

---

At this point, a single line containing only `logs:` is inserted. Finally, for each Parameter on the *cache server's Profile* assigned to this Config File with a *Name* like `LogObjectN.Filename` where N is once again either an empty string or a natural number on the interval [1,9] the YAML fragment in *Log Object Snippet* will be inserted. In this snippet, for a given value of N *TYPE* is the *Value* of the Parameter with the *Name* `LogObjectN.Type`, *FILENAME* is the *Value* of the Parameter with the *Name* `LogObjectN.Filename`, *FORMAT* is the *Value* of the Parameter with the *Name* `LogObjectN.Format`.

#### 14: Log Object Snippet

```
- mode: TYPE
  filename: FILENAME
  format: FORMAT
  ROLLING_OR_FILTERS
```

**Note:** When, for a given value of N a Parameter with the *Name* `LogObjectN.Filename` exists, but a Parameter with the *Name* `LogObjectN.Type` does *not* exist, the value of TYPE in *Log Object Snippet* will be `ascii`.

ROLLING\_OR\_FILTERS will be one of two YAML fragments based on the *Value* of the Parameter with the name `LogObjectN.Type`. In particular, if it is exactly `pipe`, then ROLLING\_OR\_FILTERS will be `filters: [FILTERS]` where FILTERS is the *Value* of the Parameter assigned to this Config File with the *Name* `LogObjectN.Filters` for the same value of N. If, however, the *Value* of the Parameter with the *Name* `LogObjectN.Type` is **not** exactly `pipe`, ROLLING\_OR\_FILTERS will have the format given by *Log Object (not a “pipe”) Snippet*. In that snippet, ROLLING is the *Value* of the Parameter with the *Name* `LogObjectN.RollingEnabled`, INTERVAL is the *Value* of the Parameter with the *Name* `LogObjectN.RollingIntervalSec`, OFFSET is the *Value* of the Parameter with the *Name* `LogObjectN.RollingOffsetHr`, and SIZE is the *Value* of the Parameter with the *Name* `LogObjectN.RollingSizeMb` - all for the same value of N and assigned to the `logging.yaml` Config File, obviously.

#### 15: Log Object (not a “pipe”) Snippet

```
rolling_enabled: ROLLING
rolling_interval_sec: INTERVAL
rolling_offset_hr: OFFSET
rolling_size_mb: SIZE
```

#### See also:

For an explanation of YAML syntax, refer to the [official specification thereof](#). For an explanation of the syntax of a valid Apache Traffic Server `logging.yaml` configuration file, refer to [that project’s dedicated documentation](#).

### logs\_xml.config

This configuration file is somewhat more complex than most Config Files, in that it generates XML document tree segments<sup>1</sup> for each Parameter on the *cache server’s Profile* rather than simply a plain-text line. Specifically, up to ten of the document fragment shown in *LogFormat Snippet* will be inserted, one for each Parameter with this Config File value on the *cache*

<sup>1</sup> The contents of this file are not valid XML, but are rather XML-like so developers writing procedures that will consume and parse it should be aware of this, and note the actual syntax as specified in the [Apache Traffic Server documentation for logs\\_xml.config](#)



*server's Profile* that has a *Name* like `LogFormatN.Name` where N is either the empty string or a natural number on the range [1,9]. In that snippet, the string NAME is actually the *Value* of the Parameter with the *Name* `LogFormatN.Name` FORMAT is the *Value* of the Parameter with the *Name* `LogFormatN.Format`<sup>Page 45, 2</sup>, where again N is either the empty string or a natural number on the interval [1,9] - same-valued N Parameters are associated.

### 16: LogFormat Snippet

```
<LogFormat>
  <Name = "NAME"/>
  <Format = "FORMAT"/>
</LogFormat>
```

---

**Tip:** The order in which these Parameters are considered is exactly the numerical ordering implied by N (starting with it being empty).

---

Furthermore, for a given value of N, if a Parameter exists on the *cache server's Profile* having this Config File value with the *Name* `LogObjectN.Filename`, the document fragment shown in *LogObject Snippet* will be inserted. In that snippet, OBJ\_FORMAT is the *Value* of the Parameter with the *Name* `LogObjectN.Format`, FILENAME is the *Value* of the Parameter with the *Name* `LogObjectN.Filename`, ROLLING is the *Value* of the Parameter with the *Name* `LogObjectN.RollingEnabled`, INTERVAL is the *Value* of the Parameter with the *Name* `LogObjectN.RollingIntervalSec`, OFFSET is the *Value* of the Parameter with the *Name* `LogObjectN.RollingOffsetHr`, SIZE is the *Value* of the Parameter with the *Name* `LogObjectN.RollingSizeMb`, and HEADER is the *Value* of the Parameter with the *Name* `LogObjectN.Header` - all having the same value of N, and the Config File value `logs_xml.config`, of course.

### 17: LogObject Snippet

```
<LogObject>
  <Format = "OBJ_FORMAT"/>
  <Filename = "FILENAME"/>
  <RollingEnabled = ROLLING/>
  <RollingInterval = INTERVAL/>
  <RollingOffsetHr = OFFSET/>
  <RollingSizeMb = SIZE/>
  <Header = "HEADER"/>
</LogObject>
```

**Warning:** The contents of these fields are not validated by Traffic Control - handle with care!

**See also:**

The [Apache Traffic Control documentation on the logs\\_xml.config configuration file](#)



Deprecated since version ATCv3.0: This file is only used by Apache Traffic Server version 6.x. The use of Apache Traffic Server version < 7.1 has been deprecated, and will not be supported in the future. Developers are encouraged to instead configure the *logging.config* configuration file.

## package

This is a special, reserved Config File that isn't a file at all. When a Parameter's Config File is *package*, then its name is interpreted as the name of a package. *ORT* on the server using the *Profile* that has this Parameter will attempt to install a package by that name, interpreting the Parameter's *Value* as a version string if it is not empty. The package manager used will be *yum(8)*, regardless of system (though the Python version of *ORT* will attempt to use the host system's package manager - *yum(8)*, *apt(8)* and *pacman* are supported) but that shouldn't be a problem because only Rocky Linux 8 and CentOS 7 are supported.

The current implementation of *ORT* will expect Parameters to exist on a *cache server*'s *Profile* with the *Names* *astats\_over\_http* and *trafficserver* before being run the first time, as both of these are required for a *cache server* to operate within a Traffic Control CDN. It is possible to install these outside of *ORT* - and indeed even outside of *yum(8)* - but such configuration is not officially supported.

## packages

This Config File is reserved, and is used by *ORT* to pull bulk information about all of the Parameters with Config File values of *package*. It doesn't actually correspond to any configuration file.

## parent.config

This configuration file is generated entirely from *Cache Group* relationships, as well as *Delivery Service* configuration. This file *can* be affected by Parameters on the server's *Profile* if and only if its *Name* is one of the following:

- *algorithm*
- *qstring*
- *psel.qstring\_handling*
- *not\_a\_parent* - unlike the other Parameters listed (which have a 1:1 correspondence with Apache Traffic Server configuration options), this Parameter affects the generation of *parent* relationships between *cache servers*. When a Parameter with this *Name* and Config File exists on a *Profile* used by a *cache server*, it will not be added as a *parent* of any other *cache server*, regardless of *Cache Group* hierarchy. Under ordinary circumstances, there's no real reason for this Parameter to exist.

Additionally, *Delivery Service Profiles* can have special Parameters with the *Name* "mso.parent\_retry" to *Configure Multi-Site Origin*.

### See also:

There are many Parameters with this Config File that only apply on *Delivery Service Profiles*. Those are documented in *their section of the Delivery Service overview page*.

### See also:

To see how the *Values* of these Parameters are interpreted, refer to the *Apache Traffic Server documentation on the parent.config configuration file*

## plugin.config

For each Parameter with this Config File value on the same *Profile*, a line in the resulting configuration file is produced in the format *NAME VALUE* where *NAME* is the Parameter's *Name* with trailing characters matching the regular expression `__\d+$` stripped out and *VALUE* is the Parameter's *Value*.

**Caution:** In order for Parameters for Config Files relating to Apache Traffic Server plugins - e.g. *regex\_revalidate.config* - to have any effect, a Parameter must exist with this Config File value to instruct Apache Traffic Server to load the plugin. Typically, this is more easily achieved by assigning these Parameters to *The GLOBAL Profile* than on a server-by-server basis.

### See also:

The *Apache Traffic server documentation on the plugin.config configuration file* explains what *Value* and *Name* a Parameter should have to be valid.

## rascal.properties

This Config File is meant to be on Parameters assigned to either Traffic Monitor *Profiles* or *cache server Profiles*. Its allowed *Parameter Names* are all configuration options for Traffic Monitor. The *Names* with meaning are as follows.

### See also:

#### *Health Protocol*

**health.polling.format** The *Value* of this Parameter should be the name of a parsing format supported by Traffic Monitor, used to decode statistics when polling for health and statistics. If this Parameter does not exist on a *cache server's Profile*, the default format (*astats*) will be used. The only supported values are

- *astats* parses the statistics output from the *astats\_over\_http* plugin.
- *stats\_over\_http* parses the statistics output from the *stats\_over\_http* plugin.
- *noop* no statistics are parsed; the *cache servers* using this *Value* will always be considered healthy, but statistics will never be gathered for them.

For more information on Traffic Monitor plug-ins that can expand the parsed formats, refer to [Extensions](#).

**health.polling.url** The *Value* of this Parameter sets the URL requested when Traffic Monitor polls cache servers that have this Parameter in their *Profiles*. Specifically, the *Value* is interpreted as a template - in a format reminiscent of variable interpolation in double-quoted strings in Bash -, that offers the following substitutions:

- `${hostname}` Replaced by the *IP Address* of the *cache server* being polled, and **not** its (short) hostname. The IP address used will be its IPv4 service address if it has one, otherwise its IPv6 service address. IPv6 addresses are properly formatted when inserted into the template, so the template need not include “square brackets” ([ and ]) around `${hostname}`s even when they anticipate they will be IPv6 addresses.
- `${interface_name}` Replaced by the name of the network interface that contains the *cache server*’s service address(es). For most cache servers (specifically those using the `stats_over_http` ATS plugin to report their health and statistics) using this in a template won’t be necessary.

If the template doesn’t include a specific port number, the *cache server*’s TCP port will be inserted if the URL uses the HTTP scheme, or its HTTPS Port if the *cache server* uses the the HTTPS scheme.

Table [health.polling.url Value Examples](#) gives some examples of templates, inputs, and outputs.

Table 48: health.polling.url Value Examples

Template	Chosen Service IP	TCP Port	HTTP Port	Interface Name	Output
<code>http://\${hostname}/_astats?inf.name=\${interface_name}</code>	192.0.2.42	8080	8443	eth0	<code>http://192.0.2.42:8080/_astats?inf.name=eth0</code>
<code>https://\${hostname}/_stats</code>	2001:DB8:0:0:1::1	8080	8443	eth0	<code>https://[2001:DB8:0:0:1::1]/_stats</code>
<code>http://\${hostname}:80/custom/stats/path/\${interface_name}</code>	192.0.2.42	8080	8443	eth0	<code>http://192.0.2.42:80/custom/stats/path/eth0</code>

**health.threshold.loadavg** The *Value* of this Parameter sets the “load average” above which the associated *Profile*’s *cache server* will be considered “unhealthy”.

**See also:**

The definition of a “load average” can be found in the documentation for the Linux/Unix command `uptime(1)`.

**Caution:** If more than one Parameter with this *Name* and Config File exist on the same *Profile* with different *Values*, the actual *Value* used by any given Traffic Monitor instance is undefined (though it will be the *Value* of one of those Parameters).

**health.threshold.availableBandwidthInKbps** The *Value* of this Parameter sets the amount of bandwidth (in kilobits per second) that Traffic Control will try to keep available on the *cache server* - for all network interfaces. For example a *Value* of “>1500000” indicates that the *cache server* will be marked “unhealthy” if its available remaining bandwidth across all of the network interfaces used by the caching proxy fall below 1.5Gbps.

**Caution:** If more than one Parameter with this *Name* and Config File exist on the same *Profile* with different *Values*, the actual *Value* used by any given Traffic Monitor instance is undefined (though it will be the *Value* of one of those Parameters).

**history.count** The *Value* of this Parameter sets the maximum number of collected statistics will retain at a time. For example, if this is “30”, then Traffic Monitor will keep up to the past 30 collected statistics runs for the *cache servers* using the *Profile* that has this Parameter. The minimum history size is 1, and if this Parameter’s *Value* is set below that, it will be treated as though it were 1.

**Caution:** This **must** be an integer. What happens when the *Value* of this Parameter is *not* an integer is not known to this author; at a guess, in all likelihood it would be treated as though it were 1 and warnings/errors would be logged by Traffic Monitor and/or Traffic Ops. However, this is not known and setting it improperly is potentially dangerous, so *please ensure it is always an integer*.

### records.config

For each Parameter with this Config File value on the same *Profile*, a line in the resulting configuration file is produced in the format *NAME VALUE* where *NAME* is the Parameter’s *Name* with trailing characters matching the regular expression `__\d+$` stripped out and *VALUE* is the Parameter’s *Value*.

#### See also:

[The Apache Traffic Server records.config documentation](#)

## regex\_remap\_anything.config

Config Files matching this pattern - where *anything* is zero or more characters - are generated entirely from *Delivery Service* configuration, which cannot be affected by any Parameters (except “*location*”).

### See also:

For the syntax of configuration files for the “Regex Remap” plugin, see [the Regex Remap plugin’s official documentation](#). For instructions on how to enable a plugin, consult, the [plugin.config](#) documentation.

## regex\_revalidate.config

This configuration file can only be affected by the special `maxRevalDurationDays`, which is discussed in the *The GLOBAL Profile* section.

### See also:

For the syntax of configuration files for the “Regex Revalidate” plugin, see [the Regex Revalidate plugin’s official documentation](#). For instructions on how to enable a plugin, consult, the [plugin.config](#) documentation.

## remap.config

This configuration file can only be affected by Parameters on a *Profile* assigned to a *Delivery Service*. Then, for every Parameter assigned to that *Profile* that has the Config File value “remap.config” -, a parameter will be added to the line for that *Delivery Service* of the form `@pparam=Value` where Value is the Parameter’s *Value*. Each argument should have its own Parameter. Repeated arguments are allowed, but a warning is issued by *t3c* when processing configuration for cache servers that serve content for the *Delivery Service* with a *Profile* that includes duplicate arguments.

For backwards compatibility, a special case exists for the `cachekey.config` Config File for Parameters on *Delivery Service Profiles* that can also affect this configuration file. This is of the form: `pparam=--Name=Value` where Name is the Parameter’s *Name*, and Value is its *Value*. A warning will be issued by *t3c* when processing configuration for cache servers that serve content for the *Delivery Service* with a *Profile* that uses a Parameter with the Config File `cachekey.config` as well as at least one with the Config File `cachekey.pparam`.

The following plugins have support for adding args with following parameter Config File values.

- `background_fetch.pparam` Note the `--config=bg_fetch.conf` argument is already added to `remap.config` by *t3c*.
- `cachekey.pparam`
- `cache_range_requests.pparam`
- `slice.pparam` Note the `--blocksize=val` plugin parameter is specifiable directly on *Delivery Services* by setting their *Range Slice Request Block Size* property.

- `url_sig.pparam` Note the configuration file for this plugin is already added by *t3c*.

## See also:

For more information about these plugin parameters, refer to the [Apache Traffic Server documentation for the background\\_fetch plugin](#), the [Apache Traffic Server documentation for the cachekey plugin](#), the [Apache Traffic Server documentation for the cache\\_range\\_requests plugin](#), the [Apache Traffic Server documentation for the slice plugin](#), and the [Apache Traffic Server documentation for the url\\_sig plugin](#), respectively.

Deprecated since version ATCv6: `cachekey.config` is deprecated but available for backwards compatibility. `cachekey.config` Parameters will be converted by *t3c* to the “pparam” syntax with `--` added as a prefix to the *Name*. Any “empty” param value (i.e. separator) will add an extra `=` to the key.

Table 49: Equivalent `cachekey.config/cachekey.pparam` entries

<i>Name</i>	Config File	<i>Value</i>	Result
remove-all-params	<code>cachekey.config</code>	<code>true</code>	<code>@pparam=--remove-all-params=true</code>
<code>cachekey.pparam</code>	<code>remap.config</code>	<code>--remove-all-params=true</code>	<code>@pparam=--remove-all-params=true</code>
separator	<code>cachekey.config</code>	(empty value)	<code>@pparam=--separator=</code>
<code>cachekey.pparam</code>	<code>remap.config</code>	<code>--separator=</code>	<code>@pparam=--separator=</code>
<code>cachekey.pparam</code>	<code>cachekey.pparam</code>	<code>o</code>	<code>@pparam=-o</code>

In order to support difficult configurations at MID/LAST, a *Delivery Service* profile parameter is available with parameters `LastRawRemapPre` and `LastRawRemapPost`, config file `remap.config` and Value the raw remap lines. The Value in this parameter will be pre or post pended to the end of `remap.config`.

## See also:

For an explanation of the syntax of this configuration file, refer to the [Apache Traffic Server remap.config documentation](#).

## set\_dscp\_anything.config

Configuration files matching this pattern - where *anything* is a string of zero or more characters is generated entirely from a “*location*” Parameter.

---

**Tip:** *anything* in that Config File name only has meaning if it is a natural number - specifically, one of each value of *DSCP* on every *Delivery Service* to which the *cache server* using the *Profile* on which the Parameter(s) exist(s).

---

## ssl\_multicert.config

This configuration file is generated from the SSL keys of *Delivery Services*, and is unaffected by any Parameters (except “*location*”)

**See also:**

The official [ssl\\_multicert.config](#) documentation

## storage.config

This configuration file can only be affected by a handful of Parameters. If a Parameter with the *Name* “Drive Prefix” exists the generated configuration file will have a line inserted in the format *PREFIXLETTER volume=1* for each letter in the comma-delimited list that is the *Value* of the Parameter on the same *Profile* with the *Name* “Drive Letters”, where *PREFIX* is the *Value* of the Parameter with the *Name* “Drive Prefix”, and *LETTER* is each of the aforementioned letters in turn. Additionally, if a Parameter on the same *Profile* exists with the *Name* “RAM Drive Prefix” then for each letter in the comma-delimited list that is the *Value* of the Parameter on the same *Profile* with the *Name* “RAM Drive Letters”, a line will be generated in the format *PREFIXLETTER volume=i* where *PREFIX* is the *Value* of the Parameter with the *Name* “RAM Drive Prefix”, *LETTER* is each of the aforementioned letters in turn, and *i* is 1 *if and only if* a Parameter does **not** exist on the same *Profile* with the *Name* “Drive Prefix” and is 2 otherwise. Finally, if a Parameter exists on the same *Profile* with the *Name* “SSD Drive Prefix”, then a line is inserted for each letter in the comma-delimited list that is the *Value* of the Parameter on the same *Profile* with the *Name* “SSD Drive Letters” in the format *PREFIXLETTER volume=i* where *PREFIX* is the *Value* of the Parameter with the *Name* “SSD Drive Prefix”, *LETTER* is each of the aforementioned letters in turn, and *i* is 1 *if and only if both* a Parameter with the *Name* “Drive Prefix” and a Parameter with the *Name* “RAM Drive Prefix” *don’t exist on the same Profile*, or 2 if only **one** of them exists, or otherwise 3.

**See also:**

The Apache Traffic Server [storage.config](#) file documentation.

## traffic\_stats.config

This Config File value is only handled specially when the *Profile* to which it is assigned is of the special *TRAFFIC\_STATS Type*. In that case, the *Name* of any Parameters with this Config File is restrained to one of “CacheStats” or “DsStats”. When it is “Cache Stats”, the *Value* is interpreted specially based on whether or not it starts with “ats.”. If it does, then what follows must be the name of one of the [core Apache Traffic Server statistics](#). This signifies to Traffic Stats that it should store that statistic for *cache servers* within Traffic Control. Additionally, the special statistics “bandwidth”, “maxKbps” are supported as *Names* - and in fact it is suggested that they exist in every Traffic Control deployment.

When the Parameter *Name* is “DSSStats”, the allowed *Values* are:

- kbps



- status\_4xx
- status\_5xx
- tps\_2xx
- tps\_3xx
- tps\_4xx
- tps\_5xx
- tps\_total

### See also:

For more information on the statistics gathered by Traffic Stats, see *Traffic Stats Administration*. For information about how these statics are gathered, consult the only known documentation of the “astats\_over\_http” Apache Traffic Server plugin: [traffic\\_server/plugins/astats\\_over\\_http/README.md](#).

## sysctl.config

For each Parameter with this Config File value on the same *Profile*, a line in the resulting configuration file is produced in the format *NAME* = *VALUE* where *NAME* is the Parameter’s *Name* with trailing characters matching the regular expression `__\d+$` stripped out and *VALUE* is the Parameter’s *Value*.

## uri\_signing\_anything.config

Config Files matching this pattern - where *anything* is zero or more characters - are generated entirely from the URI Signing Keys configured on a *Delivery Service* through either the *Traffic Ops API* or the *Delivery Services* view in Traffic Portal.

### See also:

The [draft RFC for uri\\_signing](#) - note, however that the current implementation of `uri_signing` uses Draft 12 of that RFC document, **NOT** the latest.

## url\_sig\_anything.config

Config Files that match this pattern - where *anything* is zero or more characters - are mostly generated using the URL Signature Keys as configured either through the *Traffic Ops API* or the *Delivery Services* view in Traffic Portal. However, if no such keys have been configured, they may be provided by fall-back Parameters. In this case, for each Parameter on assigned to this Config File on the same *Profile* a line is inserted into the resulting configuration file in the format *NAME* = *VALUE* where *NAME* is the Parameter’s *Name* and *VALUE* is the Parameter’s *Value*.



**See also:**

The Apache Trafficserver documentation for the `url_sig` plugin.

## volume.config

This Config File is peculiar in that it depends only on the existence of Parameters, and not each Parameter's actual *Value*. The Parameters that affect the generated configuration file are the Parameters with the *Names* "Drive Prefix", "RAM Drive Prefix", and "SSD Drive Prefix". Each of these Parameters must be assigned to the `storage.config` Config File - **NOT this Config File** - and, of course, be on the same *Profile*. The contents of the generated Config File will be between zero and three lines (excluding headers) where the number of lines is equal to the number of the aforementioned Parameters that actually exist on the same *Profile*. Each line has the format `volume=i scheme=http size=SIZE%` where `i` is a natural number that ranges from 1 to the number of those Parameters that exist. `SIZE` is  $100/N$  - where  $N$  is the number of those special Parameters that exist - truncated to the nearest natural number, e.g.  $100/3 = 33$ .

**See also:**

The Apache Traffic Server `volume.config` file documentation.

## ID

An integral, unique identifier for a Parameter. Note that Parameters must have a unique combination of *Config File*, *Name*, and *Value*, and so those should be used for identifying a unique Parameter whenever possible.

---

## Implementation Detail

If two *Profiles* have been assigned Parameters that have the same values for *Config File*, *Name*, and *Value* then Traffic Ops actually only stores one Parameter object and merely *links* it to both *Profiles*. This can be seen by inspecting the Parameters' IDs, as they will be the same. There are many cases where a user or developer must rely on this implementation detail, but both are encouraged to do so only when absolutely necessary.

---

## Name

The Name of a Parameter has different meanings depending on the type of any and all *Profiles* to which it is assigned, as well as the *Config File* to which the Parameter belongs, but most generally it is used in Apache Traffic Server configuration files as the name of a configuration option in a name/value pair. Traffic Ops interprets the Name and *Value* of a Parameter in intelligent ways depending on the type of object to which the *Profile* using the Parameter is assigned. For example, if *Config File* is `records.config` and the Parameter's *Profile* is assigned to a *cache server*, then a single line is placed in the configuration file specified by *Config File*, and that line will have the contents `Name Value`. However, if the *Config File* of the Parameter is something without special meaning to Traffic Ops e.g. "foo", then a line containing **only** the

Parameter's *Value* would be inserted into that file (presuming it also has a Parameter with a Name of "*location*" and a *Config File* of "foo"). Additionally, there are a few Names that are treated specially by Traffic Control.

**location** The *Value* of this Parameter is to be interpreted as a path under which the configuration file specified by *Config File* shall be found (or written, if not found). Any configuration file that is to exist on a server must have an associated "location" Parameter, even if the contents of the file cannot be affected by Parameters.

**Caution:** If a single *Profile* has multiple "location" Parameters for the same *Config File* with different *Values*, the actual location of the generated configuration file is undefined (but will be one of those Parameters' *Values*).

**header** If the *Profile* containing this Parameter is assigned to a server, **and** if the *Config File* is not one of the special values that Traffic Ops uses to determine special syntax formatting, then the *Value* of this Parameter will be used instead of the typical Traffic Ops header - *unless* it is the special string "none", in which case no header will be inserted at all.

**Caution:** If a single *Profile* has multiple "header" Parameters for the same *Config File* with different *Values*, the actual header is undefined (but will be one of those Parameters' *Values*).

**refetch\_enabled** When a Parameter by this Name exists, and has the *Config File* value of exactly "global", then its *Value* may be used by Traffic Ops to decide whether or not the "REFETCH" *Invalidation Type* of *Content Invalidation Jobs* are allowed to be created. The *Value* "true" (case-insensitive) indicates that such *Content Invalidation Jobs* should be allowed, while all other *Values* indicate they should not.

---

**Note:** Any leading or trailing whitespace in the *Value* of these Parameters is ignored.

---

**Caution:** There is no limit to the number of these Parameters that may exist, and no association to any existing *Profiles* is considered when choosing which Parameter to use. If more than one Parameter with the Name `refetch_enabled` exists with the *Config File* "global", then the actual *Value* used to determine if "REFETCH" *Invalidation Type* of *Content Invalidation Jobs* are allowed to be created is undefined (but will be the *Value* of one of said Parameters). In particular, there is **no special handling** when any of these Parameters is assigned to *The GLOBAL Profile*, and being thusly assigned **in no way means that the assigned Parameter will have any kind of priority over others that collide with it in Name and Config File**.

## Secure

When this is ‘true’, a user requesting to see this Parameter will see the value `*****` instead of its actual value if the user’s permission *Role* isn’t ‘admin’.

## Value

In general, a Parameter’s *Value* can be anything, and in the vast majority of cases the Value is *in no way validated by Traffic Control*. Usually, though, the Value has a special meaning depending on the values of the Parameter’s *Config File* and/or *Name*.

## 2.1.7 Roles

A *Role* is a collection of Permissions that may be associated with zero or more users.

### See also:

The [github.com/apache/trafficcontrol/lib/go-tc.RoleV40](https://github.com/apache/trafficcontrol/lib/go-tc.RoleV40) Go structure documentation.

The model for a Role in the most recent version of the *Traffic Ops API* is given in *Role as a Typescript Interface* as a Typescript interface.

18: Role as a Typescript Interface

```
interface Role {
  description: string;
  lastUpdated?: Date; //RFC3339 timestamp string - only in responses
  name: string;
  permissions: Array<string>;
}
```

## Description

The description of a Role is some human-readable text that describes what the Role is for.

Changed in version ATCv6.1.0: In versions of ATC earlier than 6.1.0, this property was allowed to be null or blank - this is no longer the case.

## Last Updated

The date and time at which the Role was last modified.

## Name

A unique name for the Role. The only Role Name that is treated specially is *The Admin Role*.

Changed in version ATCv6.1.0: Though it was undocumented, there used to be many Role names that were treated specially by ATC in versions earlier than 6.1.0, including (but not limited to), “disallowed”, “operations”, and “steering”.

## The Admin Role

A Role with a Name that is exactly “admin” is special in that it is always treated as having all *Permissions*, regardless of what *Permissions* are actually assigned to it. For this reason, the “admin” Role may never be deleted or modified.

## Permissions

A Role’s *Permissions* is a set of the Permissions afforded to the Role that define the *Traffic Ops API* interactions it is allowed to have. Usually - but not always - this maps directly to some HTTP method of some *Traffic Ops API* endpoint.

Table 50: Aliases

Name	Use(s)	Type(s)
Capabili- ties	Legacy <i>Traffic Ops API</i> versions’ requests and re- sponses	unchanged (array/set of strings)

## 2.1.8 Traffic Monitor

Traffic Monitor is an HTTP service that monitors the *cache servers* in a CDN for a variety of metrics. These metrics are for use in determining the overall “health” of a given *cache server* and the related *Delivery Services*. A given CDN can operate a number of Traffic Monitors, from a number of geographically diverse locations, to prevent false positives caused by network problems at a given site. Traffic Monitors operate independently, but use the state of other Traffic Monitors in conjunction with their own state to provide a consistent view of CDN *cache server* health to downstream applications such as *Traffic Router*. *Health Protocol* governs the *cache server* and *Delivery Service* availability. Traffic Monitor provides a view into CDN health using several RESTful JSON endpoints, which are consumed by other Traffic Monitors and downstream components such as *Traffic Router*. Traffic Monitor is also responsible for serving the overall CDN configuration to *Traffic Router*, which ensures that the configuration of these two critical components remain synchronized as operational and health related changes propagate through the CDN.

## Cache Monitoring

Traffic Monitor polls all *cache servers* configured with a status of REPORTED or ADMIN\_DOWN at an interval specified as a configuration parameter in *Traffic Ops*. If the *cache server* is set to ADMIN\_DOWN it is marked as unavailable but still polled for availability and statistics. If the *cache server* is explicitly configured with a status of ONLINE or OFFLINE, it is not polled by Traffic Monitor and presented to *Traffic Router* as configured, regardless of actual availability. Traffic Monitor makes HTTP requests at regular intervals to a special URL on each Edge-tier *cache server* and consumes the JSON output. The special URL is served by a plugin running on the ATS *cache servers* called “astats”, which is restricted to Traffic Monitor only. The astats plugin provides insight into application and system performance, such as:

- Throughput (e.g. bytes in, bytes out, etc).
- Transactions (e.g. number of 2xx, 3xx, 4xx responses, etc).
- Connections (e.g. from clients, to parents, origins, etc).
- Cache performance (e.g.: hits, misses, refreshes, etc).
- Storage performance (e.g.: writes, reads, frags, directories, etc).
- System performance (e.g: load average, network interface throughput, etc).

Many of the application-level statistics are available at the global or aggregate level, some at the *Delivery Service* level. Traffic Monitor uses the system-level performance to determine the overall health of the *cache server* by evaluating network throughput and load against values configured in *Traffic Ops*. Traffic Monitor also uses throughput and transaction statistics at the *Delivery Service* level to determine *Delivery Service* health. If astats is unavailable due to a network-related issue or the system statistics have exceeded the configured thresholds, Traffic Monitor will mark the *cache server* as unavailable. If the *Delivery Service* statistics exceed the configured thresholds, the *Delivery Service* is marked as unavailable, and *Traffic Router* will start sending clients to the overflow destinations for that *Delivery Service*, but the *cache server* remains available to serve other content.

### See also:

For more information on ATS statistics, see the [ATS documentation](#)

## Health Protocol

### Optimistic Health Protocol

Redundant Traffic Monitor servers operate independently from each other but take the state of other Traffic Monitors into account when asked for health state information. In *Cache Monitoring*, the behavior of a single Traffic Monitor instance is described. The *Health Protocol* adds another dimension to the health state of the CDN by merging the states of all Traffic Monitors into one, and then taking the *optimistic* approach when dealing with a *cache server* or *Delivery Service* that might have been marked as unavailable by this particular instance or a peer instance of Traffic Monitor. Upon startup or configuration change in *Traffic Ops*, in addition to *cache servers*, Traffic Monitor begins polling its peer Traffic Monitors whose state is set to ONLINE.

Each ONLINE Traffic Monitor polls all of its peers at a configurable interval and saves the peer's state for later use. When polling its peers, Traffic Monitor asks for the raw health state from each respective peer, which is strictly that instance's view of the CDN's health. When any ONLINE Traffic Monitor is asked for CDN health by a downstream component, such as *Traffic Router*, the component gets the Health Protocol-influenced version of CDN health (non-raw view). In operation of the Health Protocol, Traffic Monitor takes all health states from all peers, including the locally known health state, and serves an optimistic outlook to the requesting client. This means that, for example, if three of the four Traffic Monitors see a given *cache server* or *Delivery Service* as exceeding its thresholds and unavailable, it is still considered available. Only if all Traffic Monitors agree that the given object is unavailable is that state propagated to downstream components. This optimistic approach to the Health Protocol is counter to the “fail fast” philosophy, but serves well for large networks with complicated geography and/or routing. The optimistic Health Protocol allows network failures or latency to occur without affecting overall traffic routing, as Traffic Monitors can and do have a different view of the network when deployed in geographically diverse locations.

### Optimistic Quorum

In order to prevent split-brain monitoring scenarios, a minimum of three Traffic Monitors are required to properly monitor a given CDN and the optimistic quorum feature should be enabled. If three or more Traffic Monitors are set to ONLINE, the optimistic quorum can be employed by setting the `peer_optimistic_quorum_min` property in `traffic_monitor.cfg` to a value greater than zero. This value represents the minimum number of peers that must be available in order to participate in the *Optimistic Health Protocol*. If Traffic Monitor detects that the number of available peers is less than this number, Traffic Monitor withdraws itself from participation in the health protocol by serving 503s for cache health state calls until connectivity is restored.

The optimistic quorum prevents invalid state propagation caused by a Traffic Monitor losing connectivity to the network and consequently marking all peers and caches as unavailable. When connectivity is restored, a race between peering recovery and polling from Traffic Routers begins. If Traffic Router were to poll a Traffic Monitor that has no available peers and optimistic quorum is not enabled or cannot be used (i.e.: too few Traffic Monitors), the Traffic Monitor will serve its local state only until peer connectivity is restored. If Traffic Router polls the Traffic Monitor when in this state, that is, prior to regaining peering, negative cache states caused by the lack of connectivity would be consumed and directly impact which caches are available for consideration for routing, until the Traffic Router polls a Traffic Monitor that has good state, or peering is restored. For this reason, it is recommended to run a minimum of three Traffic Monitors, with `peer_optimistic_quorum_min` set to a value of 1 or greater. Note that this value cannot exceed the number of peers of any given Traffic Monitor; that is, a value of 2 is the maximum value that can be used when three Traffic Monitors are in use. If this number exceeds the number of peers, the Traffic Monitor will always serve 503s and an error will be logged.



## Protocol Engagement

Short polling intervals of both the *cache servers* and Traffic Monitor peers help to reduce customer impact of outages. It is not uncommon for a *cache server* to be marked unavailable by Traffic Monitor - in fact, it is business as usual for many CDNs. Should a widely requested video asset cause a single *cache server* to get close to its interface capacity, the Health Protocol will “kick in,” and Traffic Monitor marks the *cache server* as unavailable. New clients want to see the same asset, and now *Traffic Router* will send these customers to another *cache server* in the same *Cache Group*. The load is now shared between the two *cache servers*. As clients finish watching the asset on the overloaded *cache server*, it will drop below the threshold and gets marked available again, and new clients will begin to be directed to it once more. It is less common for a *Delivery Service* to be marked unavailable by Traffic Monitor. The *Delivery Service* thresholds are usually used for overflow situations at extreme peaks to protect other *Delivery Services* in the CDN from being impacted.

### 2.1.9 Traffic Ops

Traffic Ops is the tool for administration (configuration and monitoring) of all components in a Traffic Control CDN. *Traffic Portal* uses the *Traffic Ops API* to manage servers, *Cache Groups*, *Delivery Services*, etc. In many cases, a configuration change requires propagation to several, or even all, *cache servers* and only explicitly after or before the same change propagates to *Traffic Router*. Traffic Ops takes care of this required consistency between the different components and their configuration.

Traffic Ops uses a *PostgreSQL* database to store the configuration information, and a combination of the *Mojolicious framework* and *Go* to provide the *Traffic Ops API*. Not all configuration data is in this database however; for sensitive data like private SSL keys or token-based authentication shared secrets, *Traffic Vault* is used as a separate, key/value store, allowing administrators to harden the *Traffic Vault* server better from a security perspective (i.e only allow Traffic Ops to access it, verifying authenticity with a certificate). The Traffic Ops server, by design, needs to be accessible from all the other servers in the Traffic Control CDN.

Traffic Ops generates all the application-specific configuration files for the *cache servers* and other servers. The *cache servers* and other servers check in with Traffic Ops at a regular interval to see if updated configuration files require application. On *cache servers* this is done by the *ORT* script.

Traffic Ops also runs a collection of periodic checks to determine the operating state of the *cache servers*. These periodic checks are customizable by the Traffic Ops administrative user using *Traffic Ops Extensions*.

### Traffic Ops Extension

Traffic Ops Extensions are a way to enhance the basic functionality of Traffic Ops in a custom manner. There are two types of extensions:

*Check Extensions* Allow you to add custom checks to the *Monitor* → *Cache Checks* view in *Traffic Portal*.

### 2.1.10 Traffic Portal

Traffic Portal is an [AngularJS 1.x](#) client served from a [Node.js](#) web server designed to consume the *Traffic Ops API*. It is the official replacement for the legacy *Traffic Ops* UI.

#### Features

- CDN Monitoring
- CDN Administration
- *Delivery Service* Configuration
- *cache server* Maintenance

#### See also:

See *Traffic Portal - Using* for an overview of the Traffic Portal UI.

### 2.1.11 Traffic Router

Traffic Router's function is to send clients to the most optimal *cache server*. 'Optimal' in this case is based on a number of factors:

- Distance between the *cache server* and the client (not necessarily measured in physical distance, but quite often in layer 3 network hops). Less network distance between the client and *cache server* yields better performance and lower network load. Traffic Router helps clients connect to the best-performing *cache server* for their location at the lowest network cost.
- Availability of *cache servers* and the system processing/network load on the *cache servers*. A common issue in Internet and television distribution scenarios is having many clients attempting to retrieve the same content at the same time. Traffic Router helps clients route around overloaded or purposely disabled *cache servers*.
- Availability of content on a particular *cache server*. Reusing of content through "cache hits" is the most important performance gain a CDN can offer. Traffic Router sends clients to the *cache server* that is most likely to already have the desired content.

Traffic routing options are often configured at the *Delivery Service* level.



## DNS Content Routing

For a DNS *Delivery Service* the client might receive a URL such as `http://video.demo1.mycdn.ciab.test/`. When the LDNS is resolving this `video.demo1.mycdn.ciab.test` hostname to an IP address, it ends at Traffic Router because it is the authoritative DNS server for `mycdn.ciab.test` and the domains below it, and subsequently responds with a list of IP addresses from the eligible *cache servers* based on the location of the LDNS. When responding, Traffic Router does not know the actual client IP address or the path that the client is going to request. The decision on what *cache server* IP address (or list of *cache server* IP addresses) to return is solely based on the location of the LDNS and the health of the *cache servers*. The client then connects to port 80 (HTTP) or port 443 (HTTPS) on the *cache server*, and sends the `Host: video.demo1.mycdn.ciab.test` header. The configuration of the *cache server* includes the “remap rule” `http://video.demo1.mycdn.ciab.test http://origin.infra.ciab.test` to map the routed name to an *Origin* hostname.

## HTTP Content Routing

For an HTTP *Delivery Service* the client might receive a URL such as `http://video.demo1.mycdn.ciab.test/`. The LDNS resolves this `video.demo1.mycdn.ciab.test` to an IP address, but in this case Traffic Router returns its own IP address. The client opens a connection to port 80 (HTTP) or port 443 (HTTPS) on the Traffic Router’s IP address, and sends its request.

### 19: Example Client Request to Traffic Router

```
GET / HTTP/1.1
Host: video.demo1.mycdn.ciab.test
Accept: */*
```

Traffic Router uses an HTTP 302 Found response to redirect the client to the best *cache server*.

### 20: Traffic Router Redirect to Edge-tier *cache server*

```
HTTP/1.1 302 Found
Location: http://edge.demo1.mycdn.ciab.test/
Content-Length: 0
Date: Tue, 13 Jan 2015 20:01:41 GMT
```

In this case Traffic Router has access to more information when selecting a *cache server* because it has a full HTTP request instead of just a hostname. Traffic Router can be configured to select a *cache server* based on any of the following parts of the HTTP request:

- The client’s IP address.
- The URL the client is requesting.
- All HTTP/1.1 headers.

The client follows the redirect and performs a DNS request for the IP address for `edge.demo1.mycdn.ciab.test`, and normal HTTP steps follow, except the sending of the `Host:` header when connected to the cache is `Host: edge.demo1.mycdn.ciab.test`, and the configura-

tion of the *cache server* includes the “remap rule” (e.g. `http://edge.demo1.mycdn.ciab.test` `http://origin.infra.ciab.test`). Traffic Router sends all requests for the same path in a *Delivery Service* to the same *cache server* in a *Cache Group* using consistent hashing, in this case all *cache servers* in a *Cache Group* are not carrying the same content, and there is a much larger combined cache in the *Cache Group*. In many cases DNS content routing is the best possible option, especially in cases where the client is receiving small objects from the CDN like images and web pages. Traffic Router is redundant and horizontally scalable by adding more instances into the DNS hierarchy using NS records.

### 2.1.12 Traffic Stats

*Traffic Stats* is a program written in [Go](#) that is used to acquire and store statistics about CDNs controlled by Traffic Control. Traffic Stats mines metrics from the *Traffic Monitor APIs* and stores the data in [InfluxDB](#) or [Kafka](#). Data is typically stored in InfluxDB on a short-term basis (30 days or less), and Kafka data is available for consumption based on retention period. The data from InfluxDB is then used to drive graphs created by [Grafana](#) - which are linked to from *Traffic Portal* - as well as provide data exposed through the *Traffic Ops API*. Traffic Stats performs two functions:

- Gathers statistics for Edge-tier *cache servers* and *Delivery Services* at a configurable interval (10 second default) from the *Traffic Monitor APIs* and stores the data in InfluxDB or Kafka
- Summarizes all of the statistics once a day (around midnight UTC) and creates a daily report containing the Max GBPS (Gigabits per second) Served and the Total Bytes Served.

Statistics are stored in three different databases:

- `cache_stats`: Stores data gathered from edge-tier *cache servers*. The [measurements](#) stored by `cache_stats` are:
  - `bandwidth`
  - `maxKbps`
  - `client_connections` (ats.proxy.process.http.current\_client\_connections).

Cache Data is stored with [tags](#) for hostname, *Cache Group*, and CDN. Data can be queried using tags.

- `deliveryservice_stats`: Stores data for *Delivery Services*. The measurements stored by `deliveryservice_stats` are:
  - `kbps`
  - `status_4xx`
  - `status_5xx`
  - `tps_2xx`
  - `tps_3xx`
  - `tps_4xx`

- tps\_5xx
- tps\_total

*Delivery Service* statistics are stored with tags for *Cache Group*, CDN, and *Delivery Service xml\_id*.

- **daily\_stats**: Stores summary data for daily activities. The statistics that are currently summarized are:
  - Max Bandwidth
  - Bytes Served

Daily stats are stored by CDN.

When Kafka is enabled, Cache and Delivery Service statistics are sent through JSON format with optional TLS authentication.

Traffic Stats does not influence overall CDN operation, but is required with InfluxDB enabled in order to display charts in *Traffic Portal*.

### 2.1.13 Traffic Vault

Traffic Vault is a data store used for storing the following types of sensitive information:

- SSL Certificates
  - Private Key
  - Certificate
  - CSR (Certificate Signing Request)
- DNSSEC (DNS Security Extensions) Keys
  - Key Signing Key
    - \* private key
    - \* public key
  - Zone Signing Key
    - \* private key
    - \* public key
- URL Signing Keys

As the name suggests, Traffic Vault is meant to be a “vault” of private keys that only certain users are allowed to access. In order to create, add, and retrieve keys a user must have administrative privileges. Keys can be created via the *Traffic Portal* UI, but they can only be retrieved via the *Traffic Ops API*. Currently, the supported data stores used by Traffic Vault are PostgreSQL and Riak (deprecated). Support for Riak may be removed in a future release.



## ADMINISTRATOR'S GUIDE

How to deploy and manage a Traffic Control CDN.

### 3.1 Administrator's Guide

Traffic Control is distributed in source form for the developer, but also as a binary package. This guide details how to install and configure a Traffic Control CDN using the binary packages, as well as how to perform common operations running a CDN.

When installing a complete CDN from scratch, a sample recommended order is:

1. Traffic Ops DB (PostgreSQL)
2. [InfluxDB](#) [Optional]
3. Traffic Vault (PostgreSQL or Riak) [Optional]
4. Fake Origin [Optional]
5. Traffic Ops
6. Traffic Portal
7. Initial Traffic Ops Dataset Setup [if applicable]
8. Traffic Monitor
9. Apache Traffic Server Mid-Tier Caches
10. Apache Traffic Server Edge-Tier Caches
11. Grove [Optional]
12. Traffic Router
13. [InfluxDB-relay](#) [Optional]
14. Traffic Stats [Optional]

Once everything is installed, you will need to configure the servers to talk to each other. You will also need Origin server(s), from which the Mid-Tier Cache(s) will obtain content. An Origin server is simply an HTTP(S) server which serves the content you wish to cache on the CDN.

### 3.1.1 Traffic Ops

Traffic Ops is quite possibly the single most complex and most important Traffic Control component. It has many different configuration options that affect a wide range of other components and their interactions.

#### Installing

#### System Requirements

The user must have the following for a successful minimal install:

- CentOS 7 or later
- Two machines - physical or virtual -, each with at least two (v)CPUs, 4GB of RAM, and 20 GB of disk space
- Access to CentOS Base and EPEL *yum(8)* repositories
- Access to [The Comprehensive Perl Archive Network \(CPAN\)](#)

#### Guide

1. Install PostgreSQL Database. For a production install it is best to install PostgreSQL on its own server/virtual machine.

**See also:**

For more information on installing PostgreSQL, see [their documentation](#).

21: Example PostgreSQL Install Procedure

```
yum update -y
yum install -y https://download.postgresql.org/pub/repos/
↳yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.
↳rpm
yum install -y postgresql13-server
su - postgres -c '/usr/pgsql-13/bin/initdb -A md5 -W' #-W
↳forces the user to provide a superuser (postgres)
↳password
```

2. Edit `/var/lib/pgsql/13/data/pg_hba.conf` to allow the Traffic Ops instance to access the PostgreSQL server. For example, if the IP address of the machine to be used as the Traffic Ops host is `192.0.2.1` add the line `host all all 192.0.2.1/32 md5` to the appropriate section of this file.
3. Edit the `/var/lib/pgsql/13/data/postgresql.conf` file to add the appropriate `listen_addresses` or `listen_addresses = '*'`, set `timezone = 'UTC'`, and start the database

22: Starting PostgreSQL with `systemd`(1)

```
systemctl enable postgresql-13
systemctl start postgresql-13
systemctl status postgresql-13 # Prints the status of the
↳PostgreSQL service, to prove it's running
```

4. Build a `traffic_ops-version string.rpm` file using the instructions under the [Building Traffic Control](#) page - or download a pre-built release from the [Apache Continuous Integration server](#).
5. Install a PostgreSQL client on the Traffic Ops host

23: Installing PostgreSQL Client from a Hosted Source

```
yum install -y https://download.postgresql.org/pub/repos/
↳yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.
↳rpm
```

6. Install the Traffic Ops RPM. The Traffic Ops RPM file should have been built in an earlier step.

24: Installing a Generated Traffic Ops RPM

```
yum install -y ./dist/traffic_ops-3.0.0-xxxx.yyyyyyy.el7.
↳x86_64.rpm
```

---

**Note:** This will install the PostgreSQL client, `psql` as a dependency.

---

7. Login to the Database from the Traffic Ops machine. At this point you should be able to login from the Traffic Ops (hostname `to` in the example) host to the PostgreSQL (hostname `pg` in the example) host

25: Example Login to Traffic Ops Database from Traffic Ops Server

```
to-# psql -h pg -U postgres
Password for user postgres:
psql (13.2)
Type "help" for help.

postgres=#
```

8. Create the user and database. By default, Traffic Ops will expect to connect as the `traffic_ops` user to the `traffic_ops` database.

## 26: Creating the Traffic Ops User and Database

```
to-# psql -U postgres -h pg -c "CREATE USER traffic_ops_
↳WITH ENCRYPTED PASSWORD 'tcr0cks';"
Password for user postgres:
CREATE ROLE
to-# createdb traffic_ops --owner traffic_ops -U postgres -
↳h pg
Password:
to-#
```

9. Now, run the following command as the root user (or with *sudo(8)*): `/opt/traffic_ops/install/bin/postinstall`. Some additional files will be installed, and then it will proceed with the next phase of the install, where it will ask you about the local environment for your CDN. Please make sure you remember all your answers and verify that the database answers match the information previously used to create the database.

## 27: Example Output

```
to-# /opt/traffic_ops/install/bin/postinstall
...

===== /opt/traffic_ops/app/conf/production/database.
↳conf=====
Database type [Pg]:
Database type: Pg
Database name [traffic_ops]:
Database name: traffic_ops
Database server hostname IP or FQDN [localhost]: pg
Database server hostname IP or FQDN: pg
Database port number [5432]:
Database port number: 5432
Traffic Ops database user [traffic_ops]:
Traffic Ops database user: traffic_ops
Password for Traffic Ops database user:
Re-Enter Password for Traffic Ops database user:
Writing json to /opt/traffic_ops/app/conf/production/
↳database.conf
Database configuration has been saved
===== /opt/traffic_ops/app/db/dbconf.yml=====
Database server root (admin) user [postgres]:
Database server root (admin) user: postgres
Password for database server admin:
Re-Enter Password for database server admin:
===== /opt/traffic_ops/app/conf/cdn.conf=====
Generate a new secret? [yes]:
```

(continues on next page)



(continued from previous page)

```

Generate a new secret?: yes
Number of secrets to keep? [10]:
Number of secrets to keep?: 10
Not setting up ldap
===== /opt/traffic_ops/install/data/json/users.
↪ json=====
Administration username for Traffic Ops [admin]:
Administration username for Traffic Ops: admin
Password for the admin user:
Re-Enter Password for the admin user:
Writing json to /opt/traffic_ops/install/data/json/users.
↪ json
===== /opt/traffic_ops/install/data/json/openssl_
↪ configuration.json=====
Do you want to generate a certificate? [yes]:
Country Name (2 letter code): US
State or Province Name (full name): CO
Locality Name (eg, city): Denver
Organization Name (eg, company): Super CDN, Inc
Organizational Unit Name (eg, section):
Common Name (eg, your name or your server's hostname):
RSA Passphrase:
Re-Enter RSA Passphrase:
===== /opt/traffic_ops/install/data/json/profiles.
↪ json=====
Traffic Ops url [https://localhost]:
Traffic Ops url: https://localhost
Human-readable CDN Name. (No whitespace, please)↵
↪ [kabletown_cdn]: blue_cdn
Human-readable CDN Name. (No whitespace, please): blue_cdn
DNS sub-domain for which your CDN is authoritative [cdn1.
↪ kabletown.net]: blue-cdn.supercdn.net
DNS sub-domain for which your CDN is authoritative: blue-
↪ cdn.supercdn.net
Writing json to /opt/traffic_ops/install/data/json/
↪ profiles.json

... much SQL output skipped

Starting Traffic Ops
Restarting traffic_ops (via systemctl):
↪ [ OK ]
Waiting for Traffic Ops to restart
Success! Postinstall complete.

```

Table 51: Explanation of the information that needs to be provided:

Field	Description
Database type	This requests the type of database to be used. Answer with the default - 'Pg' to indicate a PostgreSQL database.
Database name	The name of the database Traffic Ops uses to store the configuration information.
Database server host-name IP or FQDN	The hostname of the database server (pg in the example).
Database port number	The database port number. The default value, 5432, should be correct unless you changed it during the setup.
Traffic Ops database user	The username Traffic Ops will use to read/write from the database.
Password for Traffic Ops	The password for the database user that Traffic Ops uses.
Database server root (admin) user name	Privileged database user that has permission to create the database and user for Traffic Ops...
Database server root (admin) user password	The password for the privileged database user.
Traffic Ops URL	The URL to connect to this instance of Traffic Ops, usually <code>https://Traffic Ops host</code>
Human-readable CDN Name	The name of the first CDN which Traffic Ops will be manage.
DNS sub-domain for which your CDN is authoritative	The DNS domain that will be delegated to this Traffic Control CDN.
Administration user-name for Traffic Ops	The Administration (highest privilege) Traffic Ops user to create. Use this user to login for the first time and create other users.
Password for the admin user	The password for the administrative Traffic Ops user.

Deprecated since version ATCv6: The postinstall script is now written in Python. If you run into issues with the postinstall script, you are encouraged to file an issue at <https://github.com/apache/trafficcontrol/issues/new/choose>. The original Perl postinstall script is deprecated and will be removed in a future ATC release. To use the deprecated version anyway, run `/opt/traffic_ops/install/bin/_postinstall.pl` directly instead of `/opt/traffic_ops/install/bin/postinstall`.

The postinstall script can also be run non-interactively using `traffic_ops/install/bin/input.json`

#### 28: Postinstall in Automatic (-a) mode

```
/opt/traffic_ops/install/bin/postinstall -a --cfile /opt/traffic_
↪ops/install/bin/input.json
```

Deprecated since version ATCv6: Once the Perl script is removed, the values in `input.json` for the "hidden" properties will be changed from "1" and "0" to `true` and `false`.

## Upgrading

To upgrade from older Traffic Ops versions, stop the service, use `yum(8)` to upgrade to the latest available Traffic Ops package, and use the [admin](#) tool to perform the database upgrade.

---

**Tip:** In order to upgrade to the latest version of Traffic Ops, please be sure that you have first upgraded to the latest available minor or patch version of your current release. For example, if your current Traffic Ops version is 3.0.0 and version 3.1.0 is available, you must first upgrade to 3.1.0 before proceeding to upgrade to 4.0.0. (Specifically, this means running all migrations, `traffic_ops/app/db/seeds.sql`, and `traffic_ops/app/db/patches.sql` for the latest of your current major version - which should be handled by the **admin** tool). The latest migration available before the release of 4.0.0 (pending at the time of this writing) was `traffic_ops/app/db/migrations/20180814000625_remove_capabilities_for_reseed.sql`, so be sure that migrations up to this point have been run before attempting to upgrade Traffic Ops.

---

#### See also:

[app/db/admin](#) for more details about **admin**.

### 29: Sample Script for Upgrading Traffic Ops

```
systemctl stop traffic_ops
yum upgrade traffic_ops
pushd /opt/traffic_ops/app/
./db/admin --env production upgrade
./db/admin --env production --trafficvault upgrade
popd
```

After this completes, see [Guide](#) for instructions on running the **postinstall** script. Once the **postinstall** script, has finished, run the following command as the root user (or with *sudo(8)*): `systemctl start traffic_ops` to start the service.

## Upgrading to 6.0

As of Apache Traffic Control 6.0, Traffic Ops supports PostgreSQL version 13.2. In order to migrate from the prior PostgreSQL version 9.6, it is recommended to use the [pg\\_upgrade](#) tool.

## Running

While this section contains instructions for running Traffic Ops manually, the only truly supported method is via *systemd(8)*, e.g. `systemctl start traffic_ops` (this method starts the program properly and uses its default configuration file locations).

### traffic\_ops\_golang

```
traffic_ops_golang [--version] [--plugins] [--api-routes] --cfg
CONFIG_PATH --dbcfg DB_CONFIG_PATH [--riakcfg RIAK_CONFIG_PATH]
[--backendcfg BACKEND_CONFIG_PATH]
```

#### **--cfg** CONFIG\_PATH

This **mandatory** command line flag specifies the absolute or relative path to the configuration file to be used by Traffic Ops - [cdn.conf](#).

#### **--dbcfg** DB\_CONFIG\_PATH

This **mandatory** command line flag specifies the absolute or relative path to a configuration file used by Traffic Ops to establish connections to the PostgreSQL database - [database.conf](#)

#### **--plugins**

List the installed plugins and exit.

#### **--api-routes**

Print information about all API routes and exit. If also used with the **--cfg** option, also print out the configured routing blacklist information from [cdn.conf](#).

**--riakcfg** RIAK\_CONFIG\_PATH

Deprecated since version 6.0: This optional command line flag specifies the absolute or relative path to a configuration file used by Traffic Ops to establish connections to Riak when used as the Traffic Vault backend - *riak.conf*. Please use "traffic\_vault\_backend": "riak" and "traffic\_vault\_config": {...} (with the contents of *riak.conf*) instead.

---

**Implementation Detail**

The name of this flag is derived from the current database used in the implementation of Traffic Vault - *Riak KV*.

---

**--backendcfg** BACKEND\_CONFIG\_PATH

This optional command line flag specifies the absolute or relative path to a configuration file used by Traffic Ops to act as a reverse proxy and forward requests on the specified paths to the corresponding hosts - *backends.conf*

**--version**

Print version information and exit.

## Configuring

**traffic\_ops\_golang** uses several configuration files, but the most important of these is *cdn.conf*.

## Configuration Files

### cdn.conf

This file deals with the configuration parameters of running Traffic Ops itself. It is a JSON-format set of options and their respective values. *traffic\_ops\_golang* will use whatever file is specified by its *--cfg* option. The keys of the file are described below.

**acme\_accounts** This is an optional array of objects to define ref:external\_account\_binding information to an existing ACME (Automatic Certificate Management Environment) account. The *acme\_provider* and *user\_email* combination must be unique.

New in version 5.1.

**acme\_provider** The certificate provider. This field needs to correlate to the AuthType field for each certificate so the renewal functionality knows which provider to use.

**user\_email** The email used to set up the account with the provider.

**acme\_url** The URL for the ACME.

**kid** The key ID provided by the ACME provider for ref:*external\_account\_binding*.

**hmac\_encoded** The HMAC (Hashed Message Authentication Code) key provided by the ACME provider for ref:*external\_account\_binding*. This should be in Base64 URL encoded.

**acme\_renewal** This object contains the information for the automatic renewal script for certificates.

New in version 5.1.

**renew\_days\_before\_expiration** Set the number of days before expiration date to renew certificates.

**summary\_email** The email address to use for summarizing certificate expiration and renewal status. If it is blank, no email will be sent.

**default\_certificate\_info** This is an optional object to define default values when generating a self signed certificate when an HTTPS delivery service is created or updated. If this is an empty object or not present in the *cdn.conf* then the term “Placeholder” will be used for all fields.

**business\_unit** An optional field which, if present, will represent the business unit for which the SSL certificate was generated

**city** An optional field which, if present, will represent the resident city of the generated SSL certificate

**organization** An optional field which, if present, will represent the organization for which the SSL certificate was generated

**country** An optional field which, if present, will represent the resident country of the generated SSL certificate

**state** An optional field which, if present, will represent the resident state or province of the generated SSL certificate

**geniso** This object contains configuration options for system ISO generation.

**iso\_root\_path** Sets the filesystem path to the root of the ISO generation directory. For default installations, this should usually be set to `/opt/traffic_ops/app/public`.

Deprecated since version ATCv6: The `geniso.iso_root_path` configuration option is unused now that Traffic Ops is rewritten from Perl to Golang and will be removed in a future ATC release.

**See also:**

*Generate ISO*

**hypnotoad** This is a group of options that mainly no longer have any meaning..

**group** Serves no known purpose anymore.

**heartbeat\_timeout** Serves no known purpose anymore.

**listen** This must always be an array containing a single string. This very odd string is apparently a URL. The scheme of the URL doesn't matter, as *traffic\_ops\_golang* ignores that and always uses HTTPS. The host (and optionally port) of the URL used to have a purpose, but no longer does. The “cert” query parameter sets the location of the SSL certificate to use for encrypting connections, while the “key” query parameter is the certificate's corresponding private key. The default configuration file also has the “verify” query parameter which serves no known purpose anymore.

**pid\_file** Serves no known purpose anymore.

**user** Serves no known purpose anymore.

**workers** Serves no known purpose anymore.

**inactivity\_timeout** Serves no known purpose anymore.

**influxdb\_conf\_path** An optional field which gives *traffic\_ops\_golang* the absolute or relative path to an *influxdb.conf* file. Default if not specified is to first check if the *MOJO\_MODE* environment variable is set. If it is, then Traffic Ops will look in the current working directory for a subdirectory named *conf/*, then inside that for a subdirectory with the name that is the value of the *MOJO\_MODE* variable, and inside that directory for a file named *influxdb.conf*. If *MOJO\_MODE* is *not* set, then Traffic Ops will look for a file named *influxdb.conf* in the same directory as this *cdn.conf* file.

New in version 4.0.

**Warning:** While relative paths are allowed, they are discouraged, as the path will be relative to the working directory of the *traffic\_ops\_golang* process itself, not relative to the *cdn.conf* configuration file, which can be confusing.

**ldap\_conf\_location** An optional field which gives *traffic\_ops\_golang* the absolute or relative path to an *ldap.conf* file. Default if not specified is a file named *ldap.conf* in the same directory as this *cdn.conf* file.

**Warning:** While relative paths are allowed, they are discouraged, as the path will be relative to the working directory of the *traffic\_ops\_golang* process itself, not relative to the *cdn.conf* configuration file, which can be confusing.

**lets\_encrypt** New in version 4.1.

**user\_email** A required email address to create an account with Let's

Encrypt or to receive expiration updates. If this is not included then [rate limits](#) may apply for the number of certificates.

**send\_expiration\_email** A boolean option to send email summarizing certificate expiration status

Deprecated since version 5.1: Future versions of Traffic Ops will not support this legacy configuration option, see `acme_renewal`: `{ summary_email: <string> }` instead.

**convert\_self\_signed** A boolean option to convert self signed to Let's Encrypt certificates as they expire. This only works for certificates labeled as Self Signed in the Certificate Source field.

**renew\_days\_before\_expiration** Set the number of days before expiration date to renew certificates.

Deprecated since version 5.1: Future versions of Traffic Ops will not support this legacy configuration option, see `acme_renewal`: `{ renew_days_before_expiration: <int> }` instead.

**environment** This specifies which Let's Encrypt environment to use: 'staging' or 'production'. It defaults to 'production'.

**portal** This section provides information regarding a connected UI with which users interact, so that emails can include links to it.

**base\_url** This URL should be the root and/or landing page of the UI. For Traffic Portal instances, this should include the fragment part of the URL, e.g. `https://trafficportal.infra.ciab.test/#!/.`

**docs\_url** The actual use of this URL is unknown, but supposedly it ought to point to the documentation for the Traffic Control instance. It's hard to imagine a fantastic reason this shouldn't just always be <https://traffic-control-cdn.readthedocs.io>

**email\_from** Most emails sent from the Traffic Ops server will use `to.email_from`, but specifically password reset requests (which contain a link to a fragment under `portal.base_url`) will instead use this as the value of their *From* field.

**pass\_reset\_path** A path to be added to `base_url` that is the URL of the UI's password reset interface. For Traffic Portal instances, this should always be set to "user".

**user\_register\_path** A path to be added to `base_url` that is the URL of the UI's new user registration interface. For Traffic Portal instances, this should always be set to "user".

**secrets** This is an array of strings, which cannot be empty. The first secret in the array is used to encrypt Traffic Ops authentication cookies - multiple Traffic Ops instances serving the same CDN need to share secrets in order for users



logged into one to be able to use their cookie as authentication with other instances.

**smtp** This optional section contains options for connecting to and authenticating with an SMTP (Simple Mail Transfer Protocol) server for sending emails. If this section is undefined (or if `enabled` is explicitly `false`), Traffic Ops will not be able to send emails and certain *Traffic Ops API* endpoints that depend on that functionality will fail to operate.

New in version 4.0.

**address** This is the address of the SMTP which will be used to send emails. Should include the port number, e.g. "localhost:25" for `sendmail(8)` on the Traffic Ops server.

**enabled** A boolean flag that determines whether or not connection to an SMTP ought to be allowed. Whatever the settings of the other fields in the `smtp` object, email cannot and will not be sent if this is `false`.

**password** The password to be used when authenticating with the SMTP server.

**user** The name of the user to be used when authenticating with the SMTP server.

---

**Note:** The SMTP integration currently only supports Login Auth.

---

**to** Contains information to identify Traffic Ops in a network sense.

**base\_url** This field is used to identify the location for the now-removed Traffic Ops UI. It no longer serves any purpose.

**email\_from** Sets the address that will appear in the *From* field of Emails sent by Traffic Ops.

**no\_account\_found\_msg** When a password reset is requested for an email address not registered to any known user, this is the message that will be sent to that email address.

**traffic\_ops\_golang** This group configuration options is used exclusively by *traffic\_ops\_golang*.

**crconfig\_emulate\_old\_path** An optional boolean that controls the value of a part of *Snapshots* that report what *Traffic Ops API* endpoint is used to generate *Snapshots*. If this is `true`, it forces Traffic Ops to report that a legacy, deprecated endpoint is used, whereas if it's `false` Traffic Ops will report the actual, current endpoint. Default if not specified is `false`.

Deprecated since version 3.0: Future versions of Traffic Ops will not support this legacy configuration option, and will always report the current endpoint.

**crconfig\_snapshot\_use\_client\_request\_host** An optional boolean which controls the value of the Traffic Ops server's URL as inserted into *Snapshots*. If this is `true`, then the value used will be taken from the *Host* header of the request that generated the *Snapshot*. If it's `false`, then it will instead use the value of the global "tm.url" *Parameter*. Default if not specified is `false`.

Deprecated since version 3.0: Future versions of Traffic Ops will not support this legacy configuration option, and will always use the global "tm.url" *Parameter*.

**db\_conn\_max\_lifetime\_seconds** An optional field that sets the maximum lifetime in seconds of any given connection to the Traffic Ops Database. If set to zero, connections are held open until explicitly closed. Default if not specified is the value of `DBConnMaxLifetimeSecondsDefault`.

**db\_max\_idle\_connections** An optional limit on the number of connections to the Traffic Ops Database to keep alive while idle. If this is less than `max_db_connections`, that number will be used instead - *even if this field is unset and using its default*. Default if not specified is the value of `DBMaxIdleConnectionsDefault`.

**db\_query\_timeout\_seconds** An optional field specifying a timeout on database *transactions* (not actually single queries in most cases) within API route handlers. Effectively this is a timeout on a single handler's ability to interact with the Traffic Ops Database. Default if not specified is the value of `DefaultDBQueryTimeoutSecs`.

**idle\_timeout** An optional timeout in seconds for idle client connections to Traffic Ops. If set to zero, the value of `read_timeout` will be used instead. If both are zero, then the value of `read_header_timeout` will be used. If all three fields are zero, there is no timeout and connections will be kept alive indefinitely - **not** recommended. Default if not specified is zero.

**insecure** An optional boolean which, if set to `true` will cause Traffic Ops to skip verification of client certificates whenever necessary/possible. If set to `false`, the normal verification behavior is exhibited. Default if not specified is `false`.

Deprecated since version 5.0: Future versions of Traffic Ops will not support this legacy configuration option, see `tls_config`: { `InsecureSkipVerify`: <bool> } instead

**log\_location\_debug** This optional field, if specified, should either be the location of a file to which debug-level output will be logged, or one of the special strings "stdout" which indicates that STDOUT should be used, "stderr" which indicates that STDERR should be used or "null" which indicates that no out-

put of this level should be generated. An empty string ("") and literally `null` are equivalent to `"null"`. Default if not specified is `"null"`.

**log\_location\_error** This optional field, if specified, should either be the location of a file to which error-level output will be logged, or one of the special strings `"stdout"` which indicates that `STDOUT` should be used, `"stderr"` which indicates that `STDERR` should be used or `"null"` which indicates that no output of this level should be generated. An empty string ("") and literally `null` are equivalent to `"null"`. Default if not specified is `"null"`. This field is also used to determine where server profiling statistics are written. Assuming `profiling_enabled` is `true` and `profiling_location` is unset, if this field's value is given as a path to a regular file, a file named `profiling` will be written to the same directory containing the profiling information - overwriting any existing files by that name.

**log\_location\_event** This optional field, if specified, should either be the location of a file to which event-level output will be logged, or one of the special strings `"stdout"` which indicates that `STDOUT` should be used, `"stderr"` which indicates that `STDERR` should be used or `"null"` which indicates that no output of this level should be generated. An empty string ("") and literally `null` are equivalent to `"null"`. Default if not specified is `"null"`.

**log\_location\_info** This optional field, if specified, should either be the location of a file to which informational-level output will be logged, or one of the special strings `"stdout"` which indicates that `STDOUT` should be used, `"stderr"` which indicates that `STDERR` should be used or `"null"` which indicates that no output of this level should be generated. An empty string ("") and literally `null` are equivalent to `"null"`. Default if not specified is `"null"`.

**log\_location\_warning** This optional field, if specified, should either be the location of a file to which warning-level output will be logged, or one of the special strings `"stdout"` which indicates that `STDOUT` should be used, `"stderr"` which indicates that `STDERR` should be used or `"null"` which indicates that no output of this level should be generated. An empty string ("") and literally `null` are equivalent to `"null"`. Default if not specified is `"null"`.

**max\_db\_connections** An optional limit on the number of allowed concurrent connections to the Traffic Ops Database. If it is less than or equal to zero, there is no limit. Default if not specified is zero.

**oauth\_client\_secret** An optional secret string to be shared with

OAuth-capable clients attempting to authenticate via OAuth. The default behavior if this is not defined - or is an empty string ("") or null is to disallow authentication via OAuth.

**Warning:** OAuth support in Traffic Ops is still in its infancy, so most users are advised to avoid defining this field without good cause.

**plugins** An optional array of enabled plugin names. These names must be unique. Note that a plugin that is installed will not be used unless its name appears in this list - thus “enabling” it. If not specified no plugins will be enabled.

**plugin\_config** This optional object maps plugin names - which **must** appear in the **plugins** array - to arbitrary JSON configurations for said plugins. It is up to the plugins themselves to parse these configurations. The default if not specified is no configuration information, somewhat obviously.

**plugin\_shared\_config** This optional object is just an arbitrary JSON object that is converted into a native object and made available to any and all loaded and enabled plugins. A typical use-case for this field is avoiding repetition of identical configuration in **plugin\_config**. The default if not specified is null.

**port** Sets the port on which Traffic Ops will listen for incoming connections.

**profiling\_enabled** An optional boolean which, if **true** will enable the gathering of profiling statistics on the Traffic Ops server. Default if not specified is **false**.

**profiling\_location** An optional string which, if set, should be the absolute path (relative paths are allowed but not recommended) to a file where profiling statistics for the Traffic Ops server will be written. If **profiling\_enabled** is **true** but this is not specified, or is an empty string ("") or null, then a file named “profiling” will be created or overwritten in the same directory as the file specified in **log\_location\_error**. If that file is not a regular file, then Traffic ops will instead create a temporary directory and write profiling statistics to a file named “profiling” within that directory.

**proxy\_keep\_alive** Serves no known purpose anymore.

**proxy\_read\_handler\_timeout** Serves no known purpose anymore.

**proxy\_timeout** Serves no known purpose anymore.

**proxy\_tls\_timeout** Serves no known purpose anymore.

**read\_header\_timeout** An optional timeout in seconds before which Traffic Ops must be able to finish reading the headers of an incoming request or it will drop the connection. If set to zero, there is no timeout. Default if not specified is zero.

**read\_timeout** An optional timeout in seconds before which Traffic Ops must be able to finish reading an entire incoming request (including body) or it will drop the connection. If set to zero, there is no timeout. Default if not specified is zero.

**request\_timeout** An optional timeout in seconds that serves as the maximum time each Traffic Ops middleware can take to execute. If it is exceeded, the text “server timed out” is served in place of a response. If set to 0, 60 is used instead. Default if not specified is 60.

**riak\_port** An optional field that sets the port on which Traffic Ops will try to contact Traffic Vault for storage and retrieval of sensitive encryption keys.

Deprecated since version 6.0: Please use a "port" field in `traffic_vault_config` instead when using `"traffic_vault_backend": "riak"`.

---

### Implementation Detail

The name of this field is derived from the current database used in the implementation of Traffic Vault - [Riak KV](#).

---

**whitelisted\_oauth\_url** An optional array of URLs which are allowed to authenticate Traffic Ops users via OAuth. The default behavior if this field is not defined is to not allow OAuth authentication.

**Warning:** OAuth support in Traffic Ops is still in its infancy, so most users are advised to avoid defining this field without good cause.

**write\_timeout** An optional timeout in seconds set on handlers. After reading a request's header, the server will have this long to send back a response. If set to zero, there is no timeout. Default if not specified is zero.

**traffic\_vault\_backend** New in version 6.0: Optional. The name of which backend to use for Traffic Vault. Currently, the only supported backend is “riak”.

**traffic\_vault\_config** New in version 6.0: Optional. The JSON configuration which is unique to the chosen Traffic Vault backend.

See *Traffic Vault Administration* for the configuration options for each supported backend.

**routing\_blacklist** Optional configuration for explicitly disabling any routes via `disabled_routes`.

New in version 4.0.

**perl\_routes** Serves no known purpose anymore.

Deprecated since version 6.0: This was used back when Traffic Ops was still in the process of being rewritten from Perl. It serves no purpose anymore, and will be removed in the future.

**disabled\_routes** A list of API route IDs to disable. Requests matching these routes will receive a 503 response. To find the route ID for a given path you would like to disable, run `./traffic_ops_golang` using the `--api-routes` option to view all the route information, including route IDs and paths.

**ignore\_unknown\_routes** If `false` (default) return an error and prevent startup if unknown route IDs are found. Otherwise, log a warning and continue startup.

**tls\_config** An optional stanza for TLS configuration. The values of which conform to the `crypto/tls.Config` structure.

**use\_ims** New in version 5.0: This is an optional boolean value to enable the handling of the “If-Modified-Since” HTTP request header. Default: `false`

**role\_based\_permissions** Toggle whether or not to use Role Based Permissions.

New in version 6.1: The blueprint can be seen [Pull Request 5848](#)

**disable\_auto\_cert\_deletion** This optional boolean value can be used to disable the automatic deletion of certificates for Delivery Services that no longer exist (which happens after a CDN Snapshot is taken). Default: `false`.

New in version 6.1.

**cdni** This is an optional section of configurations for CDNI (Content Delivery Network Interconnect) operations.

New in version 6.2.

**dcdn\_id** A string representing this CDN to be used in the JWT (JSON Web Token) and subsequently in CDNI operations.

**user\_cache\_refresh\_interval\_sec** This optional integer value specifies the interval (in seconds) between refreshing the in-memory Users cache. Default: 0 (disabled).

**Warning:** Enabling the Users cache improves performance by reducing the number of queries made to the Traffic Ops database, but it means that it may take up to this many seconds before any changes to Users and/or Roles are enforced.

New in version 7.0.

**server\_update\_status\_cache\_refresh\_interval\_sec** This optional integer value specifies the interval (in seconds) between refreshing the in-memory server update status cache. Default: 0 (disabled).

**Warning:** Enabling the server update status cache improves performance by reducing the number of queries made to the Traffic Ops database, but it means that it may take up to this many seconds before any server updates or revalidations are reflected in the *servers/{{hostname}}/update\_status* API.

New in version 7.0.

### Example `cdn.conf`

```
{
  "hypnotoad" : {
    "listen" : [
      "https://[::]:60443?cert=/etc/pki/tls/certs/localhost.crt&
      ↪key=/etc/pki/tls/private/localhost.key&verify=0x00&ciphers=AES128-
      ↪GCM-SHA256:HIGH:!RC4:!MD5:!aNULL:!EDH:!ED"
    ],
    "user" : "trafops",
    "group" : "trafops",
    "heartbeat_timeout" : 20,
    "pid_file" : "/var/run/traffic_ops.pid",
    "workers" : 12
  },
  "traffic_ops_golang" : {
    "port" : "443",
    "proxy_timeout" : 60,
    "proxy_keep_alive" : 60,
    "proxy_tls_timeout" : 60,
    "proxy_read_header_timeout" : 60,
    "read_timeout" : 60,
    "request_timeout" : 60,
    "read_header_timeout" : 60,
    "write_timeout" : 60,
    "idle_timeout" : 60,
  }
}
```

(continues on next page)

(continued from previous page)

```

    "log_location_error": "/var/log/traffic_ops/error.log",
    "log_location_warning": "",
    "log_location_info": "",
    "log_location_debug": "",
    "log_location_event": "/var/log/traffic_ops/access.log",
    "max_db_connections": 20,
    "db_max_idle_connections": 15,
    "db_conn_max_lifetime_seconds": 60,
    "db_query_timeout_seconds": 20,
    "whitelisted_oauth_urls": [],
    "oauth_client_secret": "",
    "traffic_vault_backend": "",
    "traffic_vault_config": {},
    "routing_blacklist": {
        "ignore_unknown_routes": false,
        "disabled_routes": []
    },
    "profiling_enabled": false,
    "supported_ds_metrics": [ "kpbs", "tps_total", "tps_2xx", "tps_
→3xx", "tps_4xx", "tps_5xx" ],
    "tls_config": {
        "MinVersion": 769
    }
},
"disable_auto_cert_deletion": false,
"user_cache_refresh_interval_sec": 0,
"server_update_status_cache_refresh_interval_sec": 0,
"use_ims": false,
"role_based_permissions": true,
"cors" : {
    "access_control_allow_origin" : "*"
},
"to" : {
    "base_url" : "http://localhost:3000",
    "email_from" : "no-reply@traffic-ops-domain.com",
    "no_account_found_msg" : "A Traffic Ops user account is
→required for access. Please contact your Traffic Ops user
→administrator."
},
"portal" : {
    "base_url" : "https://localhost/#!/",
    "docs_url" : "https://traffic-control-cdn.readthedocs.io/en/
→latest/admin/traffic_portal/usingtrafficportal.html",
    "email_from" : "no-reply@traffic-portal-domain.com",
    "pass_reset_path" : "user",
    "user_register_path" : "user"

```

(continues on next page)



(continued from previous page)

```

},
"secrets" : [
    "mONKEYD0mONKEYSEE."
],
"inactivity_timeout" : 60,
"smtp" : {
    "enabled" : false,
    "user" : "",
    "password" : "",
    "address" : ""
},
"lets_encrypt" : {
    "user_email" : "",
    "send_expiration_email": false,
    "convert_self_signed": false,
    "renew_days_before_expiration": 30,
    "environment": "production"
},
"acme_renewal": {
    "summary_email": "",
    "renew_days_before_expiration": 30
},
"acme_accounts": [
    {
        "acme_provider" : "",
        "user_email" : "",
        "acme_url" : "",
        "kid" : "",
        "hmac_encoded" : ""
    }
],
"default_certificate_info" : {
    "business_unit" : "",
    "city" : "",
    "organization" : "",
    "country" : "",
    "state" : ""
},
"cdni" : {
    "dcdn_id" : ""
}
}

```

### database.conf

This file deals with configuration of the Traffic Ops Database; in particular it tells Traffic Ops how to connect with the database for its current environment. *traffic\_ops\_golang* will read this file in from the path pointed to by its *--dbcfg* flag. `database.conf` is encoded as a JSON object, and its keys are described below.

**dbname** The name of the PostgreSQL database used. Typically different databases are used for different environments, e.g. “trafficops\_test”, “trafficops”, etc. Many environments choose to use `traffic_ops`.

**description** An optional, human friendly description of the database. Generally this should just describe the purpose of the database e.g. “This database is used for integration testing with our toolset”.

**hostname** The hostname (FQDN) of the server that runs the Traffic Ops Database.

**password** The password to use when authenticating with the Traffic Ops database. In a typical install process, the `postinstall` script will ask for a password to use for this connection, and this should match that.

**port** The port number (as a string) on which the Traffic Ops Database is listening for incoming connections. *traffic\_ops\_golang* ignores this and always uses the default PostgreSQL port (5432).

**ssl** A boolean that sets whether or not the Traffic Ops Database encrypts its connections with SSL.

**type** A string that gives the “type” of database pointed to by all the other options. Once upon a time it was possible for this to either be “mysql” or “postgres”, but the only valid value anymore is “postgres” - and *traffic\_ops\_golang* ignores this field entirely (and in fact doesn’t even care if it’s defined at all) and only supports “postgres” databases.

**user** The name of the user as whom to connect to the database. In a typical install process, the `postinstall` script will ask for the name of a user to set up for the Traffic Ops Database, and this should match that. Many environments choose to use `traffic_ops`.

### Example database.conf

```
{
  "description": "Local PostgreSQL database on port 5432",
  "dbname": "traffic_ops",
  "hostname": "localhost",
  "user": "traffic_ops",
  "password": "password",
  "port": "5432",
  "ssl": false,
  "type": "Pg"
}
```

## influxdb.conf

This file deals with configuration of the InfluxDB cluster that serves Traffic Stats; specifically it tells Traffic Ops how to authenticate with the InfluxDB cluster and which measurements to check. *traffic\_ops\_golang* will look for this file at the path given by the value of `influx_db_conf_path` in *cdn.conf*. This file is encoded as a JSON object, and its keys are described below.

### See also:

For more information about InfluxDB, see [the InfluxDB documentation](#).

**cache\_stats\_db\_name** This field sets the name of the “database” (measurement) used to query for *Cache Group* statistics. *traffic\_ops\_golang* will default to "cache\_stats" if this field is not defined. It is recommended that this field not be defined.

**Danger:** The **only** valid value for this is "cache\_stats", if it is anything else Traffic Stats data for *Cache Group* statistics will be inaccessible through the *Traffic Ops API*.

**deliveryservice\_stats\_db\_name** This field sets the name of the “database” (measurement) used to query for *Delivery Service* statistics. *traffic\_ops\_golang* will default to "deliveryservice\_stats" if this field is not defined. It is recommended that this field not be defined.

**Danger:** The **only** valid value for this is "deliveryservice\_stats", if it is anything else Traffic Stats data for *Delivery Service* statistics will be inaccessible through the *Traffic Ops API*.

**password** Sets the password to use when authenticating with InfluxDB clusters.

**secure** An optional boolean that sets whether or not to use SSL encrypted connections to the InfluxDB cluster (the InfluxDB servers would need to be configured to use SSL). Default if not specified is `false`.

**user** Sets the user name as whom to authenticate with InfluxDB clusters.

## Example influxdb.conf

```
{
  "user": "influxuser",
  "password": "password",
  "deliveryservice_stats_db_name": "deliveryservice_stats",
  "cache_stats_db_name": "cache_stats",
```

(continues on next page)

(continued from previous page)

```
"secure": false
}
```

## ldap.conf

This file defines methods of connection to an LDAP (Lightweight Directory Access Protocol) server and semantics for searching for users on it for the purpose of authentication. *trafic\_ops\_golang* will look for this file at the path given by the value of `ldap_conf_location` in *cdn.conf*. `ldap.conf`'s contents are a JSON-encoded object, the keys of which are detailed below.

### See also:

For more information on LDAP see [the LDAP Wikipedia page](#) and [RFC 4511](#).

**admin\_dn** The LDAP DN (Distinguished Name) of the administrative user.

**admin\_pass** The password of the administrative user for the LDAP.

**host** The full hostname of the LDAP server, preceded by a scheme (only `ldap://` and `ldaps://` are supported), optionally including port number.

**insecure** A boolean that tells Traffic Ops whether or not to verify the certificate chain of the LDAP server if it uses TLS-encrypted communications.

**ldap\_timeout\_secs** Sets a timeout in seconds for connections to the LDAP.

**search\_base** The directory relative to which searches for users should be conducted.

**search\_query** A query to be used to search for users. The string `%s` should appear exactly once in this string, where user names will be inserted procedurally by the handler for LDAP logins.

## Example ldap.conf

```
{
  "admin_pass" : "password",
  "search_base" : "dc=prefix,dc=domain,dc=suffix",
  "admin_dn" : "user@prefix.domain.suffix",
  "host" : "ldaps://host:[port]",
  "search_query" : "(&
→(objectCategory=person)(objectClass=user)(userName=%s))",
  "insecure" : false,
  "ldap_timeout_secs" : 20
}
```

## riak.conf

Deprecated since version 6.0: The `riak.conf` configuration file and associated `--riakcfg` flag have been deprecated and will be removed from Traffic Control in the future. Please use `"traffic_vault_backend": "riak"` and put the existing contents of `riak.conf` into `"traffic_vault_config": {...}` in `cdn.conf` instead.

This file sets authentication options for connections to Riak when used as the Traffic Vault backend. `traffic_ops_golang` will look for this file at the path given by the value of the `--riakcfg` flag as passed on startup. The contents of `riak.conf` are encoded as a JSON object, the keys of which are described in *Riak (deprecated)*.

---

### Implementation Detail

The name of this file is derived from the current database used in the implementation of Traffic Vault - *Riak KV*.

---

## backends.conf

This file deals with the configuration parameters of running Traffic Ops as a reverse proxy for certain endpoints that need to be served externally by other backend services. It is a JSON-format set of options and their respective values. `traffic_ops_golang` will use whatever file is specified (if any) by its `--backendscfg` option. The keys of the file are described below.

**routes** This is an array of options to configure Traffic Ops to forward requests of specified types to the appropriate backends.

**path** The regex matching the endpoint that will be served by the backend, for example, `^/api/4.0/foo?$.`

**method** The HTTP method for the above mentioned path, for example, GET or PUT.

**routeId** The integral identifier for the new route being added.

**hosts** An array of the host object, which specifies the protocol, host-name and port where the request (if matched) needs to be forwarded to.

**protocol** The protocol/scheme to be followed while forwarding the requests to the backend service.

**hostname** The hostname of the server where the backend service is running.

**port** The port (integer) on the backend server where the service is running.

**insecure** A boolean specifying whether or not TO should verify the backend server's certificate chain and host name. This is not

recommended for production use. This is an optional parameter, defaulting to `false` when not present.

**permissions** An array of permissions (strings) specifying the permissions required by the user to use this API route.

**opts** A collection of key value pairs to control how the requests should be forwarded/ handled, for example, `"alg": "roundrobin"`. Currently, only `roundrobin` is supported (which is also the default if nothing is specified) by Traffic Ops.

### Example `backends.conf`

```
{
  "routes": [
    {
      "path": "^/api/4.0/foo?$",
      "method": "GET",
      "hosts": [
        {
          "protocol": "https",
          "hostname": "localhost",
          "port": 8444
        },
        {
          "protocol": "https",
          "hostname": "localhost",
          "port": 8445
        }
      ],
      "insecure": true,
      "permissions": [
        "CDN:READ"
      ],
      "routeId": 123456,
      "opts": {
        "alg": "roundrobin"
      }
    },
    {
      "path": "^/api/4.0/foos?$",
      "method": "GET",
      "hosts": [
        {
          "protocol": "https",
          "hostname": "localhost",
          "port": 8444
        }
      ]
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "protocol": "https",
      "hostname": "localhost",
      "port": 8445
    }
  ],
  "insecure": true,
  "permissions": [
    "CDN:READ"
  ],
  "routeId": 123457,
  "opts": {
    "alg": "roundrobin"
  }
}
]
}

```

## Installing the SSL Certificate

By default, Traffic Ops runs as an SSL web server (that is, over HTTPS), and a certificate needs to be installed.

### Self-signed Certificate (Development)

#### 30: Example Procedure

```

$ openssl genrsa -des3 -passout pass:x -out localhost.pass.key 2048
Generating RSA private key, 2048 bit long modulus
...
$ openssl rsa -passin pass:x -in localhost.pass.key -out localhost.key
writing RSA key
$ rm localhost.pass.key

$ openssl req -new -key localhost.key -out localhost.csr
You are about to be asked to enter information that will be
→ incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or
→ a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

```

(continues on next page)

(continued from previous page)

```
-----
Country Name (2 letter code) [XX]:US<enter>
State or Province Name (full name) []:CO<enter>
Locality Name (eg, city) [Default City]:Denver<enter>
Organization Name (eg, company) [Default Company Ltd]: <enter>
Organizational Unit Name (eg, section) []: <enter>
Common Name (eg, your name or your server's hostname) []: <enter>
Email Address []: <enter>
```

Please enter the following 'extra' attributes  
to be sent with your certificate request

A challenge password []: pass<enter>

An optional company name []: <enter>

```
$ openssl x509 -req -sha256 -days 365 -in localhost.csr -signkey _
→localhost.key -out localhost.crt
```

Signature ok

subject=/C=US/ST=CO/L=Denver/O=Default Company Ltd

Getting Private key

```
$ sudo cp localhost.crt /etc/pki/tls/certs
```

```
$ sudo cp localhost.key /etc/pki/tls/private
```

```
$ sudo chown trafops:trafops /etc/pki/tls/certs/localhost.crt
```

```
$ sudo chown trafops:trafops /etc/pki/tls/private/localhost.key
```

### Certificate from Certificate Authority (Production)

---

**Note:** You will need to know the appropriate answers when generating the certificate request file `trafficopss.csr`.

---

#### 31: Example Procedure

```
$ openssl genrsa -des3 -passout pass:x -out trafficops.pass.key 2048
Generating RSA private key, 2048 bit long modulus
...
$ openssl rsa -passin pass:x -in trafficops.pass.key -out trafficops.
→key
writing RSA key
$ rm localhost.pass.key
```

Generate the CSR file needed for CA (Certificate Authority) request

#### 32: Example Certificate Signing Request File Generation

```
$ openssl req -new -key trafficops.key -out trafficops.csr
You are about to be asked to enter information that will be_
→incorporated
```

(continues on next page)



(continued from previous page)

```
into your certificate request.
What you are about to enter is what is called a Distinguished Name or ↵
↵a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]: <enter country code>
State or Province Name (full name) []: <enter state or province>
Locality Name (eg, city) [Default City]: <enter locality name>
Organization Name (eg, company) [Default Company Ltd]: <enter ↵
↵organization name>
Organizational Unit Name (eg, section) []: <enter organizational unit ↵
↵name>
Common Name (eg, your name or your server's hostname) []: <enter server
↵'s hostname name>
Email Address []: <enter e-mail address>

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: <enter challenge password>
An optional company name []: <enter>
$ sudo cp trafficops.key /etc/pki/tls/private
$ sudo chown trafops:trafops /etc/pki/tls/private/trafficops.key
```

You must then take the output file `trafficops.csr` and submit a request to your CA. Once you get approved and receive your `trafficops.crt` file

### 33: Certificate Installation

```
sudo cp trafficops.crt /etc/pki/tls/certs
sudo chown trafops:trafops /etc/pki/tls/certs/trafficops.crt
```

If necessary, install the CA certificate's `.pem` and `.crt` files in `/etc/pki/tls/certs`.

You will need to update `cdn.conf` with any necessary changes.

### 34: Sample ‘listen’ Line When Path to `trafficops.crt` and `trafficops.key` are Known

```
'hypnotoad' => ...
  'listen' => 'https://[::]:443?cert=/etc/pki/tls/certs/trafficops.
→ crt&key=/etc/pki/tls/private/trafficops.key&ca=/etc/pki/tls/certs/
→ localhost.ca&verify=0x00&ciphers=AES128-GCM-SHA256:HIGH:!RC4:!MD5:!
→ aNULL:!EDH:!ED'
  ...
```

## Managing Traffic Ops Extensions

Traffic Ops supports “*Check Extensions*”, which are analytics scripts that collect and display information as columns in the table under *Monitor* → *Cache Checks* in Traffic Portal.

### See also:

Traffic Ops also supports a more involved type of extension in the form of *Go Plugins*.

## Check Extensions

Check Extensions are scripts that, after registering with Traffic Ops, have a column reserved in the *Monitor* → *Cache Checks* view and usually run periodically using `cron(8)`. Each extension is a separate executable located in `$TO_HOME/bin/checks/` on the Traffic Ops server (though all of the default extensions are written in Perl, this is in *no way* a requirement; they can be any valid executable). The currently registered extensions can be listed by running `/opt/traffic_ops/app/bin/extensions -a`. Some extensions automatically registered with the Traffic Ops database (`to_extension` table) at install time (see `traffic_ops/app/db/seeds.sql`). However, `cron(8)` must still be configured to run these checks periodically. The extensions are called like so:

### 35: Example Check Extension Call

```
$TO_HOME/bin/checks/<name> -c "{\"base_url\": \"https://\",<traffic_
→ ops_ip>\", \"check_name\": \"<check_name>\"}" -l <log level>
```



**name** The basename of the extension executable

**traffic\_ops\_ip** The IP address or FQDN of the Traffic Ops server

**check\_name** The name of the check e.g. CDU, CHR, DSCP, MTU, etc...

**log\_level** A whole number between 1 and 4 (inclusive), with 4 being the most verbose. Implementation of this field is optional

It is the responsibility of the check extension script to iterate over the servers it wants to check and post the results. An example script might proceed by logging into the Traffic Ops server using the HTTPS `base_url` provided on the command line. The script is hard-coded with an authentication token that is also provisioned in the Traffic Ops User database. This token allows

the script to obtain a cookie used in later communications with the Traffic Ops API. The script then obtains a list of all *cache servers* to be polled by accessing *servers*. This list is then iterated, running a command to gather the stats from each server. For some extensions, an HTTP GET request might be made to the ATS *astats* plugin, while for others the server might be pinged, or a command might run over *ssh(1)*. The results are then compiled into a numeric or boolean result and the script submits a POST request containing the result back to Traffic Ops using *servercheck*. A check extension can have a column of 's and 's (CHECK\_EXTENSION\_BOOL) or a column that shows a number (CHECK\_EXTENSION\_NUM).

## Check Extensions Installed by Default

**CDU (Cache Disk Usage)** This check shows how much of the available total cache disk is in use. A “warm” *cache server* should show 100.00.

**CHR (Cache Hit Ratio)** The cache hit ratio for the cache in the last 15 minutes (the interval is determined by the *cron(8)* entry).

**DSCP** Checks if the returning traffic from the cache has the correct DSCP value as assigned in the *Delivery Service*. (Some routers will overwrite DSCP)

**MTU (Maximum Transmission Unit)** Checks if the Traffic Ops host (if that is the one running the check) can send and receive 8192B packets to the *ip\_address* of the server in the server table.

**ORT (Operational Readiness Test)** The ORT column shows how many changes the *ORT* script would apply if it was run. The number in this column should be 0 for *cache servers* that do not have updates pending.

**10G** Is the *ip\_address* (the main IPv4 address) from the server table ping-able?

**ILO (Integrated Lights-Out)** Is the *ilo\_ip\_address* (the lights-out-management IPv4 address) from the server table ping-able?

**10G6** Is the *ip6\_address* (the main IPv6 address) from the server table ping-able?

**FQDN** Is the FQDN (the concatenation of *host\_name* and *.* and *domain\_name* from the server table) ping-able?

**RTR (Responds to Traffic Router)** Checks the state of each *cache server* as perceived by all Traffic Monitors (via Traffic Router). This extension asks each Traffic Router for the state of the *cache server*. A check failure is indicated if one or more monitors report an error for a *cache server*. A *cache server* is only marked as good if all reports are positive.

---

**Note:** This is a pessimistic approach, opposite of how Traffic Monitor marks a *cache server* as up, i.e. “the optimistic approach”.

---

## Example Cron File

The `cron(8)` file should be edited by running `crontab(1)` as the `traffops` user, or with `sudo(8)`. You may need to adjust the path to your `$TO_HOME` to match your system. Edit `$TO_USER` and `$TO_PASS` to match `ORT` input parameters.

36: Example Cron File

```
PERL5LIB=/opt/traffic_ops/app/local/lib/perl5:/opt/traffic_ops/app/lib

# IPv4 ping examples - The 'select: ["hostName","domainName"]' works but,
→ if you want to check DNS resolution use FQDN.
*/15 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"10G\", \
→ "select\: [\"hostName\", \"domainName\"]}" >> /var/log/traffic_ops/
→ extensionCheck.log 2>&1
*/15 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"10G\", \
→ "select\: \"ipAddress\"" >> /var/log/traffic_ops/extensionCheck.
→ log 2>&1
*/15 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"10G\", \"name\
→ \": \"IPv4 Ping\", \"select\: \"ipAddress\", \"syslog_facility\: \
→ "local0\"" > /dev/null 2>&1

# IPv6 ping examples
*/15 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"10G6\", \"name\
→ \": \"IPv6 Ping\", \"select\: \"ip6Address\", \"syslog_facility\: \
→ "local0\"" >/dev/null 2>&1
*/15 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"10G6\", \
→ "select\: \"ip6Address\"" >> /var/log/traffic_ops/extensionCheck.
→ log 2>&1

# iLO ping
18 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"ILO\", \
→ "select\: \"iloIpAddress\"" >> /var/log/traffic_ops/extensionCheck.
→ log 2>&1
18 * * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"ILO\", \"name\
→ \": \"ILO ping\", \"select\: \"iloIpAddress\", \"syslog_facility\: \
→ "local0\"" >/dev/null 2>&1

# MTU ping
45 0 * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
→ "base_url\: \"https://localhost\", \"check_name\: \"MTU\", \
→ "select\: \"ipAddress\"" >> /var/log/traffic_ops/extensionCheck.
→ log 2>&1
```

(continues on next page)

(continued from previous page)

```

45 0 * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"MTU\", \
↳ "select\: \"ip6Address\"}" >> /var/log/traffic_ops/extensionCheck.
↳ log 2>&1
45 0 * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"MTU\", \"name\
↳ \": \"Max Trans Unit\", \"select\: \"ipAddress\", \"syslog_facility\
↳ \": \"local0\"}" > /dev/null 2>&1
45 0 * * * root /opt/traffic_ops/app/bin/checks/ToPingCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"MTU\", \"name\
↳ \": \"Max Trans Unit\", \"select\: \"ip6Address\", \"syslog_facility\
↳ \": \"local0\"}" > /dev/null 2>&1

# FQDN
27 * * * * root /opt/traffic_ops/app/bin/checks/ToFQDNCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"FQDN\"\" >> /
↳ var/log/traffic_ops/extensionCheck.log 2>&1
27 * * * * root /opt/traffic_ops/app/bin/checks/ToFQDNCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"FQDN\", \"name\
↳ \": \"DNS Lookup\", \"syslog_facility\: \"local0\"}" > /dev/null 2>&1

# DSCP
36 * * * * root /opt/traffic_ops/app/bin/checks/ToDSCPCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"DSCP\", \"cms_
↳ interface\: \"eth0\"}" >> /var/log/traffic_ops/extensionCheck.log 2>
↳ &1
36 * * * * root /opt/traffic_ops/app/bin/checks/ToDSCPCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"DSCP\", \"name\
↳ \": \:term:Delivery Service\", \"cms_interface\: \"eth0\", \
↳ "syslog_facility\: \"local0\"}" > /dev/null 2>&1

# RTR
10 * * * * root /opt/traffic_ops/app/bin/checks/ToRTRCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"RTR\"}" >> /
↳ var/log/traffic_ops/extensionCheck.log 2>&1
10 * * * * root /opt/traffic_ops/app/bin/checks/ToRTRCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"RTR\", \"name\
↳ \": \"Content Router Check\", \"syslog_facility\: \"local0\"}" > /
↳ dev/null 2>&1

# CHR
*/15 * * * * root /opt/traffic_ops/app/bin/checks/ToCHRCheck.pl -c "{\
↳ "base_url\: \"https://localhost\", \"check_name\: \"CHR\"}" >> /
↳ var/log/traffic_ops/extensionCheck.log 2>&1

# CDU

```

(continues on next page)

(continued from previous page)

```
20 * * * * root /opt/traffic_ops/app/bin/checks/ToCDUCheck.pl -c "{\
→"base_url\: \"https://localhost\", \"check_name\: \"CDU\"}" >> /
→var/log/traffic_ops/extensionCheck.log 2>&1

# ORT
40 * * * * ssh_key_edge_user /opt/traffic_ops/app/bin/checks/
→ToORTCheck.pl -c "{\"base_url\: \"https://localhost\", \"check_name\
→\": \"ORT\", \"to_user\:\"$TO_USER\", \"to_pass\:\"$TO_PASS\"}" >>
→ /var/log/traffic_ops/extensionCheck.log 2>&1
40 * * * * ssh_key_edge_user /opt/traffic_ops/app/bin/checks/
→ToORTCheck.pl -c "{\"base_url\: \"https://localhost\", \"check_name\
→\": \"ORT\", \"name\: \"Operational Readiness Test\", \"syslog_
→facility\: \"local0\", \"to_user\:\"$TO_USER\", \"to_pass\:\"$TO_
→PASS\"}" > /dev/null 2>&1
```

### 3.1.2 Environment Creation

Apache Traffic Control is a complex set of components to construct a functional CDN. To help users get started more quickly, two paths exist to help based on the audience and needs. These tools are optional and administrators may use whatever means are at their disposal to perform their deployments.

#### CDN-in-a-Box (CIAB)

##### Audience for CIAB

- Developers who need a local full-stack to aid in development using Docker
- A common platform for developers to write tests against and executed by a CI system
- New users who are learning how a CDN using Apache Traffic Control works
- Functional demonstrations

##### CDN in a Box

“CDN in a Box” is a name given to the time-honored tradition of new Traffic Control developers/potential users attempting to set up their own, miniature CDN to just see how it all fits together. Historically, this has been a nightmare of digging through leftover `virsh` scripts and manually configuring pretty hefty networking changes (don’t even get me started on DNS) and just generally having a bad time. For a few years now, different people had made it to various stages of merging the project into Docker for ease of networking, but certain constraints hampered progress - until now. The project has finally reached a working state, and now getting a mock/test CDN running can be a very simple task (albeit rather time-consuming).

## Getting Started

Because it runs in Docker, the only true prerequisites are:

- Docker version  $\geq$  17.05.0-ce
- Docker Compose<sup>1</sup> version  $\geq$  1.9.0

## Building

The CDN in a Box directory is found within the Traffic Control repository at `infrastructure/cdn-in-a-box/`. CDN in a Box relies on the presence of pre-built `component.rpm` files for the following Traffic Control components:

- Traffic Monitor - at `infrastructure/cdn-in-a-box/traffic_monitor/traffic_monitor.rpm`
- Traffic Ops - at `infrastructure/cdn-in-a-box/traffic_ops/traffic_ops.rpm`
- Traffic Portal - at `infrastructure/cdn-in-a-box/traffic_portal/traffic_portal.rpm`
- Traffic Router - at `infrastructure/cdn-in-a-box/traffic_router/traffic_router.rpm` - also requires an Apache Tomcat RPM at `infrastructure/cdn-in-a-box/traffic_router/tomcat.rpm`

---

**Note:** These can also be specified via the `RPM` variable to a direct Docker build of the component - with the exception of Traffic Router, which instead accepts `TRAFFIC_ROUTER_RPM` to specify a Traffic Router RPM and `TOMCAT_RPM` to specify an Apache Tomcat RPM.

---

These can all be supplied manually via the steps in *Building Traffic Control* (for Traffic Control component RPMs) or via some external source. Alternatively, the `infrastructure/cdn-in-a-box/Makefile` file contains recipes to build all of these - simply run `make(1)` from the `infrastructure/cdn-in-a-box/` directory. Once all RPM dependencies have been satisfied, run `docker-compose build --parallel` from the `infrastructure/cdn-in-a-box/` directory to construct the images needed to run CDN in a Box.

---

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make native` instead of `make` to build the RPMs quickly. Another option is running `make -j4` to build 4 components at once, if your computer can handle it.

---

---

**Tip:** When updating CDN-in-a-Box, there is no need to remove old images before building new ones. Docker detects which files are updated and only reuses cached layers that have not changed.

---

---

<sup>1</sup> It is perfectly possible to build and run all containers without Docker Compose, but it's not recommended and not covered in this guide.



By default, CDN in a Box will be based on Rocky Linux 8. To base CDN in a Box on CentOS 7, set the `BASE_IMAGE` environment variable to `centos` and set the `RHEL_VERSION` environment variable to 7 (for CDN in a Box, `BASE_IMAGE` defaults to `rockylinux` and `RHEL_VERSION` defaults to 8):

37: Building CDN in a Box to run CentOS 7 instead of Rocky Linux 8

```
export BASE_IMAGE=centos RHEL_VERSION=7
make # Builds RPMs for CentOS 7
docker-compose build --parallel # Builds CentOS 7 CDN in a Box images
```

### Usage

In a typical scenario, if the steps in *Building* have been followed, all that's required to start the CDN in a Box is to run `docker-compose up` - optionally with the `-d` flag to run without binding to the terminal - from the `infrastructure/cdn-in-a-box/` directory. This will start up the entire stack and should take care of any needed initial configuration. The services within the environment are by default not exposed locally to the host. If this is the desired behavior when bringing up CDN in a Box the command `docker-compose -f docker-compose.yml -f docker-compose.expose-ports.yml up` should be run. The ports are configured within the `infrastructure/cdn-in-a-box/docker-compose.expose-ports.yml` file, but the default ports are shown in *Service Info*. Some services have credentials associated, which are totally configurable in *variables.env*.



Table 52: Service Info

Service	Ports exposed and their usage	Username	Password
DNS	DNS name resolution on 9353	N/A	N/A
Edge Tier Cache	Apache Trafficserver 8.1 HTTP caching reverse proxy on port 9000	N/A	N/A
Mid Tier Cache	Apache Trafficserver 8.1 HTTP caching forward proxy on port 9100	N/A	N/A
Second Mid-Tier Cache (parent of the first Mid-Tier Cache)	Apache Trafficserver 8.1 HTTP caching forward proxy on port 9100	N/A	N/A
Mock Origin Server	Example web page served on port 9200	N/A	N/A
SMTP Server	Passwordless, cleartext SMTP server on port 25 (no relay) Web interface exposed on port 4443 (port 443 in the container)	N/A	N/A
Traffic Monitor	Web interface and API on port 80	N/A	N/A
Traffic Ops	API endpoints on port 6443	TO_ADMIN_USER in <i>variables.env</i>	TO_ADMIN_PASSWORD in <i>variables.env</i>
Traffic Ops PostgreSQL Database	PostgresQL connections accepted on port 5432 (database name: DB_NAME in <i>variables.env</i> )	DB_USER in <i>variables.env</i>	DB_USER_PASS in <i>variables.env</i>
Traffic Portal	Web interface on 443 (Javascript required)	TO_ADMIN_USER in <i>variables.env</i>	TO_ADMIN_PASSWORD in <i>variables.env</i>
Traffic Router	Web interfaces on ports 3080 (HTTP) and 3443 (HTTPS), with a DNS service on 53 and an API on 3333 (HTTP) and 2222 (HTTPS)	N/A	N/A
Traffic Vault	Riak key-value store on port 8010	TV_ADMIN_USER in <i>variables.env</i>	TV_ADMIN_PASSWORD in <i>variables.env</i>
Traffic Stats	N/A	N/A	N/A
Traffic Stats Influxdb	Influxdbd connections accepted on port 8086 (database name: cache_stats, daily_stats and deliveryservice_stats)	INFLUXDB_ADMIN_USER in <i>variables.env</i>	INFLUXDB_ADMIN_PASSWORD in <i>variables.env</i>

**See also:**

*Traffic Router API* and *Traffic Monitor APIs*

While the components may be interacted with by the host using these ports, the true operation of the CDN can only truly be seen from within the Docker network. To see the CDN in action, connect to a container within the CDN in a Box project and use cURL to request the URL `http://video.demo1.mycdn.ciab.test` which will be resolved by the DNS container to the IP of the Traffic Router, which will provide a 302 FOUND response pointing to the Edge-

Tier cache. A typical choice for this is the “enroller” service, which has a very nuanced purpose not discussed here but already has the `curl(1)` command line tool installed. For a more user-friendly interface into the CDN network, see *VNC Server*.

To test the demo1 Delivery Service:

### 38: Example Command to See the CDN in Action

```
sudo docker-compose exec enroller curl -L "http://video.demo1.mycdn.
→ciab.test"
```

To test the `foo.kabletown.net`. Federation:

### 39: Query the Federation CNAME using the Delivery Service hostname

```
sudo docker-compose exec trafficrouter dig +short @trafficrouter.infra.
→ciab.test -t CNAME video.demo2.mycdn.ciab.test
```

```
# Expected response:
foo.kabletown.net.
```

When the CDN is to be shut down, it is often best to do so using `sudo docker-compose down -v` due to the use of shared volumes in the system which might interfere with a proper initialization upon the next run.

## Readiness Check

In order to check the “readiness” of your CDN, you can optionally start the Readiness Container, which will continually `curl(1)` the *Delivery Services* in your CDN until they all return successful responses before exiting successfully.

### 40: Example Command to Run the Readiness Container

```
sudo docker-compose -f docker-compose.readiness.yml up
```

## Integration Tests

There also exist TP and TO integration tests containers. Both of these containers assume that CDN in a Box is already running on the local system.

### 41: Running TP Integration Tests

```
sudo docker-compose -f docker-compose.traffic-portal-test.yml up
```

## 42: Running TO Integration Tests

```
sudo docker-compose -f docker-compose.traffic-ops-test.yml up
```

**Note:** If all CDN in a Box containers are started at once (example: `docker-compose -f docker-compose.yml -f docker-compose.traffic-ops-test.yml up -d edge enroller dns db smtp trafficops trafficvault integration`), the *Enroller* initial data load is skipped to prevent data conflicts with the *Traffic Ops API tests fixtures*.

### variables.env

```
TV_AES_KEY_LOCATION=/opt/traffic_ops/app/conf/aes.key
# Unset TV_BACKEND to use riak as the traffic_vault backend and run
# the traffic_vault image from the optional directory
TV_BACKEND=postgres
TLD_DOMAIN=ciab.test
INFRA_SUBDOMAIN=infra
CDN_NAME=CDN-in-a-Box
CDN_SUBDOMAIN=mycdn
DS_HOSTS='demo1 demo2 demo3'
FEDERATION_CNAME=foo.kabletown.net.
X509_CA_NAME=CIAB-CA
X509_CA_COUNTRY=US
X509_CA_STATE=Colorado
X509_CA_CITY=Denver
X509_CA_COMPANY=Kabletown
X509_CA_ORG=CDN-in-a-Box
X509_CA_ORGUNIT=CDN-in-a-Box
X509_CA_EMAIL=no-reply@infra.ciab.test
X509_CA_DIGEST=sha256
X509_CA_DURATION_DAYS=365
X509_CA_KEYTYPE=rsa
X509_CA_KEYSIZE=4096
X509_CA_UMASK=0000
X509_CA_DIR=/shared/ssl
X509_CA_PERSIST_DIR=/ca
X509_CA_PERSIST_ENV_FILE=/ca/environment
X509_CA_ENV_FILE=/shared/ssl/environment
DB_NAME=traffic_ops
TV_DB_NAME=traffic_vault
DB_PORT=5432
DB_SERVER=db
TV_DB_SERVER=db
```

(continues on next page)

(continued from previous page)

```
DB_USER_PASS=twelve
DB_USER=traffic_ops
TV_DB_USER=traffic_vault
TV_DB_USER_PASS=twelve
TV_DB_PORT=5432
DNS_SERVER=dns
DBIC_TRACE=0
ENROLLER_HOST=enroller
PGPASSWORD=twelve
POSTGRES_PASSWORD=twelve
EDGE_HOST=edge
INFLUXDB_HOST=influxdb
INFLUXDB_PORT=8086
INFLUXDB_ADMIN_USER=influxadmin
INFLUXDB_ADMIN_PASSWORD=influxadminpassword
GRAFANA_ADMIN_USER=grafanaadmin
GRAFANA_ADMIN_PASSWORD=grafanaadminpassword
GRAFANA_PORT=443
MID_01_HOST=mid-01
MID_02_HOST=mid-02
ORIGIN_HOST=origin
SMTP_HOST=smtp
SMTP_PORT=25
TM_HOST=trafficmonitor
TM_PORT=80
TM_EMAIL=tmonitor@cdn.example.com
TM_PASSWORD=jhds1vhdfsuklvfhsuvlhs
TM_USER=tmon
TM_LOG_ACCESS=stdout
TM_LOG_EVENT=stdout
TM_LOG_ERROR=stdout
TM_LOG_WARNING=stdout
TM_LOG_INFO=stdout
TM_LOG_DEBUG=stdout
TO_ADMIN_PASSWORD=twelve12
TO_ADMIN_USER=admin
TO_ADMIN_FULL_NAME='James Cole'
# Set ENROLLER_DEBUG_ENABLE to true`to debug the enroller with Delve
ENROLLER_DEBUG_ENABLE=false
# To debug a t3c component with Delve in the `edge` container, set T3C_
→DEBUG_COMPONENT_EDGE to t3c-apply, t3c-check, t3c-check-refs, t3c-
→check-reload, t3c-diff, t3c-generate, t3c-request, or t3c-update
T3C_DEBUG_COMPONENT_EDGE=none
# To debug a t3c component with Delve in the `mid-01` container, set
→T3C_DEBUG_COMPONENT_MID_01 to t3c-apply, t3c-check, t3c-check-refs,
→t3c-check-reload, t3c-diff, t3c-generate, t3c-request, or t3c-update
```

(continues on next page)

(continued from previous page)

```
T3C_DEBUG_COMPONENT_MID_01=none
# To debug a t3c component with Delve in the `mid-02` container, set
→T3C_DEBUG_COMPONENT_MID_02 to t3c-apply, t3c-diff, t3c-generate, t3c-
→request, t3c-update, or t3c-verify
T3C_DEBUG_COMPONENT_MID_02=none
# Set TM_DEBUG_ENABLE to true to debug Traffic Monitor with Delve
TM_DEBUG_ENABLE=false
# Set TO_DEBUG_ENABLE to true to debug Traffic Ops with Delve
TO_DEBUG_ENABLE=false
# Set TR_DEBUG_ENABLE to true to debug Traffic Router with JPDA
TR_DEBUG_ENABLE=false
# Set TS_DEBUG_ENABLE to true to debug Traffic Stats with Delve
TS_DEBUG_ENABLE=false
TO_EMAIL=cdnadmin@example.com
TO_HOST=trafficops
TO_PORT=443
TO_LOG_ERROR=/var/log/traffic_ops/error.log
TO_LOG_WARNING=/var/log/traffic_ops/warning.log
TO_LOG_INFO=/var/log/traffic_ops/info.log
#TO_LOG_DEBUG=/var/log/traffic_ops/debug.log
TO_LOG_DEBUG=/dev/null
TO_LOG_EVENT=/var/log/traffic_ops/event.log
TP_HOST=trafficportal
TP_EMAIL=tp@cdn.example.com
TR_HOST=trafficrouter
TR_DNS_PORT=53
TR_HTTP_PORT=80
TR_HTTPS_PORT=443
TR_API_PORT=3333
TP_PORT=443
TS_EMAIL=tstats@cdn.example.com
TS_HOST=trafficstats
TS_PASSWORD=trafficstatspassword
TS_USER=tstats
TV_HOST=trafficvault
TV_USER=tvault
TV_PASSWORD=mwL5GP6Ghu_uJpkfjfiBmii3l9vfgLl0
TV_EMAIL=tvault@cdn.example.com
TV_ADMIN_USER=admin
TV_ADMIN_PASSWORD=riakAdmin
TV_RIAK_USER=riakuser
TV_RIAK_PASSWORD=riakPassword
TV_INT_PORT=8087
TV_HTTP_PORT=8098
TV_HTTPS_PORT=8088
ENROLLER_DIR=/shared/enroller
```

(continues on next page)

(continued from previous page)

```
AUTO_SNAPQUEUE_ENABLED=true
AUTO_SNAPQUEUE_SERVERS=trafficops,trafficmonitor,trafficrouter,edge,
↪mid-01,mid-02
AUTO_SNAPQUEUE_POLL_INTERVAL=2
AUTO_SNAPQUEUE_ACTION_WAIT=2
```

**Note:** While these port settings can be changed without hampering the function of the CDN in a Box system, note that changing a port without also changing the matching port-mapping in `infrastructure/cdn-in-a-box/docker-compose.yml` for the affected service *will* make it unreachable from the host.

## X.509 SSL/TLS Certificates

All components in Apache Traffic Control utilize SSL/TLS secure communications by default. For SSL/TLS connections to properly validate within the “CDN in a Box” container network a shared self-signed X.509 Root CA is generated at the first initial startup. An X.509 Intermediate CA is also generated and signed by the Root CA. Additional “wildcard” certificates are generated/signed by the Intermediate CA for each container service and `demo1`, `demo2`, and `demo3` *Delivery Services*. All certificates and keys are stored in the `ca` host volume which is located at `infrastructure/cdn-in-a-box/traffic_ops/ca`<sup>4</sup>.

Table 53: Self-Signed X.509 Certificate List

Filename	Description	X.509 CN/SAN
CIAB-CA-root.crt	Shared Root CA Certificate	N/A
CIAB-CA-intr.crt	Shared Intermediate CA Certificate	N/A
CIAB-CA-fullchain.crt	Shared CA Certificate Chain Bundle <sup>5</sup>	N/A
infra.ciab.test.crt	Infrastructure Certificate	<code>prefix.infra.ciab.test</code>
demo1.mycdn.ciab.test.crt	Demo1 <i>Delivery Service</i> Certificate	<code>prefix.demo1.mycdn.ciab.test</code>
demo2.mycdn.ciab.test.crt	Demo2 <i>Delivery Service</i> Certificate	<code>prefix.demo2.mycdn.ciab.test</code>
demo3.mycdn.ciab.test.crt	Demo3 <i>Delivery Service</i> Certificate	<code>prefix.demo3.mycdn.ciab.test</code>

<sup>4</sup> The `ca` volume is not purged with normal `docker volume` commands. This feature is by design to allow the existing shared SSL certificate to be trusted at the system level across restarts. To re-generate all SSL certificates and keys, remove the `infrastructure/cdn-in-a-box/traffic_ops/ca` directory before startup.

<sup>5</sup> The full chain bundle is a file that contains both the Root and Intermediate CA certificates.

## Trusting the Certificate Authority

For developer and testing use-cases, it may be necessary to have full x509 CA validation by HTTPS clients<sup>67</sup>. For x509 validation to work properly, the self-signed x509 CA certificate must be trusted either at the system level or by the client application itself.

---

**Note:** HTTP Client applications such as Chromium, Firefox, *curl(1)*, and *wget(1)* can also be individually configured to trust the CA certificate. Review each program's respective documentation for instructions.

---

## Importing the CA Certificate on OSX

1. Copy the CIAB root and intermediate CA certificates from `infrastructure/cdn-in-a-box/traffic_ops/ca/` to the Mac.
2. Double-click the CIAB root CA certificate to open it in Keychain Access.
3. The CIAB root CA certificate appears in login.
4. Copy the CIAB root CA certificate to System.
5. Open the CIAB root CA certificate, expand *Trust*, select *Use System Defaults*, and save your changes.
6. Reopen the CIAB root CA certificate, expand *Trust*, select *Always Trust*, and save your changes.
7. Delete the CIAB root CA certificate from login.
8. Repeat the previous steps with the Intermediate CA certificate to import it as well
9. Restart all HTTPS clients (browsers, etc).

## Importing the CA certificate on Windows

1. Copy the CIAB root CA and intermediate CA certificates from `infrastructure/cdn-in-a-box/traffic_ops/ca/` to Windows filesystem.
2. As Administrator, start the Microsoft Management Console.
3. Add the Certificates snap-in for the computer account and manage certificates for the local computer.
4. Import the CIAB root CA certificate into *Trusted Root Certification Authorities* → *Certificates*.
5. Import the CIAB intermediate CA certificate into *Trusted Root Certification Authorities* → *Certificates*.

---

<sup>6</sup> All containers within CDN-in-a-Box start up with the self-signed CA already trusted.

<sup>7</sup> The 'demo1' *Delivery Service* X509 certificate is automatically imported into Traffic Vault on startup.

6. Restart all HTTPS clients (browsers, etc).

### Importing the CA certificate on Rocky Linux 8 (Linux)

1. Copy the CIAB full chain CA certificate bundle from `infrastructure/cdn-in-a-box/traffic_ops/ca/CIAB-CA-fullchain.crt` to path `/etc/pki/ca-trust/source/anchors/`.
2. Run `update-ca-trust-extract` as the root user or with `sudo(8)`.
3. Restart all HTTPS clients (browsers, etc).

### Importing the CA certificate on Ubuntu (Linux)

1. Copy the CIAB full chain CA certificate bundle from `infrastructure/cdn-in-a-box/traffic_ops/ca/CIAB-CA-fullchain.crt` to path `/usr/local/share/ca-certificates/`.
2. Run `update-ca-certificates` as the root user or with `sudo(8)`.
3. Restart all HTTPS clients (browsers, etc).

## Advanced Usage

This section will be amended as functionality is added to the CDN in a Box project.

### The Enroller

The “enroller” began as an efficient way for Traffic Ops to be populated with data as CDN in a Box starts up. It connects to Traffic Ops as the “admin” user and processes files placed in the docker volume shared between the containers. The enroller watches each directory within the `/shared/enroller` directory for new `filename.json` files to be created there. These files must follow the format outlined in the API guide for the POST method for each data type, (e.g. for a region, follow the guidelines for [POST/regions](#)). Of note, the enroller does not require fields that reference database ids for other objects within the database.

#### **--dir** directory

Base directory to watch for data. Mutually exclusive with `--http`.

#### **--http** port

Act as an HTTP server for POST requests on this port. Mutually exclusive with `--dir`.

#### **--started** filename

The name of a file which will be created in the `--dir` directory when given, indicating service was started (default: “enroller-started”).

The enroller runs within CDN in a Box using `--dir` which provides the above behavior. It can also be run using `--http` to instead have it listen on the indicated port. In this case,



it accepts only POST requests with the JSON provided in the request payload, e.g. `curl -X POST https://enroller/api/4.0/regions -d @newregion.json`. CDN in a Box does not currently use this method, but may be modified in the future to avoid using the shared volume approach.

## Auto Snapshot/Queue-Updates

An automatic *Snapshot* of the current Traffic Ops CDN configuration/topology will be performed once the “enroller” has finished loading all of the data and a minimum number of servers have been enrolled. To enable this feature, set the boolean `AUTO_SNAPQUEUE_ENABLED` to `true`<sup>8</sup>. The *Snapshot* and *Queue Updates* actions will not be performed until all servers in `AUTO_SNAPQUEUE_SERVERS` (comma-delimited string) have been enrolled. The current enrolled servers will be polled every `AUTO_SNAPQUEUE_POLL_INTERVAL` seconds, and each action (*Snapshot* and *Queue Updates*) will be delayed `AUTO_SNAPQUEUE_ACTION_WAIT` seconds<sup>9</sup>.

## Mock Origin Service

The default “origin” service container provides a basic static file HTTP server as the central repository for content. Additional files can be added to the origin root content directory located at `infrastructure/cdn-in-a-box/origin/content`. To request content directly from the origin directly and bypass the CDN:

- Origin Service URL: `http://origin.infra.ciab.test/index.html`
- Docker Host: `http://localhost:9200/index.html`

## Optional Containers

All optional containers that are not part of the core CDN-in-a-Box stack are located in the `infrastructure/cdn-in-a-box/optional` directory.

- `infrastructure/cdn-in-a-box/optional/docker-compose.NAME.yml`
- `infrastructure/cdn-in-a-box/optional/NAME/Dockerfile`

Multiple optional containers may be combined by using a shell alias:

### 43: Starting Optional Containers with an Alias

```
# From the infrastructure/cdn-in-a-box directory
# (Assuming the names of the optional services are stored in the_
→ `NAME1` and `NAME2` environment variables)
alias mydc="docker-compose -f $PWD/docker-compose.yml -f $PWD/optional/
→ docker-compose.$NAME1.yml -f $PWD/optional/docker-compose.$NAME2.yml
→ "
```

(continues on next page)

<sup>8</sup> Automatic *Snapshot/Queue Updates* is enabled by default in `variables.env`.

<sup>9</sup> Server poll interval and delay action wait are defaulted to a value of 2 seconds.

(continued from previous page)

```
docker volume prune -f
mydc build
mydc up
```

## VNC Server

The TightVNC optional container provides a basic lightweight window manager (fluxbox), Firefox browser, xterm, and a few other utilities within the CDN-In-A-Box “tcnet” bridge network. This can be very helpful for quick demonstrations of CDN-in-a-Box that require the use of real container network FQDNs and full X.509 validation.

1. Download and install a VNC client. TightVNC client is preferred as it supports window resizing, host-to-vnc copy/pasting, and optimized frame buffer compression.
2. Set your VNC console password by adding the VNC\_PASSWD environment variable to `infrastructure/cdn-in-a-box/variables.env`. The password needs to be at least six characters long. The default password is randomized for security.
3. Start up CDN-in-a-Box stack. It is recommended that this be done using a custom bash alias

### 44: CIAB Startup Using Bash Alias

```
# From infrastructure/cdn-in-a-box
alias mydc="docker-compose "` \
    "-f $PWD/docker-compose.yml "` \
    "-f $PWD/docker-compose.expose-ports.yml "` \
    "-f $PWD/optional/docker-compose.vnc.yml "` \
    "-f $PWD/optional/docker-compose.vnc.expose-ports.yml "`
docker volume prune -f
mydc build
mydc kill
mydc rm -fv
mydc up
```

4. Connect with a VNC client to localhost port 5909.
5. When Traffic Portal becomes available, the Firefox within the VNC instance will subsequently be started.
6. An xterm with bash shell is also automatically spawned and minimized for convenience.

## Socks Proxy

Dante's socks proxy is an optional container that can be used to provide browsers and other clients the ability to resolve DNS queries and network connectivity directly on the tcnet bridged interface. This is very helpful when running the CDN-In-A-Box stack on OSX/Windows docker host that lacks network bridge/IP-forward support. Below is the basic procedure to enable the Socks Proxy support for CDN-in-a-Box:

1. Start the CDN-in-a-Box stack at least once so that the x.509 self-signed CA is created.
2. On the host, import and Trust the CA for your target Operating System. See [Trusting the Certificate Authority](#)
3. On the host, using either Firefox or Chromium, download the [FoxyProxy browser plugin](#) which enables dynamic proxy support via URL regular expression
4. Once FoxyProxy is installed, click the Fox icon on the upper right hand of the browser window, select *Options*
5. Once in Options Dialog, Click *Add New Proxy* and navigate to the General tab:
6. Fill in the General tab according to the table

Table 54: General Tab Values

Name	Value
Proxy Name	CIAB
Color	Green

7. Fill in the Proxy Details tab according to the table

Table 55: Proxy Details Tab Values

Name	Value
Manual Proxy Configuration	CIAB
Host or IP Address	localhost
Port	9080
Socks Proxy	checked
Socks V5	selected

8. Go to URL Patterns tab, click Add New Pattern, and fill out form according to

Table 56: URL Patters Tab Values

Name	Value
Pattern Name	CIAB Pattern
URL Pattern	*.test/*
Whitelist	selected
Wildcards	selected

9. Enable dynamic 'pre-defined and patterns' mode by clicking the fox icon in the upper right of the browser. This mode only forwards URLs that match the wildcard \*.test/\*

to the Socks V5 proxy.

10. On the docker host start up CDN-in-a-Box stack. It is recommended that this be done using a custom bash alias

### 45: CIAB Startup Using Bash Alias

```
# From infrastructure/cdn-in-a-box
alias mydc="docker-compose -f $PWD/docker-compose.yml -f
↳$PWD/optional/docker-compose.socksproxy.yml"
docker volume prune -f
mydc build
mydc kill
mydc rm -fv
mydc up
```

11. Once the CDN-in-a-box stack has started, use the aforementioned browser to access Traffic Portal via the socks proxy on the docker host.

### See also:

The official [Docker Compose documentation CLI reference](#) for complete instructions on how to pass service definition files to the `docker-compose` executable.

## Static Subnet

Since `docker-compose` will randomly create a subnet and it has a chance to conflict with your network environment, using static subnet is a good choice.

#### 46: CIAB Startup with Static Subnet

```
# From the infrastructure/cdn-in-a-box directory
alias mydc="docker-compose -f $PWD/docker-compose.yml -f $PWD/optional/
↳docker-compose.static-subnet.yml"
docker volume prune -f
mydc build
mydc up
```

## VPN Server

This container provides an OpenVPN service. It's primary use is to allow users and developers to easily access CIAB network.

### How to use it

1. It is recommended that this be done using a custom bash alias.

#### 47: CIAB Startup with VPN

```
# From infrastructure/cdn-in-a-box
alias mydc="docker-compose -f $PWD/docker-compose.yml -f
↳$PWD/docker-compose.expose-ports.yml -f $PWD/optional/
↳docker-compose.vpn.yml -f $PWD/optional/docker-compose.
↳vpn.expose-ports.yml"
mydc down -v
mydc build
mydc up
```

2. All certificates, keys, and client configuration are stored at `infrastructure/cdn-in-a-box/optional/vpn/vpnca`. You just simply change `REALHOSTIP` and `REALPORT` of `client.ovpn` to fit your environment, and then you can use it to connect to this OpenVPN server.

## The proposed VPN client

On Linux, we suggest `openvpn`. On most Linux distributions, this will also be the name of the package that provides it.

#### 48: Install openvpn on ubuntu/debian

```
apt-get update && apt-get install -y openvpn
```

On OSX, it only works with brew installed `openvpn` client, not the *OpenVPN GUI client*.

### 49: Install openvpn on OSX

```
brew install openvpn
```

If you want a GUI version of VPN client, we recommend [Tunnelblick](#).

## Private Subnet for Routing

Since `docker-compose` randomly creates a subnet, this container prepares 2 default private subnets for routing:

- 172.16.127.0/255.255.240.0
- 10.16.127.0/255.255.240.0

The subnet that will be used is determined automatically based on the subnet prefix. If the subnet prefix which `docker-compose` selected is 192. or 10., this container will select 172.16.127.0/255.255.240.0 for its routing subnet. Otherwise, it selects 10.16.127.0/255.255.240.0.

Of course, you can decide which routing subnet by supplying the environment variables `PRIVATE_NETWORK` and `PRIVATE_NETMASK`.

## Pushed Settings

Pushed settings are shown as follows:

- DNS
- A routing rule for the CIAB subnet

---

**Note:** It will not change your default gateway. That means apart from CDN in a Box traffic and DNS requests, all other traffic will use the standard interface bound to the default gateway.

---

## Grafana

This container provides a Grafana service. It's an open platform for analytics and monitoring. This container has prepared necessary *datasources* and *scripted dashboards*. Please refer to [Configuring Grafana](#) for detailed Settings.

## How to start it

It is recommended that this be done using a custom bash alias.

### 50: CIAB Startup with Grafana

```
# From infrastructure/cdn-in-a-box
alias mydc="docker-compose -f $PWD/docker-compose.yml -f $PWD/optional/
→docker-compose.grafana.yml -f $PWD/optional/docker-compose.grafana.
→expose-ports.yml"
mydc down -v
mydc build
mydc up
```

Apart from start Grafana, the above commands also expose port 3000 for it.

## Check the charts

There are some “scripted dashboards” that can show easily comprehended charts. The data displayed on different charts is controlled using query string parameters.

- `https://Grafana Host/dashboard/script/traffic_ops_cachegroup.js?which=Cache Group name`. The query string parameter `which` in this particular URL should be the *Cache Group*. With default CIAB (CDN-in-a-Box) data, it can be filled in with **CDN\_in\_a\_Box\_Edge** or **CDN\_in\_a\_Box\_Edge**.
- `https://Grafana Host/dashboard/script/traffic_ops_deliveryservice.js?which=XML ID`. The query string parameter `which` in this particular URL should be the *xml\_id* of the desired *Delivery Service*.
- `https://Grafana Host/dashboard/script/traffic_ops_server.js?which=hostname`. The query string parameter `which` in this particular URL should be the **hostname** (not **FQDN**). With default CIAB data, it can be filled in with **edge** or **mid**.

## Debugging

See *Debugging inside CDN-in-a-Box*.

## Ansible-based Lab Deployment

### Audience for Ansible-based Lab Deployment

- DevOps engineers building more production-like lab environments
- DevOps engineers building an initial greenfield production environment
- Developers who need test environments that cannot be modeled on local resources

### Ansible-based Lab Creation

The scope of the Ansible work presented is a set of generic roles an implementor could use to expose the common installation and configuration tasks of each component. Additionally, it provides some suggestions and sample scaffolding on how to divide the full end-to-end lab construction.

#### Why Ansible?

There are many excellent tools to facilitate application installation and configuration. Ansible is a leading open-source tool in this marketplace backed by major corporate sponsorship and adoption. Most importantly it facilitates the abstractions desired without creating technological conflicts with existing infrastructure management solutions.

#### What about Security?

Each organization should review the instructions being performed in each Ansible playbook to determine if they satisfy their security requirements. Additionally, each implementor should select and implement their secret store of choice such as the built-in `ansible-vault` or a more advanced secret-as-a-service solution for any sensitive variables.

### Lab Implementation Concepts

The basic idea is to separate responsibilities to allow each implementation to use the tools/technologies that are already in use within their organizations.

#### Provisioning Layer

The provisioning layer deals with the lowest levels of compute/network/storage/load balancer resources. Its objective is to bring systems from nothingness to a functional operating system (at least minimally). Additionally, it is responsible for setting up proper DNS for each system, CDN components, and DNS NS record delegations.

Since DNS is a part of this layer, this unfortunately necessitates a small number of CDN concepts being present in the provisioning configuration. It is expected that upon completion of this layer, a compatible Ansible inventory file is generated and placed in the lab's Ansible inventory directory.



## The Provisioning Layer Output

Ansible supports inventory files in several formats such as JSON, YAML, INI, or TOML. An example output is located at `infrastructure/ansible/sample.lab/inventory/provisioning.inventory`.

When creating systems, each will probably be designated for some particular component in the CDN. Ansible groups and hostvars specifying the component name must be provided and will be mapped to server types later on in the dataset loader.

Like the systems being allocated to a particular component name, they must also be assigned to a particular DNS NS Delegation name where relevant or the ALL cdn as needed by the component. The NS Delegation name is what you would find in DNS for Traffic Router to be authoritative for. In this example it's MKGA rather than the CDN Name which is Kabletown2.0.

It is not recommended that an *origin server* be used by more than one *Delivery Service* or it could lead to inconsistent behavior and conflated log data at the *Mid-tier*. As a workaround to this it is better for a lab to create DNS CNAME for each *Delivery Service* pointing at a particular *origin server* and return that set of names as a CSV hostvar `ds_names` on each *origin server*. These names will later be translated to additional inventory hosts used only for the creation of server objects in Traffic Ops and assignment to *Delivery Services*.

## Steady-state OS Layer

The steady-state layer deals with the adaptation of a generic OS image to something that meets the organization's requirements. For a CDN lab, it is also important to consider the deployment of necessary SSL data to known locations.

## Application Layer

This is the primary area for contribution within Apache Traffic Control. Ansible has built-in variable precedence and role inclusion. Conceptually these playbooks are split into implementation specific driver playbooks versus a generic core.

## Implementation Specific Driver Playbook

Generally, this should handle some basic lab variable plumbing as well as any implementation specific tasks not generally useful to other ATC implementations. For example, an implementation might use an internal application as their application alerting engine which wouldn't make sense to others. Sticking with the example of an application alerting engine, even if it were an open implementation such as Nagios it still probably wouldn't be appropriate as not all implementations make use of that tool and maintaining it after contribution would incur ongoing costs.

### Generic Core Playbook

This is the important piece for collaboration as it's based on the question, "Is this task/feature/function something all ATC implementations would benefit from?". Typically, the yes answers involve exposing or simplifying application functionality on the part of the lab environment maintainer or developer. Generally, the default values of a generic core role match or improve upon those present inside the RPM of the software.

### Lab Implementation Layout

#### Ansible variable hierarchy

This is a topic better covered by [Ansible documentation](#), but the short version to keep in mind if you follow the sample lab design and markdown readme is:

(Highest precedence) CLI → Lab Vault → Lab Vars → Playbook Vars → Task Vars → Role Defaults (Lowest precedence)

Each of the generic core roles uses a prefix on its variables to avoid collision, and to make life easier it's recommended that you map them on the associated import role task variables. This makes keeping track of what variables were intentionally overwritten from the role defaults clearer.

### Sample Driver Playbooks

There are a few sample playbooks located at [infrastructure/ansible/](#). As an implementor develops their implementation specific driver playbooks they should go here.

- [infrastructure/ansible/steady-state.yml](#) is an example of a playbook that can deal with generating SSL certificates and distributing them across a lab as well as dynamically generating a secondary inventory file with additional data.
- [infrastructure/ansible/sample.driver.playbook.yml](#) shows a general skeleton to leverage when building out your implementation's driver playbooks around the generic core roles.
- [infrastructure/ansible/influxdb\\_relay.yml](#) shows a more advanced example of a driver playbook that involves querying Traffic Ops for supplementary information in a secondary play.
- [infrastructure/ansible/test.urls.yml](#) is just a simple playbook to query a known asset list on all delivery service urls to ensure a basic 200 http response for every asset and url.

## The Lab directory

A simple scaffold for a lab directory is included at `infrastructure/ansible/sample.lab`. This directory should encapsulate all pieces that make one environment unique from another. Ideally making new environments is as simple as copy-paste this directory and tweak the variables desired inside.

- The `infrastructure/ansible/sample.lab/ansible` subdirectory should be used to hold variables specific to a particular lab in either `vars.yml` or an encrypted Ansible vault
- The `infrastructure/ansible/sample.lab/inventory` directory is where it's recommended for your provisioning layer to drop a valid Ansible inventory file describing what was allocated. When using Ansible, it's important to point the inventory source to this directory so that it will merge all available inventory files together for you.
- The `infrastructure/ansible/sample.lab/out/ssl` directory is generated with the first run of the lab and holds your local copy of the lab SSL data. The out directory is also handy for holding temporary data from the provisioning or steady-state layers to help triage failures.
- The docker and docker-compose related files are present as an optional wrapper for Linux hosts (doesn't work on OSX) around all the lab plumbing dependencies for Ansible. This is particularly handy for automated systems who perform regular redeployments such as in a CI/CD tool.
- `infrastructure/ansible/sample.lab/manual.run.sh` is a scaffold for the entry-point for performing a lab rebuild from your local system.

## Gilt

Traditionally when distributing application playbooks for Ansible, many people use the built-in Ansible Galaxy repository. There is a design limitation to the Ansible Galaxy though in that one git repository may only contain one role. In the case of Apache Traffic Control, there are many components each with their own roles. At the end of the day, the generic core roles must exist in a valid Ansible role directory location. There are many solutions to this problem, but one of the better and easier once that's been run across is using the 3<sup>rd</sup>-party tool [Gilt](#). As another alternative you can simply extract the roles from an Apache Traffic Control (ATC) source tarball from a build.

## The Roles directory

The generic core roles for each component live at `infrastructure/ansible/roles`. Each role contains a README.md with more information, but this is not a replacement for existing documentation on the components themselves. It's very useful to still review the Administrator's Guide in the documentation as you develop your implementation around the component's generic core.

If you're attempting to optimize the wallclock time needed to deploy all the components in parallel, they should be installed like the following:

### Ansible Bonuses

These roles don't require a lab environment to be useful to operations (ops) teams.

#### The `to_api` role

When reviewing the generic core roles, you'll notice that `infrastructure/ansible/roles/to_api` is a little different and doesn't map to an ATC component. This role was developed for Ops teams to integrate around daily workflows if desired.

### Using Traffic Ops as an Ansible Dynamic Inventory source

`infrastructure/ansible/dynamic.inventory` contains a python script that is compatible with Ansible as a dynamic inventory. It leverages the python native client in ATC to expose lots of Traffic Ops server related data to the operator to make powerful and precise Ansible host patterns without the need of maintaining static files.

### 3.1.3 Traffic Portal

Traffic Portal is only supported on CentOS Linux distributions version 7.x and 8.x. It runs on `NodeJS` and requires version 12 or higher.

#### Installing

1. Build the Traffic Portal RPM using the instructions in *Building Traffic Control*.
2. Copy the Traffic Portal RPM to your server
3. Install NodeJS. This can be done by building it from source, installing with `yum(8)` if it happens to be in your available repositories (at version 12+), or using the NodeSource setup script.

51: Installing NodeJS using the NodeSource Setup Script

```
curl --silent --location https://rpm.nodesource.com/setup_
↪12.x | sudo bash -
```

4. Install the Traffic Portal RPM with `yum(8)` or `rpm(8)` e.g. by running `yum install path/to/traffic_portal.rpm` as the root user or with `sudo(8)`.

## Configuring

Traffic Portal is primarily configured through three different files that affect different parts of its behavior. Those files are detailed in this section.

### `/etc/traffic_portal/conf/config.js`

This file controls the behavior of the Traffic Portal server. It is a JavaScript source file which **MUST** set `module.exports` to the value of a configuration object. Configuration objects have the following allowed properties.

---

**Tip:** If Traffic Portal is being upgraded, reconcile `config.js` with `config.js.rpmnew` and then delete `config.js.rpmnew`.

---

**api** The properties of this object control Traffic Portal’s interactions with the *Traffic Ops API*.

**base\_url** The **full** URL that points to the root of the *Traffic Ops API*, e.g. `https://trafficops.infra.ciab.test:443/api/` **not** `https://trafficops.infra.ciab.test:443` or `https://trafficops.infra.ciab.test:443/api` (trailing / required).

**files** The properties of this object describe system paths important to Traffic Portal (other than SSL-related files, which are kept in the `ssl` object).

**static** The directory where the built Traffic Portal front-end files are kept. In most cases, changing this from the default is unnecessary/discouraged.

**log** The properties of this object describe the logging behavior of Traffic Portal. If this property is missing, or is `null` (or other “falsey” values), then logging is done simply on STDOUT.

**stream** Defines a file location for Traffic Portal’s access logs. If this property is missing, or is `null` (or other “falsey” values), then logs will go to STDOUT instead.

**port** A port number that sets the port on which Traffic Portal will listen for insecure (HTTP) connections. If `useSSL` is `false`, requests made to this port will be redirected to use HTTPS on the `sslPort` instead of just serving content insecurely.

**reject\_unauthorized** A boolean that defines whether or not Traffic Portal will reject SSL certificates that fail to validate as trusted.

**Caution:** Setting this to `false` exposes Traffic Portal to security vulnerabilities such as man-in-the-middle attacks, and should *never* be done in a production setting.

**ssl** This object has properties that set the locations of SSL keys and certificates. Has no effect if `useSSL` is `false`.

**key** The file location of the SSL certificate private key.

**cert** The file location of the x509 SSL certificate that Traffic Portal will use.

**ca** The file locations of the full certificate chain for the certificate authority that signed the SSL key (in order).

**sslPort** A port number that sets the port on which Traffic Portal will listen for secure (HTTPS) connections. Has no effect if `useSSL` is `false`.

**useSSL** A boolean that defines whether or not the Traffic Portal instance will offer secure (HTTPS) connections.

**Caution:** Setting this to `false` can expose sensitive data such as authentication credentials. Do not *ever* do that in a production setting.

**timeout** This property defines the maximum time for which Traffic Portal will process requests before sending a timeout response. It can be either a number of milliseconds, or a duration string accepted by [the ms library](#).

**Warning:** Slow requests will continue to use CPU and memory, even though a response has already been sent to the client.

`/opt/traffic_portal/public/traffic_portal_properties.json`

- update `/opt/traffic_portal/public/traffic_portal_properties.json` (if Traffic Portal is being upgraded, reconcile `traffic_portal_properties.json` with `traffic_portal_properties.json.rpmnew` and then delete `traffic_portal_properties.json.rpmnew`)

`/opt/traffic_portal/public/resources/assets/css/custom.css`

This CSS (Cascading Style Sheets) file is provided for users to insert CSS to override the default styling of Traffic Portal.

## Configuring OAuth Through Traffic Portal

See *Configure OAuth Login*.

## Starting Traffic Portal

The Traffic Portal RPM comes with a `systemd(1)` unit file, so under normal circumstances Traffic Portal may be started with `systemctl(1)`.

52: Starting Traffic Portal

```
systemctl start traffic_portal
```

## Stopping Traffic Portal

The Traffic Portal RPM comes with a `systemd(1)` unit file, so under normal circumstances Traffic Portal may be stopped with `systemctl(1)`.

53: Stopping Traffic Portal

```
systemctl stop traffic_portal
```

## Using Traffic Portal

### Traffic Portal - Using

Traffic Portal is the official Traffic Control UI. Traffic Portal typically runs on a different machine than Traffic Ops, and works by using the Traffic Ops API. The following high-level items are available in the Traffic Portal menu.

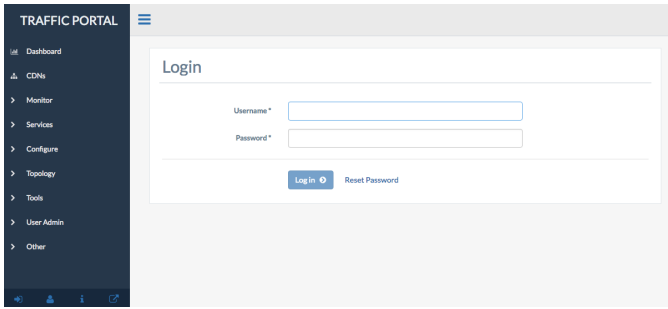


Fig. 4: Traffic Portal Start Page

Change Logs

At the top-right of every page is a bubble icon and badge count indicating the number of changes made to the CDN since the last viewing. Clicking on this expands a short list, with an option to *See All Change Logs*. Clicking on this will navigate to the “Changelog” page.

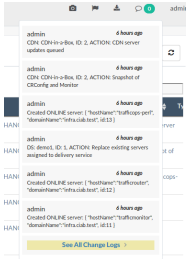


Fig. 5: The Changelog Dialog

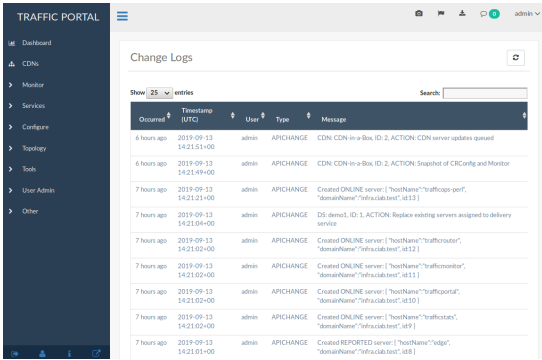


Fig. 6: The Full Change Logs Page



## Sidebar Bottom Buttons

At the bottom of the sidebar, a few action buttons may be found. At the far left, is the *Logout* button.



Fig. 7: The Logout Button

Clicking on this button will cause the current user session to end; further use of the Traffic Portal UI will require re-authentication. Next to this is the *User Profile* button.

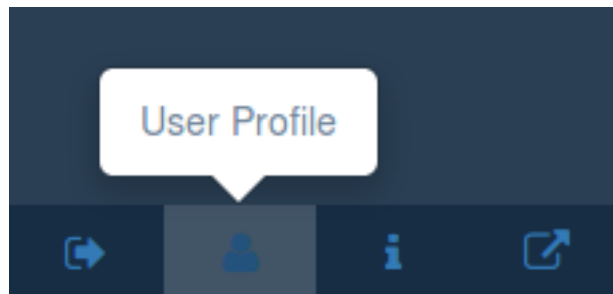


Fig. 8: The User Profile Button

Clicking on this button will take the user to his or her user profile page, where much of their information (contact, username etc.) is editable. The next button to the right is the *Release Info* button.

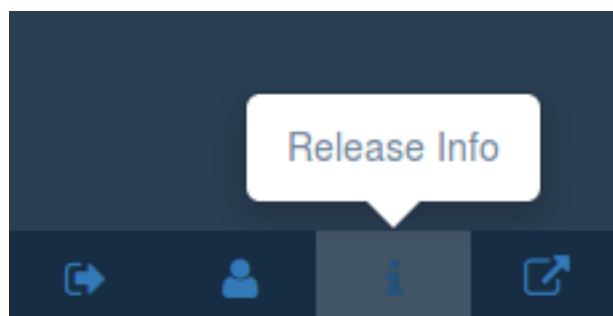


Fig. 9: The Release Info Button

When this button is clicked, a modal window will appear, displaying the Traffic Portal version, the date on which it was built, and the network location of the Traffic Ops server that serves the *Traffic Ops API* used by the Traffic Portal instance. The final button is the *Popout* button.

This button will open the current view in a new browsing context. Most browsers will implement this as a new tab, but can be configured instead to open them in a new window - check your browser's configuration and set it as desired.



Fig. 10: The Popout Button

## Dashboard

The Dashboard is the default landing page for Traffic Portal. It provides a real-time view into the main performance indicators of the CDNs managed by Traffic Control. It also displays various statistics about the overall health of your CDN.

**Current Bandwidth** The current bandwidth of all of your CDNs.

**Current Connections** The current number of connections to all of your CDNs.

**Healthy Caches** Displays the number of healthy *cache servers* across all CDNs. Click the link to view the healthy caches on the cache stats page.

**Unhealthy Caches** Displays the number of unhealthy *cache servers* across all CDNs. Click the link to view the unhealthy caches on the cache stats page.

**Online Caches** Displays the number of *cache servers* with ONLINE *Status*. Traffic Monitor will not monitor the state of ONLINE servers.

**Reported Caches** Displays the number of *cache servers* with REPORTED *Status*.

**Offline Caches** Displays the number of *cache servers* with OFFLINE *Status*.

**Admin Down Caches** Displays the number of caches with ADMIN\_DOWN *Status*.

Each component of this view is updated on the intervals defined in the `traffic_portal/app/src/traffic_portal_properties.json` configuration file.

## CDNs

A table of CDNs with the following columns:

**Name** The name of the CDN

**Domain** The CDN's TLD (Top-Level Domain)

**DNSSEC Enabled** 'true' if *DNSSEC* is enabled on this CDN, 'false' otherwise.

CDN management includes the ability to (where applicable):

- create a new CDN
- update an existing CDN

- delete an existing CDN
- *Queue Updates* on all servers in a CDN, or clear such updates
- Compare CDN *Snapshots*
- create a CDN *Snapshot*
- manage a CDN's DNSSEC keys
- manage a CDN's *Federations*
- view *Delivery Services* of a CDN
- view CDN *Profiles*
- view servers within a CDN

## Monitor

The *Monitor* section of Traffic Portal is used to display statistics regarding the various *cache servers* within all CDNs visible to the user. It retrieves this information through the *Traffic Ops API* from Traffic Monitor instances.

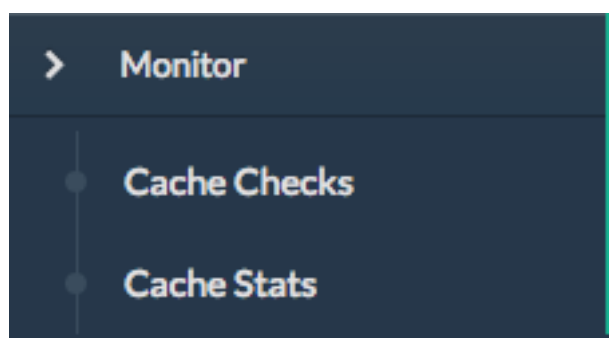


Fig. 11: The 'Monitor' Menu

## Cache Checks

A real-time view into the status of each *cache server*. The *Monitor* → *Cache Checks* page is intended to give an overview of the caches managed by Traffic Control as well as their status.

**Warning:** Several of these columns may be empty by default - particularly in the *CDN in a Box* environment - and require *Traffic Ops Extensions* to be installed/enabled/configured in order to work.

**Hostname** The (short) hostname of the *cache server*

**Profile** The *Name* of the *Profile* used by the *cache server*

**Status** The *Status* of the *cache server*

**See also:**

*Health Protocol*

**UPD** Displays whether or not this *cache server* has configuration updates pending

**RVL** Displays whether or not this *cache server* (or one or more of its *parents*) has content invalidation requests pending

**ILO** Indicates the status of an iLO (Integrated Lights-Out) interface for this *cache server*

**10G** Indicates whether or not the IPv4 address of this *cache server* is reachable via ICMP “pings”

**FQDN** DNS check that matches what the DNS servers respond with compared to what Traffic Ops has configured

**DSCP** Checks the DSCP value of packets received from this *cache server*

**10G6** Indicates whether or not the IPv6 address of this *cache server* is reachable via ICMP “pings”

**MTU** Checks the MTU by sending ICMP “pings” from the Traffic Ops server

**RTR** Checks the reachability of the *cache server* from the CDN’s configured Traffic Routers

**CHR** Cache-Hit Ratio (percent)

**CDU** Total Cache-Disk Usage (percent)

**ORT** Uses the *ORT* script on the *cache server* to determine if the configuration in Traffic Ops matches the configuration on *cache server* itself. The user as whom this script runs must have an SSH key on each server.

## Cache Stats

A table showing the results of the periodic *Check Extensions* that are run. These can be grouped by *Cache Group* and/or *Profile*.

**Profile** *Name* of the *Profile* applied to the Edge-tier or Mid-tier *cache server*, or the special name “ALL” indicating that this row is a group of all *cache servers* within a single *Cache Group*

**Host** ‘ALL’ for entries grouped by *Cache Group*, or the hostname of a particular *cache server*

**Cache Group** *Name of the Cache Group* to which this server belongs, or the name of the *Cache Group* that is grouped for entries grouped by *Cache Group*, or the special name “ALL” indicating that this row is an aggregate across all *Cache Groups*

**Healthy** True/False as determined by Traffic Monitor

**See also:**

*Health Protocol*

**Status** Status of the *cache server* or *Cache Group*

**Connections** Number of currently open connections to this *cache server* or *Cache Group*

**MbpsOut** Data flow rate outward from the CDN (toward client) in Megabits per second

## Services

*Services* groups the functionality to modify *Delivery Services* - for those users with the necessary permissions - or make *Delivery Service Requests* for such changes - for users without necessary permissions. Delivery Services can also be grouped by *Service Category*.

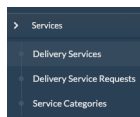


Fig. 12: The 'Services' Menu

## Delivery Services

This page contains a table displaying all *Delivery Services* visible to the user as determined by their *Tenant*.

XML ID (Key)	Tenant	Type	Active	Protocol
testqa	test and utility	HTTP	true	HTTP

Fig. 13: Table of Delivery Services

Use the *Select Columns* menu to select the delivery service columns to view and search. Columns can also be rearranged using drag-and-drop. Available delivery service columns include:

- *Active* (visible by default)
- *Anonymous Blocking*
- *CDN* (visible by default)
- *Check Path*

- *Consistent Hashing Query Parameters*
- *Consistent Hashing Regular Expression*
- *Deep Caching*
- *Display Name*
- *DNS Bypass CNAME*
- *DNS Bypass IP*
- *DNS Bypass IPv6*
- *DNS Bypass TTL*
- *DNS TTL*
- *DSCP* (visible by default)
- *Edge Header Rewrite Rules*
- *Fair-Queuing Pacing Rate Bps*
- *First Header Rewrite Rules*
- *Geo Limit*
- *Geo Limit Countries*
- *Geo Limit Redirect URL*
- *Geolocation Provider*
- *Geo Miss Default Latitude*
- *Geo Miss Default Longitude*
- *Global Max Mbps*
- *Global Max TPS*
- *HTTP Bypass FQDN*
- *Info URL*
- *Initial Dispersion*
- *Inner Header Rewrite Rules*
- *IPv6 Routing Enabled* (visible by default)
- *Last Header Rewrite Rules*
- *Long Description*
- *Max DNS Answers*
- *Max Origin Connections*
- *Mid Header Rewrite Rules*
- *Origin Shield*

- *Origin Server Base URL* (visible by default)
- *Profile*
- *Protocol* (visible by default)
- *Query String Handling* (visible by default)
- *Range Request Handling*
- *Range Slice Request Block Size*
- *Raw Remap Text*
- *Regex Remap Expression*
- *Regional Geoblocking*
- *Routing Name*
- *Signing Algorithm* (visible by default)
- *Tenant* (visible by default)
- *Topology*
- *Traffic Router Additional Response Headers*
- *Traffic Router Log Request Headers*
- *Type* (visible by default)
- *Use Multi-Site Origin Feature*
- *xml\_id* (visible by default)

*Delivery Service* management includes the ability to (where applicable):

- Create a new *Delivery Service*
- Clone an existing *Delivery Service*
- Update an existing *Delivery Service*
- Delete an existing *Delivery Service*
- Compare *Delivery Services*
- Manage *Delivery Service* SSL keys
- Manage *Delivery Service* URL signature keys
- Manage *Delivery Service* URI signing keys
- Manage *Delivery Service* invalidation requests
- Manage *Delivery Service* origins
- Manage *Delivery Service* regular expressions
- *Manage delivery service required server capabilities*
- Manage *Delivery Service* server assignments

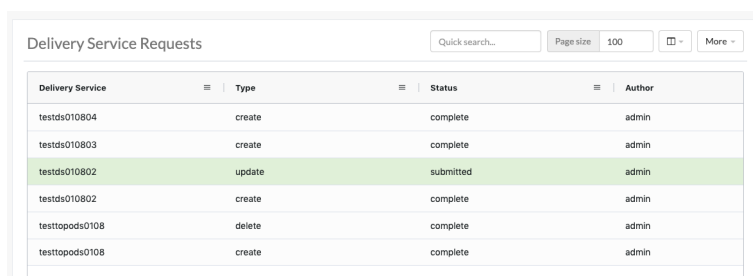
- Manage *Delivery Service* steering targets
- Manage *Delivery Service* static DNS records within a *Delivery Service* subdomain
- Test *Consistent Hashing Patterns*

**See also:**

*Configuring Static DNS Entries*

## Delivery Service Requests

If enabled in the `traffic_portal_properties.json` configuration file, all *Delivery Service* changes (create, update and delete) are captured as a *Delivery Service Request* and must be reviewed before fulfillment/deployment.



Delivery Service	Type	Status	Author
testds010804	create	complete	admin
testds010803	create	complete	admin
testds010802	update	submitted	admin
testds010802	create	complete	admin
testtopods0108	delete	complete	admin
testtopods0108	create	complete	admin

Fig. 14: Table of Delivery Service Requests

*Delivery Service*: A unique string that identifies the *Delivery Service* with which the request is associated. This unique string is also known (and often referred to within documentation and source code) as a *Delivery Service key* or 'XML ID'/'xml\_id'/'xmlid'.  
*Type*: The type of *Delivery Service Request*: 'create', 'update', or 'delete' according to what was requested.  
*Status*: The status of the *Delivery Service Request*. Has the following possible values:

**draft** The *Delivery Service Request* is *not* ready for review and fulfillment

**submitted** The *Delivery Service Request* is ready for review and fulfillment

**rejected** The *Delivery Service Request* has been rejected and cannot be modified

**pending** The *Delivery Service Request* has been fulfilled but the changes have yet to be deployed

**complete** The *Delivery Service Request* has been fulfilled and the changes have been deployed

**Author** The user responsible for creating the *Delivery Service Request*

**Assignee** The user responsible for fulfilling the *Delivery Service Request*. Currently, the operations role or above is required to assign *Delivery Service Requests*

**Last Edited By** The last user to edit the *Delivery Service Request*

**Created** Relative time indicating when the *Delivery Service Request* was created



**Actions** Actions that can be performed on a *Delivery Service Request*. The following actions are provided:

**fulfill** Implement the changes captured in the *Delivery Service Request*

**reject** Reject the changes captured in the *Delivery Service Request*

**delete** Delete the *Delivery Service Request*

*Delivery Service Request* management includes the ability to (where applicable):

- create a new *Delivery Service Request*
- update an existing *Delivery Service Request*
- delete an existing *Delivery Service Request*
- update the status of a *Delivery Service Request*
- assign a *Delivery Service Request*
- reject a *Delivery Service Request*
- fulfill a *Delivery Service Request*
- complete a *Delivery Service Request*

**See also:**

*Delivery Service Requests*

## Configure

Interfaces for managing the various components of Traffic Control and how they interact are grouped under *Configure*.

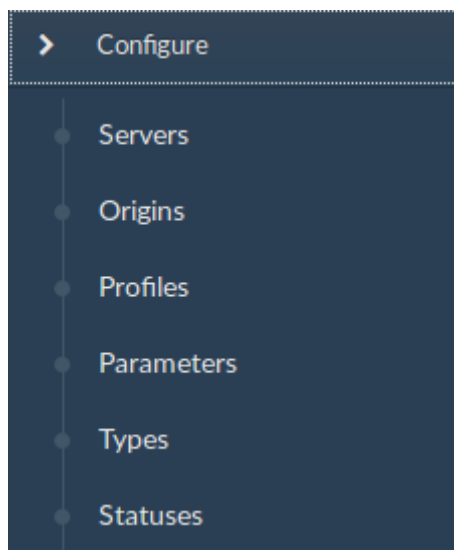


Fig. 15: The ‘Configure’ Menu

## Servers

A configurable table of all servers (of all kinds) across all *Delivery Services* and CDNs visible to the user.

Host	Domain	CDN	Network IP	Select Columns
foo	bar.com	Kabletown2.0	1.1.1.1	<input type="checkbox"/> Cache Group <input checked="" type="checkbox"/> CDN <input checked="" type="checkbox"/> Domain <input type="checkbox"/> Hash ID <input checked="" type="checkbox"/> Host <input type="checkbox"/> HTTPS Port <input type="checkbox"/> ID <input type="checkbox"/> ILO IP Address <input type="checkbox"/> ILO IP Gateway <input type="checkbox"/> ILO IP Netmask

Fig. 16: Table of Servers

Use the *Quick Search* to search across all table columns or the column filter to apply a more powerful filter to individual columns. Use the *Select Columns* menu to select the server columns to view. Columns can also be rearranged using drag-and-drop. Available server columns include:

**Cache Group** [Visible by default] The *Name of the Cache Group* to which this server belongs

**CDN** [Visible by default] The name of the CDN to which the server belongs

**Domain** [Visible by default] The domain part of the server's FQDN

**Hash ID** The identifier of the server used in Traffic Router's consistent hashing algorithm.

**Host** [Visible by default] The (short) hostname of the server

**HTTPS Port** The port on which the server listens for incoming HTTPS connections/requests

**ID** An integral, unique identifier for this server

**ILO IP Address** The IPv4 address of the server's ILO service

**See also:**

[Hewlett Packard ILO Wikipedia Page](#)

**ILO IP Gateway** The IPv4 gateway address of the server's ILO service

**ILO IP Netmask** The IPv4 subnet mask of the server's ILO service

**ILO Username** The user name for the server's ILO service

**Interface Name** The name of the primary network interface used by the server

**IPv6 Address** [Visible by default] The IPv6 address and subnet mask of interfaceName

- IPv6 Gateway** The IPv6 address of the gateway used by `interfaceName`
- Last Updated** The date and time at which this server description was last modified
- Mgmt IP Address** The IPv4 address of some network interface on the server used for ‘management’
- Mgmt IP Gateway** The IPv4 address of a gateway used by some network interface on the server used for ‘management’
- Mgmt IP Netmask** The IPv4 subnet mask used by some network interface on the server used for ‘management’
- IPv4 Gateway** The IPv4 address of the gateway used by `interfaceName`
- IPv4 Address** [Visible by default] The IPv4 address of `interfaceName`
- Network MTU** The Maximum Transmission Unit (MTU) to configured on `interfaceName`
- IPv4 Subnet** The IPv4 subnet mask used by `interfaceName`
- Offline Reason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status
- Phys Location** The name of the physical location where the server resides
- Profile** [Visible by default] The *Name* of the *Profile* used by this server
- Rack** A string indicating “server rack” location
- Reval Pending** [Visible by default] A boolean value represented as a clock (content invalidation/revalidation is pending) or green check mark (content invalidation/revalidation is not pending)
- Router Hostname** The human-readable name of the router responsible for reaching this server’s interface
- Router Port** The human-readable name of the port used by the router responsible for reaching this server’s interface
- Status** [Visible by default] The *Status* of the server
- See also:**
- Health Protocol*
- TCP Port** The port on which this server listens for incoming TCP connections
- Type** [Visible by default] The name of the *Type* of this server
- Update Pending** [Visible by default] A boolean value represented as a clock (updates are pending) or green check mark (updates are not pending), typically to be acted upon by Traffic Ops ORT

Server management includes the ability to (where applicable):

- Create a new server
- Update an existing server

- Delete an existing server
- *Queue Updates* on a server, or clear such updates
- Update server status
- View server *Delivery Services*
- Clone *Delivery Service* assignments
- Assign *Delivery Services* to server(s)
- *Manage server capabilities assigned to a server*

## Origins

A table of all *Origins*. These are automatically created for the *Origins* served by *Delivery Services* throughout all CDNs, but additional ones can be created at will. The table has the following columns:

**Name** The name of the *Origin*. If this *Origin* was created automatically for a *Delivery Service*, this will be the *xml\_id* of that *Delivery Service*.

**Tenant** The name of the *Tenant* that owns this *Origin* - this is not necessarily the same as the *Tenant* that owns the *Delivery Service* to which this *Origin* belongs.

**Primary** Either `true` to indicate that this is the “primary” *Origin* for the *Delivery Service* to which it is assigned, or `false` otherwise.

**Delivery Service** The *xml\_id* of the *Delivery Service* to which this *Origin* is assigned.

**FQDN** The FQDN of the *Origin*.

**IPv4 Address** The *Origin*’s IPv4 address, if configured.

**IPv6 Address** The *Origin*’s IPv6 address, if configured.

**Protocol** The protocol this *Origin* uses to serve content. One of

- `http`
- `https`

**Port** The port on which the *Origin* listens for incoming HTTP(S) requests.

---

**Note:** If this field appears blank in the table, it means that a default was chosen for the *Origin* based on its Protocol - 80 for “http”, 443 for “https”.

---

**Coordinate** The name of the geographic coordinate pair that defines the physical location of this *Origin*. *Origins* created for *Delivery Services* automatically will **not** have associated Coordinates. This can be rectified on the details pages for said *Origins*

**Cachegroup** The *Name of the Cache Group* to which this *Origin* belongs, if any.

**Profile** The *Name* of a *Profile* used by this *Origin*.

*Origin* management includes the ability to (where applicable):

- create a new *Origin*
- update an existing *Origin*
- delete an existing *Origin*

## Profiles

A table of all *Profiles*. From here you can see *Parameters*, servers and *Delivery Services* assigned to each *Profile*. Each entry in the table has these fields:

**Name** The *Name* of the *Profile*

**Type** The *Type* of this *Profile*, which indicates the kinds of objects to which the *Profile* may be assigned

**Routing Disabled** The *Routing Disabled* setting of this *Profile*

**Description** This *Profile*'s *Description*

**CDN** The *CDN* to which this *Profile* is restricted. To use the same *Profile* across multiple CDNs, clone the *Profile* and change the clone's *CDN* field.

*Profile* management includes the ability to (where applicable):

- create a new *Profile*
- update an existing *Profile*
- delete an existing *Profile*
- clone a *Profile*
- export a *Profile*
- view *Profile Parameters*
- view *Profile Delivery Services*
- view *Profile* servers

## Parameters

This page displays a table of *Parameters* from all *Profiles* with the following columns:

**Name** The *Name* of the *Parameter*

**Config File** The *Config File* to which the *Parameter* belongs.

**Value** The *Value* of the *Parameter*.

**Secure** Whether or not the *Parameter* is *Secure*

**Profiles** The number of *Profiles* currently using this *Parameter*  
*Parameter* management includes the ability to (where applicable):

- create a new *Parameter*
- update an existing *Parameter*
- delete an existing *Parameter*
- view *Parameter Profiles*
- manage assignments of a *Parameter* to one or more *Profiles* and/or *Delivery Services*

## Types

*Types* group *Delivery Services*, servers and *Cache Groups* for various purposes. Each entry in the table shown on this page has the following fields:

**Name** The name of the *Type*

**Use In Table** States the use of this *Type*, e.g. `server` indicates this is a *Type* assigned to servers

**Description** A short, usually user-defined, description of the *Type*

*Type* management includes the ability to (where applicable):

- create a new *Type*
- update an existing *Type*
- delete an existing *Type*
- view *Delivery Services* assigned to a *Type*
- view servers assigned to a *Type*
- view *Cache Groups* assigned to a *Type*

## Statuses

This page shows a table of *Statuses* with the following columns:

**Name** The name of this *Status*

**Description** A short, usually user-defined, description of this *Status*

*Status* management includes the ability to (where applicable):

- create a new *Status*
- update an existing *Status*
- delete an existing *Status*
- view *Statuses*

## Topology

*Topology* groups views and functionality that deal with how CDNs and their Traffic Control components are grouped and distributed, both on a logical level as well as a physical level.

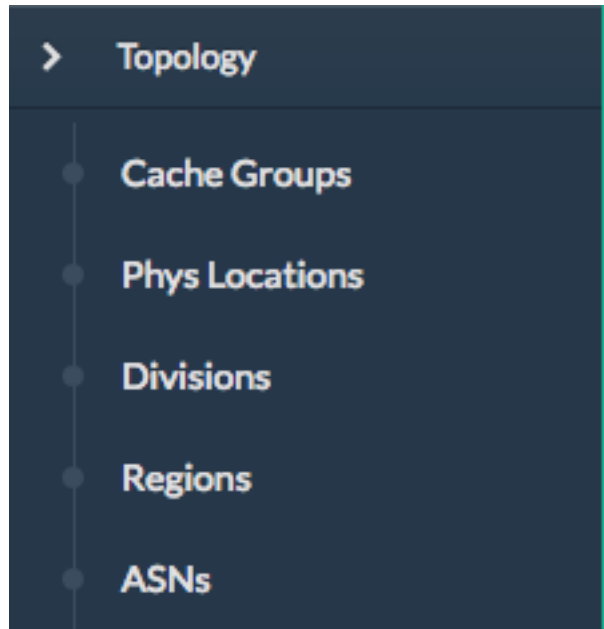


Fig. 17: ‘Topology’ Menu

## Cache Groups

This page is a table of *Cache Groups*, each entry of which has the following fields:

- Name** The full *Name of this Cache Group*
- Short Name** This *Cache Group’s Short Name*
- Type** This *Cache Group’s Type*
- Latitude** This *Cache Group’s Latitude*
- Longitude** This *Cache Group’s Longitude*

*Cache Group* management includes the ability to (where applicable):

- create a new *Cache Group*
- update an existing *Cache Group*
- delete an existing *Cache Group*
- *Queue Updates* for all servers in a *Cache Group*, or clear such updates
- view *Cache Group* ASNs

**See also:**

[The Wikipedia page on Autonomous System Numbers](#)

- view and assign *Cache Group Parameters*
- view *Cache Group* servers

### Coordinates

*Topology* → *Coordinates* allows a label to be given to a set of geographic coordinates for ease of use. Each entry in the table on this page has the following fields:

**Name** The name of this coordinate pair

**Latitude** The geographic latitude part of the coordinate pair

**Longitude** The geographic longitude part of the coordinate pair

Coordination management includes the ability to (where applicable):

- create a new coordinate pair
- update an existing coordinate pair
- delete an existing coordinate pair

### Phys Locations

A table of *Physical Locations* which may be assigned to servers and *Cache Groups*, typically for the purpose of optimizing client routing. Each entry has the following columns:

**Name** The full name of the *Physical Location*

**Short Name** A shorter, more human-friendly name for this *Physical Location*

**Address** The *Physical Location*'s street address (street number and name)

**City** The city within which the *Physical Location* resides

**State** The state within which the *Physical Location*'s city lies

**Region** The *Region* to which this *Physical Location* has been assigned

*Physical Location* management includes the ability to (where applicable):

- create a new *Physical Location*
- update an existing *Physical Location*
- delete an existing *Physical Location*
- view *Physical Location* servers



## Divisions

Each entry in the table of *Divisions* on this page has the following fields:

**Name** The name of the *Division*

*Division* management includes the ability to (where applicable):

- create a new *Division*
- delete an existing *Division*
- modify an existing *Division*
- view *Regions* within a *Division*

## Regions

Each entry in the table of *Regions* on this page has the following fields:

**Name** The name of this *Region*

**Division** The *Division* to which this *Region* is assigned

*Region* management includes the ability to (where applicable):

- create a new *Region*
- update an existing *Region*
- delete an existing *Region*
- view *Physical Locations* within a *Region*

## ASNs

Manage ASNs. Each entry in the table on this page has the following fields:

**ASN** The actual ASN

**Cache Group** The *Cache Group* to which this ASN is assigned

ASN management includes the ability to (where applicable):

- create a new ASN
- update an existing ASN
- delete an existing ASN

**See also:**

[Autonomous System \(Internet\) Wikipedia Page](#)

## Tools

*Tools* contains various tools that don't directly relate to manipulating Traffic Control components or their groupings.

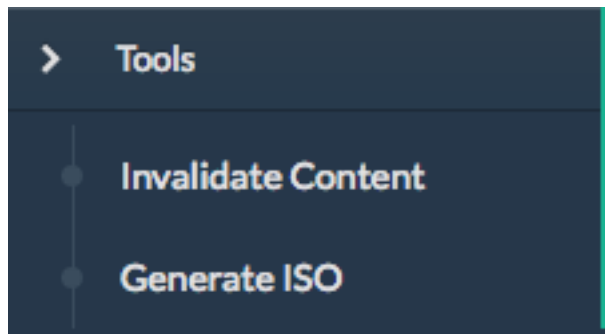


Fig. 18: The 'Tools' Menu

## Invalidate Content

Here, specific assets can be invalidated in all caches of a *Delivery Service*, forcing content to be updated from the origin. Specifically, this *doesn't* mean that *cache servers* will immediately remove items from their caches, but rather will fetch new copies whenever a request is made matching the 'Asset URL' regular expression. This behavior persists until the *Content Invalidation Job's* *TTL* expires.

**Warning:** This method forces *cache servers* to “re-validate” content, so in order to work properly the *Origin* needs to support revalidation according to section 4.3.2 of **RFC 7234**.

Each entry in the table on this page has the following fields:

**Delivery Service** The *Delivery Service* to which to apply this *Content Invalidation Job*

**Asset URL** A URL or regular expression which describes the asset(s) to be invalidated

**TTL (Hours)** A TTL (as a number of hours) over which the *Content Invalidation Job* shall remain active

**Start** An effective start time until which the *job* is delayed

**Expires** The date/time at which the *Content Invalidation Job* will end (effectively “Start” plus “TTL (Hours)”)

**Created By** The user name of the person who created this *Content Invalidation Job*

**Invalidation Type** The *Invalidation Type* of this *Content Invalidation Job*

Invalidate content includes the ability to (where applicable):

- create a new *Content Invalidation Job*

## Generate ISO

Generates a boot-able system image for any of the servers in the Servers table (or any server for that matter). Currently it only supports CentOS 7, but if you're brave and pure of heart you MIGHT be able to get it to work with other Unix-like Operating Systems. The interface is *mostly* self-explanatory, but here is a short explanation of the fields in that form.

### See also:

For instructions on setting up the Kickstart ISO generation files, see *Creating the CentOS Kickstart File*.

**Copy Server Attributes From** Optional. This option lets the user choose a server from the Traffic Ops database and will auto-fill the other fields as much as possible based on that server's properties

**OS Version** This list is populated by modifying the `osversions.json` file on the Traffic Ops server. This file maps OS names to the name of a directory under *kickstart.files.location* (`/var/www/files` by default).

**Hostname** The desired hostname of the resultant system

**Domain** The desired domain name of the resultant system

**DHCP** If this is 'no' the IP settings of the system must be specified, and the following extra fields will appear:

**IP Address** The resultant system's IPv4 address

**IPv6 Address** The resultant system's IPv6 address

**Network Subnet** The system's network subnet mask

**Network Gateway** The system's network gateway's IPv4 address

**IPv6 Gateway** The system's network gateway's IPv6 address

**Management IP Address** An optional IP address (IPv4 or IPv6) of a "management" server for the resultant system (e.g. for ILO)

**Management IP Netmask** The subnet mask (IPv4 or IPv6) used by a "management" server for the resultant system (e.g. for ILO) - only needed if the Management IP Address is provided

**Management IP Gateway** The IP address (IPv4 or IPv6) of the network gateway used by a "management" server for the resultant system (e.g. for ILO) - only needed if the Management IP Address is provided

**Management Interface** The network interface used by a "management" server for the resultant system (e.g. for ILO) - only needed if the Management IP Address is provided. Must not be the same as "Interface Name".

**Network MTU** The system's network's MTU. Despite being a text field, this can only be 1500 or 9000 - it should almost always be 1500

**See also:**

[The Maximum transmission unit Wikipedia Page](#)

**Disk for OS Install** The disk on which to install the base system. A reasonable default is `sda` (the `/dev/` prefix is not necessary)

**Root Password** The password to be used for the root user. Input is hashed using MD5 before being written to disk

**Confirm Root Password** Repeat the ‘Root Password’ to be sure it’s right

**Interface Name** Optional. The name of the resultant system’s network interface. Typical values are `bond0`, `eth4`, etc. If `bond0` is entered, a Link Aggregation Control Protocol bonding configuration will be written

**See also:**

[The Link aggregation Wikipedia Page](#)

**Stream ISO** If this is ‘yes’, then the download will start immediately as the ISO is written directly to the socket connection from Traffic Ops. If this is ‘no’, then the download will begin only *after* the ISO has finished being generated. For almost all use cases, this should be ‘yes’.

---

### Implementation Detail

Traffic Ops uses Red Hat’s *Kickstart* <[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/7/html/installation\\_guide/chap-kickstart-installations](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/chap-kickstart-installations)> to create these ISOs, so many configuration options not available here can be tweaked in the [Kickstart configuration file](#).

---

## User Admin

This section offers administrative functionality for users and their permissions.

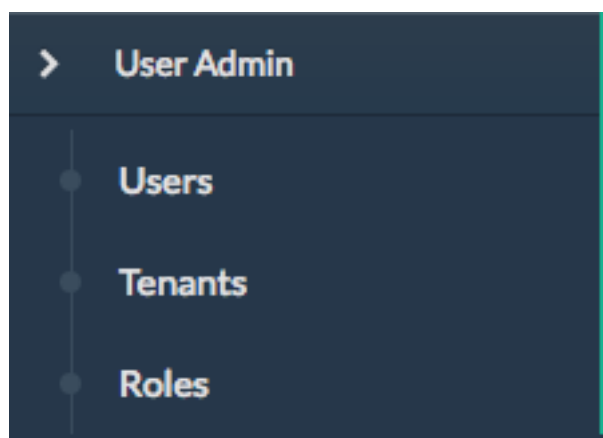


Fig. 19: The ‘User Admin’ Menu

## User

This page lists all the users that are visible to the user (so, for ‘admin’ users, all users will appear here). Each entry in the table on this page has the following fields:

**Full Name** The user’s full, real name

**Username** The user’s username

**Email** The user’s email address

**Tenant** The user’s *Tenant*

**Role** The user’s *Role*

User management includes the ability to (where applicable):

- register a new user
- create a new user
- update an existing user
- view *Delivery Services* visible to a user

---

**Note:** If OAuth is enabled, the username must exist both here as well as with the OAuth provider. A user’s rights are defined by the *Role* assigned to the user in Traffic Ops. Creating/deleting a user here will update the user’s *Role* but the user needs to be created/deleted with the OAuth provider as well.

---

## Tenants

Each entry in the table of *Tenants* on this page has the following entries:

**Name** The name of the *Tenant*

**Active** If ‘true’ users of this *Tenant* group are allowed to login and have active *Delivery Services*

**Parent** The parent of this *Tenant*. The default is the ‘root’ *Tenant*, which has no users.

*Tenant* management includes the ability to (where applicable):

- create a new *Tenant*
- update an existing *Tenant*
- delete an existing *Tenant*
- view users assigned to a *Tenant*
- view *Delivery Services* assigned to a *Tenant*

### Roles

Each entry in the table of *Roles* on this page has the following fields:

**Name** The name of the *Role*

**Privilege Level** The privilege level of this *Role*. This is a whole number that actually controls what a user is allowed to do. Higher numbers correspond to higher permission levels

**Description** A short description of the *Role* and what it is allowed to do

Role management includes the ability to (where applicable):

- view all *Roles*
- create new *Role*

---

**Note:** *Roles* cannot be deleted through the Traffic Portal UI

---

### Other

Custom menu items. By default, this contains only a link to the Traffic Control documentation.

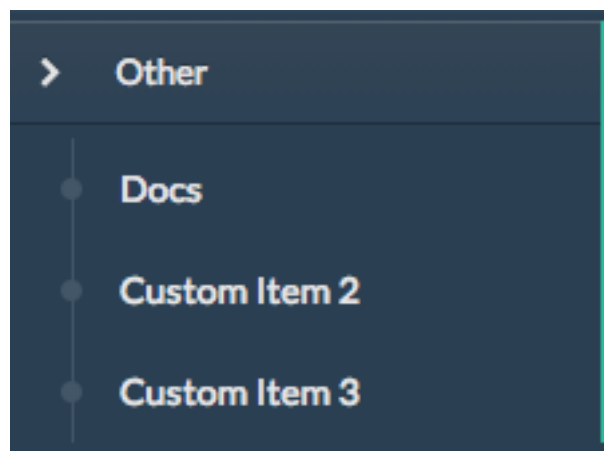


Fig. 20: The ‘Other’ Menu

### Docs

This is just a link to [the Traffic Control Documentation](#).

## Custom Menu Items

This section is configurable in the `traffic_portal_properties.json` configuration file, in the `customMenu` section.

### 3.1.4 Traffic Monitor Administration

#### Installing Traffic Monitor

The following are hard requirements requirements for Traffic Monitor to operate:

- CentOS 7 or later
- Successful install of Traffic Ops (usually on a separate machine)
- Administrative access to the Traffic Ops (usually on a separate machine)

These are the recommended hardware specifications for a production deployment of Traffic Monitor:

- 8 CPUs
- 16GB of RAM
- It is also recommended that you know the geographic coordinates and/or mailing address of the site where the Traffic Monitor machine lives for optimal performance

1. Enter the Traffic Monitor server into Traffic Portal

---

**Note:** For legacy compatibility reasons, the ‘Type’ field of a new Traffic Monitor server must be ‘RASCAL’.

---

2. Make sure the FQDN of the Traffic Monitor is resolvable in DNS.
3. Install Traffic Monitor, either from source or by installing a `traffic_monitor-version.string.rpm` package generated by the instructions in *Building Traffic Control* with `yum(8)` or `rpm(8)`
4. Configure Traffic Monitor according to *Configuring Traffic Monitor*
5. Start Traffic Monitor, usually by starting its `systemd(1)` service
6. Verify Traffic Monitor is running by e.g. opening your preferred web browser to port 80 on the Traffic Monitor host.

## Configuring Traffic Monitor

### Configuration Files

Traffic Monitor is configured via two JSON configuration files, `traffic_ops.cfg` and `traffic_monitor.cfg`, by default located in the `conf` directory in the install location.

#### `traffic_ops.cfg`

`traffic_ops.cfg` contains Traffic Ops connection information. Specify the URL, username, and password for the instance of Traffic Ops of which this Traffic Monitor is a member. However, this *also* sets some settings relating to the Traffic Monitor API server.

**cdnName** The name of the CDN to which this Traffic Monitor belongs. Used to fetch configuration and to determine which *cache servers* to monitor.

**certFile** The path to an SSL certificate file that corresponds to `keyFile` which will be used for Traffic Monitor's HTTPS API server.

**httpListener** Sets the address and port on which Traffic Monitor will listen for HTTP requests in the format *address:port*. If *address* is omitted, Traffic Monitor will listen on all available addresses.

**httpsListener** Sets the address and port on which Traffic Monitor will listen for HTTPS requests in the format *address:port*. If *address* is omitted, Traffic Monitor will listen on all available addresses. If not provided, `null`, or the empty string, Traffic Monitor will only serve HTTP, and `keyFile` and `certFile` are not used. If this is provided, the `httpListener` address will be used only to redirect clients to use HTTPS.

**insecure** A boolean that controls whether to validate the HTTPS certificate presented by the Traffic Ops server.

**keyFile** The path to an SSL key file that corresponds to `certFile` which will be used for Traffic Monitor's HTTPS API server.

**password** The password of the user identified by `username`.

**url** The URL at which Traffic Ops may be reached e.g. `"https://trafficops.infra.ciab.test"`.

**username** The username of the user as whom to authenticate with Traffic Ops.

**usingDummyTO** A boolean with no real effect. This value is used internally within the runtime of Traffic Monitor, and should never be set manually in its configuration file.

Deprecated since version ATCv7: The dependency on this field being valid will be removed in the future. It already has no effect.



## traffic\_monitor.cfg

`traffic_monitor.cfg` contains log file locations, as well as detailed application configuration variables such as processing flush times, initial poll intervals, and the polling protocols. Once started with the correct configuration, Traffic Monitor downloads its configuration from Traffic Ops, and any *Parameters* set on the Monitor's *Profile* that configure the same thing as a field in this configuration file will take precedence over said fields. The *Parameters* known to override configuration here are

- `tm.polling.interval`
- `health.polling.interval`
- `peers.polling.interval`
- `heartbeat.polling.interval`

Upon receiving this configuration, Traffic Monitor begins polling *cache server*s. Once every *cache server* has been polled, *Health Protocol* state is available via RESTful JSON endpoints and a web browser UI.

**cache\_polling\_protocol** Defines the internet protocol used to communicate with *cache servers*. This can be “ipv4only” to only allow IPv4 communication, “ipv6only” to only allow IPv6 communication, or “both” to alternate between each version. Default is “both”.

---

**Note:** both will poll IPv4 and IPv6 and report on availability based on if the respective IP addresses are defined on the server. So if only an IPv4 address is defined and the protocol is set to both then it will only show the availability over IPv4, but if both addresses are defined then it will show availability based on IPv4 and IPv6.

---

**crconfig\_backup\_file** The path to a file within which a backup of the most recently fetched CDN *Snapshot* will be stored. Default is `/opt/traffic_monitor/crconfig.backup`.

**crconfig\_history\_count** The number of historical CDN Snapshots to store, which can then be retrieved through the *Traffic Monitor APIs*. Default is 100.

**distributed\_polling** A boolean that controls whether *Distributed Polling* is enabled. Default is false.

**See also:**

The *Distributed Polling* section has more information on this setting.

**health\_flush\_interval\_ms** Defines an interval as a number of milliseconds on which Traffic Monitor will flush its collected health data such that it is made available through the *Traffic Monitor APIs*. Default is 200.

**See also:**

The *Stat and Health Flush Configuration* section has more information on this

setting.

**http\_polling\_format** A MIME-Type that will be sent in the *Accept* HTTP header in requests to *cache servers* for health and stats data. Default is *text/json* (not *application/json*).

**See also:**

The *HTTP Accept Header Configuration* section has more information on this setting.

**http\_timeout\_ms** Sets the timeout duration - in milliseconds - for all HTTP operations (both peer-polling and stat/health data polling). Default is 2000.

**log\_location\_access** A logfile location to which access logs will be written, or *null* to not log access events.<sup>1</sup> Default is *null*

**log\_location\_debug** A logfile location to which debug logs will be written, or *null* to not log debug messages.<sup>2</sup> Default is *null*

**log\_location\_error** A logfile location to which error logs will be written, or *null* to not log error messages.<sup>2</sup> Default is “*stderr*”.

**log\_location\_event** A logfile location to which event logs will be written, or *null* to not log events.<sup>2</sup> Default is “*stdout*”

**log\_location\_info** A logfile location to which informational logs will be written, or *null* to not log informational messages.<sup>2</sup> Default is *null*

**log\_location\_warning** A logfile location to which warning logs will be written, or *null* to not log warning messages.<sup>2</sup> Default is “*stdout*”

**max\_events** The maximum number of changes to stored aggregate data that should be retained at any one time. Default is 200.

**monitor\_config\_polling\_interval\_ms** The interval - in milliseconds - on which to poll Traffic Ops for this Traffic Monitor’s “monitoring configuration” as returned by *cdns/{{name}}/configs/monitoring*.

**peer\_optimistic\_quorum\_min** Specifies the minimum number of peers that must be available in order to participate in the optimistic health protocol. Default is zero.

**See also:**

The *Peering and Optimistic Quorum* section has more information on this setting.

**serve\_read\_timeout\_ms** Sets the timeout - in milliseconds - of the Traffic Monitor API server for reading incoming requests. Default is 10,000.

**serve\_write\_timeout\_ms** Sets the timeout - in milliseconds - of the Traffic Monitor API server for writing responses. Default is 10,000.

---

<sup>1</sup> These respect the rules and special string constants of [github.com/apache/trafficcontrol/lib/go-log](https://github.com/apache/trafficcontrol/blob/master/lib/go-log.go).

**short\_hostname\_override** Sets a hostname for the Traffic Monitor. It will behave as though this were its hostname, rather than the hostname actually reported by the operating system. If not provided, `null`, or the empty string, the Traffic Monitor will use the hostname provided by its host operating system. Default is the empty string.

**stat\_buffer\_interval\_ms** An interval - in milliseconds - for which to buffer collected stats before processing them. If this is not provided, `null`, or zero, then all stats will be processed immediately. Default is zero.

**See also:**

The *Stat and Health Flush Configuration* section has more information on this setting.

**stat\_flush\_interval\_ms** Defines an interval as a number of milliseconds on which Traffic Monitor will flush its collected stats data such that it is made available through the *Traffic Monitor APIs*. Default is 200.

**See also:**

The *Stat and Health Flush Configuration* section has more information on this setting.

**stat\_polling** A boolean that controls whether *cache servers* are polled for stats data. Default is `true`.

**See also:**

The *Optional Stat Polling* section has more information on this setting.

**static\_file\_dir** The directory within which Traffic Monitor will look for its web interface's static files. Default is `/opt/traffic_monitor/static`.

**tmconfig\_backup\_file** A file location to which a backup of the “monitoring configuration” as returned by `cdns/{{name}}/configs/monitoring` currently in use by Traffic Monitor will be written. Default is `/opt/traffic_monitor/tmconfig.backup`.

**traffic\_ops\_disk\_retry\_max** The number of times Traffic Monitor should attempt to log in to Traffic Ops before using its backup monitoring configuration and CDN Snapshot (if those exist). Default is 2.

**traffic\_ops\_max\_retry\_interval\_ms** Traffic Monitor will exponentially increase the amount of time it waits between attempts to log in to Traffic Ops each time it fails (up to a maximum number of times set by `traffic_ops_disk_retry_max`). This controls the maximum amount of time - in milliseconds - that this waiting duration will be. Default is 60,000.

**traffic\_ops\_min\_retry\_interval\_ms** Traffic Monitor will exponentially increase the amount of time it waits between attempts to log in to Traffic Ops each time it fails (up to a maximum number of times set by `traffic_ops_disk_retry_max`). This controls the minimum amount of time - in milliseconds - that this waiting duration will be. Default is 100.

### Optional Stat Polling

Traffic Monitor has the option to disable stat polling via the `stat_polling` (default: `true`) option in `traffic_monitor.cfg`. If set to `false`, Traffic Monitor will not poll caches for stats; it will only poll caches for health. This can be useful in lowering the amount of resources (CPU, bandwidth) used by Traffic Monitor while still allowing it to retain its core functionality (determining cache availability) via health polling alone. However, disabling stat polling also prevents some other ATC features from working properly (basically anything that requires stats data from caches, e.g. Traffic Stats data), so it should only be disabled when absolutely necessary.

### Distributed Polling

Traffic Monitor has the option to enable distributed polling via the `distributed_polling` (default: `false`) option in `traffic_monitor.cfg`. If set to `true`, Traffic Monitor groups will each poll their own disjoint subsets of the CDN. In order to enable this option, `stat_polling` must be disabled. In order to function properly, all Traffic Monitors in a CDN must have `distributed_polling` enabled; otherwise, the results are undefined.

---

**Note:** Traffic Monitors are said to be in the same “Traffic Monitor group” if they are in the same *Cache Group*.

---

Each Traffic Monitor in the same Traffic Monitor group (referred to as local peers) polls the same disjoint subset of the CDN and combines availability states with its local peers via the Health Protocol. This is similar to how Traffic Monitor behaves in its legacy, non-distributed mode except Traffic Monitor is not polling the entire CDN. In order to get availability data for the rest of the CDN, each Traffic Monitor also polls every other Traffic Monitor group in parallel (these are referred to as distributed peers). It does this by selecting one distributed peer per group at a time, cycling through each distributed peer in the group for subsequent polls in a round-robin manner.

Upon startup, Traffic Monitor will retrieve its config (either from TO or on-disk backup file), then begin polling the *Cache Groups* for which its Traffic Monitor group is responsible. Once it has polled the *Cache Groups*, it will start serving requests for `/publish/CrStates?raw` (the raw, uncombined health states of its local caches) and `/publish/CrStates?local` (the combined health states of its local caches derived from all Traffic Monitors in its group). Once Traffic Monitor has received `/publish/CrStates?local` responses from all other Traffic Monitor groups, it will start serving requests for `/publish/CrStates` (the combined health states of all caches in the CDN).

## Peering and Optimistic Quorum

As mentioned in the *Health Protocol* section of the *Traffic Monitor* overview, peering a Traffic Monitor with one or more other Traffic Monitors enables the optimistic health protocol. In order to leverage the optimistic quorum feature along with the optimistic health protocol, a minimum of three Traffic Monitors are required. The optimistic quorum feature allows a Traffic Monitor to withdraw itself from the optimistic health protocol when it loses connectivity to a number of its peers.

To enable the optimistic quorum feature, the `peer_optimistic_quorum_min` property in `traffic_monitor.cfg` should be configured with a value greater than zero that specifies the minimum number of peers that must be available in order to participate in the optimistic health protocol. If at any time the number of available peers falls below this threshold, the local Traffic Monitor will serve 503s whenever the aggregated, optimistic health protocol enabled view of the CDN's health is requested. Traffic Monitor will continue serving 503s and logging errors in `traffic_monitor.log` until the minimum number of peers are available. Once the minimum number of peers are available, the local Traffic Monitor can resume participation in the optimistic health protocol. This prevents negative states caused by network isolation of a Traffic Monitor from propagating to downstream components such as Traffic Router.

## Stat and Health Flush Configuration

The Monitor has a health flush interval, a stat flush interval, and a stat buffer interval. Recall that the monitor polls both stats and health. The health poll is so small and fast, a buffer is largely unnecessary. However, in a large CDN, the stat poll may involve thousands of *cache servers* with thousands of stats each, or more, and CPU may be a bottleneck.

The flush intervals, `health_flush_interval_ms` and `stat_flush_interval_ms`, indicate how often to flush stats or health, if results are continuously coming in with no break. This prevents starvation. Ideally, if there is enough CPU, the flushes should never occur. The default flush times are 200 milliseconds, which is suggested as a reasonable starting point; operators may adjust them higher or lower depending on the need to get health data and stop directing client traffic to unhealthy *cache servers* as quickly as possible, balanced by the need to reduce CPU usage.

The stat buffer interval, `stat_buffer_interval_ms`, also provides a temporal buffer for stat processing. Stats will not be processed except after this interval, whereupon all pending stats will be processed, unless the flush interval occurs as a starvation safety. The stat buffer and flush intervals may be thought of as a state machine with two states: the “buffer state” accepts results until the buffer interval has elapsed, whereupon the “flush state” is entered, and results are accepted while outstanding, and processed either when no results are outstanding or the flush interval has elapsed.

Note that this means the stat buffer interval acts as “bufferbloat,” increasing the average and maximum time a *cache server* may be down before it is processed and marked as unhealthy. If the stat buffer interval is non-zero, the average time a *cache server* may be down before being marked unavailable is half the poll time plus half the stat buffer interval, and the maximum time is the poll time plus the stat buffer interval. For example, if the stat poll time is 6 seconds, and the stat buffer interval is 4 seconds, the average time a *cache server* may be unhealthy before

being marked is  $\frac{6}{2} + \frac{4}{2} = 6$  seconds, and the maximum time is  $6 + 4 = 10$  seconds. For this reason, if operators feel the need to add a stat buffer interval, it is recommended to start with a very low duration, such as 5 milliseconds, and increase as necessary.

It is not recommended to set either flush interval to 0, regardless of the stat buffer interval. This will cause new results to be immediately processed, with little to no processing of multiple results concurrently. Result processing does not scale linearly. For example, processing 100 results at once does not cost significantly more CPU usage or time than processing 10 results at once. Thus, a flush interval which is too low will cause increased CPU usage, and potentially increased overall poll times, with little or no benefit. The default value of 200 milliseconds is recommended as a starting point for configuration tuning.

### HTTP Accept Header Configuration

The Accept header sent to caches for stat retrieval can be modified with the `http_polling_format` option. This is a string that will be inserted in to the Accept header of any requests. The default value is `text/json` which is the default value used by the `astats` plugin currently.

However newer versions of `astats` also support CSV output, which can have some CPU savings. To enable that format using `http_polling_format`: `"text/csv"` in `traffic_monitor.cfg` will set the Accept header properly.

### Troubleshooting and Log Files

Traffic Monitor log files are in `/opt/traffic_monitor/var/log/`.

### Extensions

Traffic Monitor allows extensions to its parsers for the statistics returned by *cache servers* and/or their plugins. The formats supported by Traffic Monitor by default are `astats`, `astats-dsnames` (which is an odd variant of `astats` that probably shouldn't be used), and `stats_over_http`. The format of a *cache server's* health and statistics reporting payloads must be declared on its *Profile* as the *health.polling.format Parameter*, or the default format (`astats`) will be assumed.

For instructions on how to develop a parsing extension, refer to the [github.com/apache/trafficcontrol/traffic\\_monitor/cache](https://github.com/apache/trafficcontrol/traffic_monitor/cache) package's documentation.

Importantly, though, a statistics provider *must* respond to HTTP GET requests over either plain HTTP or HTTPS (which is controlled by the *health.polling.url Parameter*), and it *must* provide the following statistics, or enough information to calculate them:

- System “loadavg” (only requires the one-minute value)

#### See also:

For more information on what “loadavg” is, refer to the `proc(5)` manual page.

- Input bytes, output bytes, and speeds for all monitored network interfaces

When using the `stats_over_http` extension this can be provided by the `system_stats` plugin which will inject that information in to the ATS stats which then get returned by `stats_over_http`. The `system_stats` plugin can be used with any custom implementations as it is already included and built with ATS when building with experimental-plugins enabled.

There are other optional and/or *Delivery Service*-related statistics that may cause Traffic Stats to not have the right information if not provided, but the above are essential for implementing *Health Protocol*.

### 3.1.5 Traffic Router Administration

#### Requirements

- CentOS 7 or later
- 4 CPUs
- 8GB of RAM
- Successful install of Traffic Ops (usually on another machine)
- Successful install of Traffic Monitor (usually on another machine)
- Administrative access to Traffic Ops

---

**Note:** Hardware requirements are generally doubled if *DNSSEC* is enabled

---

#### Installing Traffic Router

1. If no suitable *Profile* exists, create a new *Profile* for Traffic Router via the + button on the *Profiles* page in Traffic Portal

**Warning:** Traffic Ops will *only* recognize a *Profile* as assignable to a Traffic Router if its *Name* starts with the prefix `ccr-`. The reason for this is a legacy limitation related to the old name for Traffic Router (Comcast Cloud Router), and will (hopefully) be rectified in the future.

2. Enter the Traffic Router server into Traffic Portal on the *Servers* page (or via the *Traffic Ops API*), assign to it a Traffic Router *Profile*, and ensure that its status is set to ONLINE.
3. Ensure the FQDN of the Traffic Router is resolvable in DNS. This FQDN must be resolvable by the clients expected to use this CDN.
4. Install a Traffic Router server package, either from source or using a `traffic_router-version string.rpm` package generated using the instructions in *Building Traffic Control*.



Changed in version 3.0: As of version 3.0, Traffic Router depends upon a package called `tomcat`. This package should have been created when Traffic Router was built. If installing the `traffic_router` produces a dependency error, make sure that the `tomcat` package is available in an accessible `yum(8)` repository.

5. Edit `/opt/traffic_router/conf/traffic_monitor.properties` and specify the correct online Traffic Monitor(s) for your CDN.

**See also:**

*Traffic Router Configuration File Parameters*

**traffic\_monitor.properties** URL that should normally point to this file, e.g. `traffic_monitor.properties=file:/opt/traffic_router/conf/traffic_monitor.properties`

**traffic\_monitor.properties.reload.period** Period to wait (in milliseconds) between reloading this file, e.g. `traffic_monitor.properties.reload.period=60000`

6. Start Traffic Router. This is normally done by starting its `systemd(1)` service. `systemctl start traffic_router`, and test DNS lookups against that server to be sure it's resolving properly. with e.g. `dig` or `curl`. Also, because previously taken CDN *Snapshots* will be cached, they need to be removed manually to actually be reloaded. This file should be located at `/opt/traffic_router/db/cr-config.json`. This should be done before starting or restarting Traffic Router.

## 54: Starting and Testing Traffic Router

```
[root@trafficrouter /]# systemctl start traffic_router
[root@trafficrouter /]# dig @localhost mycdn.ciab.test

; <<>> DiG 9.9.4-RedHat-9.9.4-72.el7 <<>> @localhost mycdn.
↳ciab.test
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27109
;; flags: qr aa rd; QUERY: 1, ANSWER: 0, AUTHORITY: 1,↳
↳ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;mycdn.ciab.test.      IN  A

;; AUTHORITY SECTION:
mycdn.ciab.test.      30  IN  SOA trafficrouter.infra.ciab.
↳test. twelve_monkeys.mycdn.ciab.test. 2019010918 28800↳
↳7200 604800 30
```

(continues on next page)



(continued from previous page)

```
;; Query time: 28 msec
;; SERVER: ::1#53(::1)
;; WHEN: Wed Jan 09 21:27:57 UTC 2019
;; MSG SIZE rcvd: 104
```

7. Perform a CDN *Snapshot*.

---

**Note:** Once the *Snapshot* is taken, live traffic will be sent to the new Traffic Routers provided that their status has been set to **ONLINE**.

---

8. Ensure that the parent domain (e.g.: `cdn.local`) for the CDN's top level domain (e.g.: `ciab.cdn.local`) contains a delegation (Name Server records) for the new Traffic Router, and that the value specified matches the FQDN of the Traffic Router.

## Configuring Traffic Router

Changed in version 1.5: Many of the configuration files under `/opt/traffic_router/conf` are now only needed to override the default configuration values for Traffic Router. Most of the given default values will work well for any CDN. Critical values that must be changed are hostnames and credentials for communicating with other Traffic Control components such as Traffic Ops and Traffic Monitor. Pre-existing installations that store configuration files under `/opt/traffic_router/conf` will still be used and honored for Traffic Router 1.5 onward.

Changed in version 3.0: Traffic Router 3.0 has been converted to a formal Tomcat instance, meaning that is now installed separately from the Tomcat servlet engine. The Traffic Router installation package contains all of the Traffic Router-specific software, configuration and startup scripts including some additional configuration files needed for Tomcat. These new configuration files can all be found in the `/opt/traffic_router/conf` directory and generally serve to override Tomcat's default settings.

For the most part, the configuration files and *Parameters* used by Traffic Router are used to bring it online and start communicating with various Traffic Control components. Once Traffic Router is successfully communicating with Traffic Control, configuration should mostly be performed in Traffic Portal, and will be distributed throughout Traffic Control via CDN *Snapshot* process.

Table 57: Traffic Router Configuration File Parameters

Con-figu-ration File	Parame-ter Name	Description	Default Value
traf- fic_monitor.properties	traf- fic_monitor.bootstrap_hosts	Semicolon-delimited Traffic Monitor FQDNs with optional port numbers as necessary	N/A
	traf- fic_monitor.bootstrap_files	Use only the Traffic Monitors specified in local bootstrap files	false
	traf- fic_monitor.properties_monitor	Path to file: <i>traffic_monitor.properties</i> ; used internally to monitor the file for changes	/opt/ traffic_router/ traffic_monitor. properties
	traf- fic_monitor.properties_reload_period	The interval in milliseconds for Traffic Router to reparse and reload this configuration file	60000
dns.properties	dns.tcp.port	TCP port that Traffic Router will use for incoming DNS requests	53
	dns.tcp.backlog	Maximum length of the queue for incoming TCP connection requests	0
	dns.tcp.host	IP Address Traffic Router will listen on for incoming TCP DNS requests	0.0.0.0
	dns.udp.host	IP Address Traffic Router will listen on for incoming UDP DNS requests	0.0.0.0
	dns.udp.port	UDP port that Traffic Router will use for incoming DNS requests	53
	dns.max-threads	Maximum number of threads used to process incoming DNS requests	1000
	dns.queue-depth	Maximum number of threads allowed to queue when all workers threads are busy. To disable the queue, set to 0, or to allow an unlimited sized queue, set to -1.	1000
	dns.zones.dir	Path to automatically generated zone files for reference	/opt/ traffic_router/ var/ auto-zones
traf- fic_ops.properties	traf- fic_ops.username	Username with which to access the <i>Traffic Ops API</i> (must have the <i>admin Role</i> )	admin
	traf- fic_ops.password	Password for the user specified in <i>traffic_ops.username</i>	N/A
cache.properties	cache.geolocation_db_path	Full path to the local copy of a geographic IP mapping database (usually MaxMind's GeoIP2)	/opt/ traffic_router/ db/ GeoIP2-City. mmdb
	cache.geolocation_db_refresh_period	The interval in milliseconds for Traffic Router to wait between polling for changes to the GeoIP2 database	604800000
	cache.czmap_path	Full path to the local copy of the coverage zone file	/opt/ traffic_router/ db/czmap. json
186	cache.czmap_db_refresh_period	The interval in milliseconds for Traffic Router to	10800000

## The Traffic Router Profile

Much of a Traffic Router’s configuration can be obtained through the *Parameters* on its *Profile*. The *Parameters* of a Traffic Router’s *Profile* that have meaning (others are just ignored) are detailed in the *The Parameters of a Traffic Router Profile*.

<i>Name</i>	<i>Config File</i>	<i>Value Description</i>
location	dns.zone	Location to store the DNS zone files in
location	http-log4j2.xml	Location to find the log4j2.xml file for
location	dns-log4j2.xml	Location to find the dns-log4j2.xml file for
location	geolocation.properties	Location to find the log4j2.xml file for
CDN_name	rascal-config.txt	The human readable name of the CDN
CoverageZoneJsonURL	CRConfig.xml	The location (URL) where a <i>Coverage</i>
ecsEnable	CRConfig.json	Boolean value to enable or disable ED
geolocation.polling.url	CRConfig.json	The location (URL) where a geograph
geolocation.polling.interval	CRConfig.json	How often - in milliseconds - Traffic R
coveragezone.polling.interval	CRConfig.json	How often - in milliseconds - Traffic R
coveragezone.polling.url	CRConfig.json	The location (URL) where a <i>Coverage</i>
deepcoveragezone.polling.interval	CRConfig.json	How often - in milliseconds - Traffic R
deepcoveragezone.polling.url	CRConfig.json	The location (URL) where a <i>Deep Cov</i>
strip.special.query.params	CRConfig.json	If “true”, Traffic Router will strip its sp
tld.soa.expire	CRConfig.json	The value for the “expire” field the Tra
tld.soa.minimum	CRConfig.json	The value for the minimum field the T
tld.soa.admin	CRConfig.json	The DNS Start of Authority administrat
tld.soa.retry	CRConfig.json	The value for the “retry” field the Traf
tld.soa.refresh	CRConfig.json	The value for the “refresh” field the Tr
tld.ttls.NS	CRConfig.json	The TTL the Traffic Router DNS Serv
tld.ttls.SOA	CRConfig.json	The TTL the Traffic Router DNS Serv
tld.ttls.AAAA	CRConfig.json	The TTL the Traffic Router DNS Serv
tld.ttls.A	CRConfig.json	The TTL the Traffic Router DNS Serv
tld.ttls.DNSKEY	CRConfig.json	The TTL the Traffic Router DNS Serv
tld.ttls.DS	CRConfig.json	The TTL the Traffic Router DNS Serv
api.port	server.xml	The TCP port on which Traffic Router
api.cache-control.max-age	CRConfig.json	The value of the Cache-Control: m
api.auth.url	CRConfig.json	The URL of the authentication endpoi
consistent.dns.routing	CRConfig.json	Control whether <i>DNS-routed Delivery</i>
dnssec.enabled	CRConfig.json	Whether DNSSEC is enabled; this par
dnssec.zone.diffing.enabled	CRConfig.json	If DNSSEC is enabled, enabling this p
dnssec.rrsig.cache.enabled	CRConfig.json	If DNSSEC is enabled, enabling this p
dnssec.allow.expired.keys	CRConfig.json	Allow Traffic Router to use expired DN
dynamic.cache.primer.enabled	CRConfig.json	Allow Traffic Router to attempt to prin
dynamic.cache.primer.limit	CRConfig.json	Limit the number of permutations to p
edge.dns.limit	CRConfig.json	Integer that controls the default numbe
edge.dns.routing	CRConfig.json	Boolean flag to control whether edge r

<i>Name</i>	<i>Config File</i>	<i>Value</i> Description
edge.http.limit	CRConfig.json	Integer that controls the default number
edge.http.routing	CRConfig.json	Boolean flag to control whether edge r
keystore.maintenance.interval	CRConfig.json	The interval in seconds which Traffic I
keystore.api.url	CRConfig.json	The URL of the DNSSEC key manage
keystore.fetch.timeout	CRConfig.json	The timeout in milliseconds for reques
keystore.fetch.retries	CRConfig.json	The number of times Traffic Router wi
keystore.fetch.wait	CRConfig.json	The number of milliseconds Traffic Ro
signaturemanager.expiration.multiplier	CRConfig.json	Multiplier used in conjunction with a z
zonemanager.threadpool.scale	CRConfig.json	Multiplier used to determine the numb
zonemanager.cache.maintenance.interval	CRConfig.json	The interval in seconds on which Traff
zonemanager.dynamic.response.expiration	CRConfig.json	A duration (e.g.: “300s”) that defines l
zonemanager.dynamic.concurrencylevel	CRConfig.json	An integer that defines the size of the c
zonemanager.dynamic.initialcapacity	CRConfig.json	An integer that defines the initial size c
zonemanager.init.timeout	CRConfig.json	An integer that defines the number of r
DNSKEY.generation.multiplier	CRConfig.json	Used to determine when new DNSSEC
DNSKEY.effective.multiplier	CRConfig.json	Used when creating an effective date f

Deprecated since version ATCv4.0: The use of “CRConfig.xml” as a *Parameter* “*Config File*” *value* has no known meaning, and its use for configuring Traffic Router is deprecated. All configuration (?) that previously used that value should instead use the equivalent *Parameter* with the *Config File* value “CRConfig.json”.

## Consistent Hashing

Traffic Router does special optimization for some requests to ensure that requests for specific content are consistently fetched from a small number (often exactly one, but dependent on *Initial Dispersion*) of *cache servers* - thus ensuring it stays “fresh” in the cache. This is done by performing “consistent hashing” on request paths (when HTTP routing) or names requested for resolution (when DNS routing). To an extent, this behavior is configurable by modifying fields on *Delivery Services*. Consistent hashing acts differently on a *Delivery Service* based on how *Delivery Services* of its *Type* route content.

- **HTTP, HTTP\_NO\_CACHE, HTTP\_LIVE, HTTP\_LIVE\_NATNL, DNS, DNS\_LIVE, and DNS\_N**  
These *Delivery Service Types* route directly to *cache servers*, so consistent hashing is used to choose a *cache server* to which the client will be redirected.
- **STEERING and CLIENT\_STEERING** These *Delivery Service Types* route to “target” *Delivery Services*, so consistent hashing is used to choose a “target” which will service the client request.

See also:

See the [Wikipedia article on consistent hashing](#).

## Consistent Hashing Patterns

New in version 4.0.

Regular expressions (“patterns”) can be provided in the *Consistent Hashing Regular Expression* field of an HTTP-*routed* Delivery Service to influence what parts of an HTTP request path are considered when performing consistent hashing. These patterns propagate to Traffic Router through *Snapshots*.

---

**Important:** Consistent Hashing Patterns on STEERING-*Type Delivery Services* will be used for Consistent Hashing - the Consistent Hashing Pattern(s) of said *Delivery Service*’s target(s) will **not** be considered. If Consistent Hashing Patterns are important to the routing of content on a STEERING-*Type* or CLIENT\_STEERING-*Type Delivery Service*, they **must** be defined on that *Delivery Service* itself, and **not** on its target(s).

---

## How it Works

The supplied *Consistent Hashing Regular Expression* is applied to the request path to extract matching elements to build a new string *before* consistent hashing is done. For example, using the pattern `/. *?(/.*?/). *?(m3u8)` and given the request paths `/test/path/asset.m3u8` and `/other/path/asset.m3u8` the resulting string used for consistent hashing will be `/path/m3u8`

**See also:**

See Oracle’s [documentation for the java.util.regex.Pattern](#) implementation in Java.

## Testing Pattern-Based Consistent Hashing

In order to test this feature without affecting the delivery of traffic through a CDN, there are several test tools in place.

- **Traffic Router API** Several Traffic Router endpoints exist to test regular expression application against a request path, *cache server* selection, and *Delivery Service* selection.
- **Traffic Ops API** The *consistenthash* endpoint will proxy request data through to one of the Traffic Router endpoints in order to test regular expression application against a request path, in the event that direct access to the *Traffic Router API* is not possible and/or desired.
- **Traffic Portal** On the *Delivery Service* creation/modification form in Traffic Portal (under *Delivery Services*), there is a *Test Regex* section that the user can use to validate a regular expression before saving it to a *Delivery Service*.

### Consistent Hash Query Parameters

Normally, when performing consistent hashing for an HTTP-routed *Delivery Service*, any query parameters present in the request are ignored. That is, if a client requests `/some/path?key=value` consistent hashing is only performed on the string `/some/path`. However, query parameters that are part of uniquely identifying content can be specified by adding them to the set of *Consistent Hashing Query Parameters* of a *Delivery Service*. For example, suppose that the file `/video.mp4` is available on the *origin server* in different resolutions, which are specified by the `resolution` query parameter. This means that `/video.mp4?resolution=480p` and `/video.mp4?resolution=720p` share a *request path*, but represent different *content*. In that case, adding `resolution` to the *Delivery Service's Consistent Hashing Query Parameters* will cause consistent hashing to be done on e.g. `/video.mp4?resolution=480p` instead of just `/video.mp4` - however if the client requests e.g. `/video.mp4?resolution=480p&bitrate=120kbps` consistent hashing will *only* consider `/video.mp4?resolution=480p`.

---

**Note:** *Consistent Hashing Patterns* are applied *before* query parameters are considered - i.e. a pattern cannot match against query parameters, and need not worry about query parameters contaminating matches.

---

---

**Important:** Consistent Hash Query Parameters on the *targets* of *STEERING-Type Delivery Services* will be used for Consistent Hashing - the Consistent Hash Query Parameters of said *Delivery Services* themselves will **not** be considered. If Consistent Hash Query Parameters are important to the routing of content on a *STEERING-Type* or *CLIENT\_STEERING-Type Delivery Service*, they **must** be defined *on that Delivery Service's target(s)*, and **not** on the *Delivery Service itself*.

---

**Caution:** Certain query parameters are reserved by Traffic Router for its own use, and thus cannot be present in any Consistent Hash Query Parameters. These reserved parameters are:

- `trred`
- `format`
- `fakeClientIPAddress`



## DNSSEC

### See also:

[The Wikipedia page on Domain Name Security Extensions](#)

## Overview

DNSSEC is a set of extensions to DNS that provides a cryptographic mechanism for resolvers to verify the authenticity of responses served by an authoritative DNS server. Several RFCs ([RFC 4033](#), [RFC 4044](#), [RFC 4045](#)) describe the low level details and define the extensions, [RFC 7129](#) provides clarification around authenticated denial of existence of records, and finally [RFC 6781](#) describes operational best practices for administering an authoritative DNSSEC-enabled DNS server. The authenticated denial of existence [RFC 7129](#) describes how an authoritative DNS server responds in NXDOMAIN and NODATA scenarios when DNSSEC is enabled. Traffic Router currently supports DNSSEC with NSEC (Next Secure Record), however, NSEC3 (Next Secure Record version 3) and more configurable options are planned for the future.

## Operation

Upon startup or a configuration change, Traffic Router obtains keys from the ‘keystore’ API in Traffic Ops which returns KSK (Key Signing Key)s and ZSK (Zone Signing Key)s for each *Delivery Service* that is a sub-domain of the CDN’s TLD in addition to the keys for the CDN TLD itself. Each key has timing information that allows Traffic Router to determine key validity (expiration, inception, and effective dates) in addition to the appropriate TTL to use for the DNSKEY record(s). All TTLs are configurable *Parameters* in *The Traffic Router Profile*.

Once Traffic Router obtains the key data from the API, it converts each public key into the appropriate record types (DNSKEY, DS) to place in zones and uses the private key to sign zones. DNSKEY records are added to each *Delivery Service*’s zone (e.g.: `demo1.mycdn.ciab.test`) for every valid key that exists, in addition to the CDN TLD’s zone. A DS record is generated from each zone’s KSK and is placed in the CDN TLD’s zone (e.g.: `mycdn.ciab.test`); the DS record for the CDN TLD must be placed in its parent zone, which is not managed by Traffic Control.

The DNSKEY to DS record relationship allows resolvers to validate signatures across zone delegation points. With Traffic Control, we control all delegation points below the CDN’s TLD, **however, the DS record for the CDN TLD must be placed in the parent zone** (e.g.: `ciab.test`), **which is not managed by Traffic Control**. As such, the DS record must be placed in the parent zone prior to enabling DNSSEC, and prior to generating a new CDN KSK. Based on your deployment’s DNS configuration, this might be a manual process or it might be automated. Either way, extreme care and diligence must be taken and knowledge of the management of the upstream zone is imperative for a successful DNSSEC deployment.

To enable DNSSEC for a CDN in Traffic Portal, Go to *CDNs* from the sidebar and click on the desired CDN, then toggle the ‘DNSSEC Enabled’ field to ‘true’, and click on the green *Update* button to save the changes.

### Rolling Zone Signing Keys

Traffic Router currently follows the ZSK pre-publishing operational best practice described in [RFC 6781#section-4.1.1.1](#). Once DNSSEC is enabled for a CDN in Traffic Portal, key rolls are triggered by Traffic Ops via the automated key generation process, and Traffic Router selects the active ZSKs based on the expiration information returned from the ‘keystore’ API of Traffic Ops.

### Edge Traffic Routing

#### Overview

Edge Traffic Routing is a feature that enables localization for more DNS record types than just the routing name for DNS delivery services. This feature has two main components: localization of HTTP delivery service routing names and localization for CDN-managed NS records. This allows Traffic Router to scale horizontally more easily, as there is a practical limit to how many records can be in an RRset for NS, A, or AAAA record types. The practical limit is typically an answer size exceeding 512 bytes; we have observed issues where some clients and/or resolvers do not honor larger answer responses, while in some cases, resolvers are unable to use TCP for larger responses. Additionally, this feature allows Traffic Router to serve an RRset containing only Traffic Routers that are near the client (resolver), placing more control over which Traffic Routers a given resolver or client will interact with.

Localizing NS records reduces latency for the resolver, which, due to caching in DNS is of little utility for NS records, but it will force resolvers to use the closest Traffic Routers for all queries. This is important for look-ups of CDN routing name records that have very short TTLs, meaning DNS traffic for routing name records is frequent. Like localization for routing names of DNS delivery services, localizing the routing name for HTTP delivery services provides the client (end user) with a list of Traffic Routers that are physically close, reducing latency when the client moves from DNS resolution to the HTTP connection to Traffic Router for the 302 redirect to the edge cache. This feature reduces latency between a resolver and Traffic Router, reduces latency for the client’s HTTP request to Traffic Router, and allows Traffic Control to dictate which Traffic Routers are used in any given location when a client can be localized.

#### Edge DNS Routing

Edge DNS routing refers to the localization of NS records in Traffic Router. This can be turned on and off via the `edge.dns.routing` parameter; the number of records returned is controlled via `edge.dns.limit` and there is a hard-coded default limit of 4 when this feature is enabled. See [The Traffic Router Profile](#) documentation for parameter details.



## Edge HTTP Routing

Edge HTTP routing refers to the localization of A and AAAA records that represent routing names for HTTP delivery services. This can be turned on and off via the `edge.http.routing` parameter; the number of records returned is controlled via `edge.http.limit` and there is a hard-coded default limit of 4 when this feature is enabled. The default or global limit can be overridden by modifying the `maxDnsAnswers` setting on the delivery service. See *The Traffic Router Profile* documentation for parameter details.

## Edge Traffic Router Selection

Traffic Router performs localization on client requests in order to determine which Traffic Routers should service a given request. After localization, Traffic Router will perform a consistent hash on the incoming name and will use the value to refine Traffic Router selection. There are two main cases for Traffic Router selection: a localization hit, and a localization miss.

### Localization Hit: Consistent Hash (CH)

When a client is localized, Traffic Router selects the nearest Traffic Router Location (*Cache Group*) based on proximity. Proximity is determined by using the latitude and longitude of the client, regardless of whether it is a coverage zone or geolocation hit, and the latitude and longitude specified on Traffic Router Locations. Once the location is identified, a consistent hash is performed on the incoming name and the list of Traffic Routers is ordered based upon the consistent hash. Once ordered, the list is limited to the appropriate number based on the limit parameter specified by the hard-coded default (4), `edge.dns.limit`, `edge.http.limit`, or `maxDnsAnswers`, depending on the configuration and request being localized. This approach can be thought of as the CH (consistent hash) selection process.

### Localization Miss: Consistent Hash + Round Robin (CH + RR)

When a client cannot be localized, Traffic Router still needs to produce a list of Traffic Routers to service the request. Because the number of Traffic Routers in the CDN could far exceed the practical limits of what constitutes a “normal” sized answer, a selection algorithm is applied to select Traffic Routers. Much like the localization hit scenario, the incoming request name is consistent hashed and results size is limited by the same parameters. Because no client location is known, Traffic Router must distribute the load across all Traffic Router Locations. To distribute the load, Traffic Router will order all Traffic Routers at each location based on the consistent hash, selecting a Traffic Router at the *n*th position, incrementing the index, *n*, after iterating over all locations.

For example, with four Traffic Router Locations each containing 10 Traffic Routers and a limit of 6, the algorithm would:

- Consistent hash the incoming name
- Order the Traffic Routers at each location by the consistent hash

- Select the Traffic Router at the first position of each location
- Select the Traffic Router at the second position of each location, stopping after selecting the 6th Traffic Router

Because the algorithm employs consistent hashing, the answers should be consistent as long as the topology remains the same. This approach can be thought of as the consistent hash round robin (CH + RR) selection process.

### Example Request Flow

The following is an example of the request flow when a client requests the routing name for an example delivery service, `tr.service.cdn.example.com`. The request flow assumes that the resolver is cold and has yet to build a local cache of look-ups, meaning it has to walk the domain hierarchy asking for NS records until it reaches `service.cdn.example.com`. This example starts after the resolver has determined which name servers are authoritative for `cdn.example.com`. Note that the same logic is applied for each of the three queries made by the resolver.

### Troubleshooting and log files

Traffic Router log files can be found under `/opt/traffic_router/var/log` and `/opt/tomcat/logs`. Initialization and shutdown logs are in `/opt/tomcat/logs/catalinadata.out`. Application related logging is in `/opt/traffic_router/var/log/traffic_router.log`, while access logs are written to `/opt/traffic_router/var/log/access.log`.

### Event Log File Format

#### Summary

All access events to Traffic Router are logged to the file `/opt/traffic_router/var/log/access.log`. This file grows up to 200MB and gets rolled into older log files, ten log files total are kept (total of up to 2GB of logged events per Traffic Router instance)

Traffic Router logs access events in a format that largely follows ATS [event logging format](#).

#### Message Format

- Except for the first item, each event that is logged is a series of space-separated key/value pairs.
- The first item is always the Unix epoch in seconds with a decimal field precision of up to milliseconds.
- Each key/value pair is in the form of `unquoted_string="optionally quoted string"`

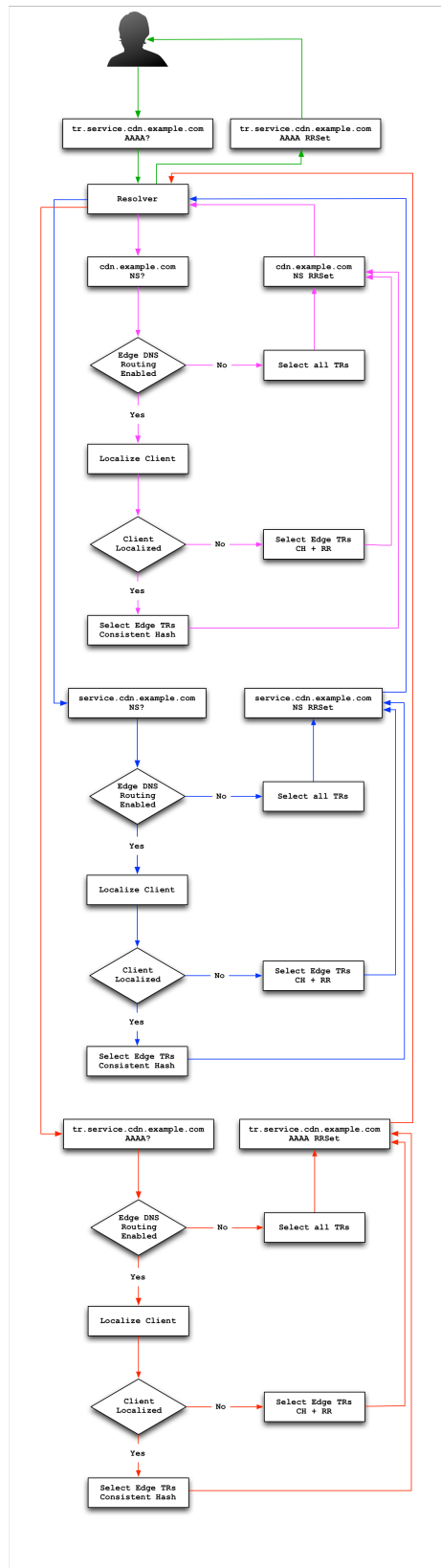


Fig. 21: Example Request Flow for Edge Traffic Routing. Note this picks up when the resolver hits the CDN managed domain.

- Values that are quoted strings may contain whitespace characters.
- Values that are not quoted should not contains any whitespace characters.

**Note:** Any value that is a single dash character or a dash character enclosed in quotes represents an empty value

Table 59: Fields Always Present

Name	Description	Data
qtype	Whether the request was for DNS or HTTP	Always “DNS” or “HTTP”
chi	The IP address of the requester	Depends on whether this was a DNS or HTTP request, see other sections
rhi	The IP address of the request source address	Depends on whether this was a DNS or HTTP request, see other sections
ttms	The amount of time in milliseconds it took Traffic Router to process the request	A number greater than or equal to zero
rtype	Routing result type	One of ERROR, CZ, DEEP_CZ, GEO, MISS, STATIC_ROUTE, DS_REDIRECT, DS_MISS, INIT, FED
rloc	GeoLocation of result	Latitude and longitude in degrees as floating point numbers
rdtl	Result details Associated with unusual conditions	One of DS_NOT_FOUND, DS_NO_BYPASS, DS_BYPASS, DS_CZ_ONLY, DS_CZ_BACKUP_CG
rerr	Message about an internal Traffic Router error	String

### See also:

If [Regional Geo-Blocking](#) is enabled on the *Delivery Service*, an additional field (rgb) will appear.

### Sample Message

Items within brackets are detailed under the HTTP and DNS sections

#### 55: Example Logfile Lines

```
144140678.000 qtype=DNS chi=192.168.10.11 rhi=- ttms=789 [Fields
↳Specific to the DNS request] rtype=CZ rloc="40.252611,58.439389"
↳rdtl=- rerr="-" [Fields Specific to the DNS result]
144140678.000 qtype=HTTP chi=192.168.10.11 rhi=- ttms=789 [Fields
↳Specific to the HTTP request] rtype=GEO rloc="40.252611,58.439389"
↳rdtl=- rerr="-" [Fields Specific to the HTTP result]
```

---

**Note:** These samples contain fields that are always present for every single access event to Traffic Router

---

## rtype Meanings

- The request was not redirected. This is usually a result of a DNS request to the Traffic Router or an explicit denial for that request

**ANON\_BLOCK** The client's IP matched an *Anonymous Blocking* rule and was blocked

**CZ** The result was derived from Coverage Zone data based on the address in the *chi* field

**DEEP\_CZ** The result was derived from Deep Coverage Zone data based on the address in the *chi* field

**DS\_MISS** *\*HTTP Only\** No HTTP *Delivery Service* supports either this request's URL path or headers

**DS\_REDIRECT** The result is using the Bypass Destination configured for the matched *Delivery Service* when that *Delivery Service* is unavailable or does not have the requested resource

**ERROR** An internal error occurred within Traffic Router, more details may be found in the *rerr* field

**FED** *\*DNS Only\** The result was obtained through federated coverage zone data outside of any *Delivery Services*

**GEO** The result was derived from geolocation service based on the address in the *chi* field

**GEO\_REDIRECT** The request was redirected based on the National Geo blocking (Geo Limit Redirect URL) configured on the *Delivery Service*

**GEO\_DS** The request was redirected to the Miss Location configured on the *Delivery Service*, because CZF couldn't resolve the client IP, and Maxmind returned the default coordinates of the country code of the client IP

**MISS** Traffic Router was unable to resolve a DNS request or find a cache for the requested resource

**RGALT** The request was redirected to the *Regional Geo-Blocking* URL. Regional Geo blocking is enabled on the *Delivery Service* and is configured through the *regional\_geoblock.polling.url* *Parameter* on the Traffic Router *Profile*

**RGDENY** *\*DNS Only\** The result was obtained through federated coverage zone data outside of any *Delivery Service* - the request was regionally blocked because there was no rule for the request made

**STATIC\_ROUTE** *\*DNS Only\** No DNS *Delivery Service* supports the hostname portion of the requested URL

### rdt1 Meanings

- The request was not redirected. This is usually a result of a DNS request to the Traffic Router or an explicit denial for that request

**DS\_BYPASS** Used a bypass destination for redirection of the *Delivery Service*

**DS\_CLIENT\_GEO\_UNSUPPORTED** Traffic Router did not find a resource supported by coverage zone data and was unable to determine the geographic location of the requesting client

**DS\_CZ\_BACKUP\_CG** Traffic Router found a backup cache via fall-back (through the `edgeLocation` field of a *Snapshot*) or via coordinates (*Coverage Zone File*) configuration

**DS\_CZ\_ONLY** The selected *Delivery Service* only supports resource lookup based on coverage zone data

**DS\_NO\_BYPASS** No valid bypass destination is configured for the matched *Delivery Service* and the *Delivery Service* does not have the requested resource

**DS\_NOT\_FOUND** Always goes with rtypes `STATIC_ROUTE` and `DS_MISS`

**GEO\_NO\_CACHE\_FOUND** Traffic Router could not find a resource via geographic location data based on the requesting client's location

**NO\_DETAILS** This entry is for a standard request

**REGIONAL\_GEO\_ALTERNATE\_WITHOUT\_CACHE** This goes with the rtype `RGDENY`. The URL is being regionally blocked

**REGIONAL\_GEO\_NO\_RULE** The request was blocked because there was no rule in the *Delivery Service* for the request

### HTTP Specifics

### 56: Sample Message

```
1452197640.936 qtype=HTTP chi=69.241.53.218 rhi=- url="http://foo.test.
→example.com/some/asset.m3u8" cqhm=GET cqhv=HTTP/1.1 rtype=GEO rloc=
→"40.252611,58.439389" rdtl=- rerr="-" pssc=302 ttms=0 rurl="http://
→cache1.test.example.com:8090/some/asset.m3u8" rh="Accept: /*/*" rh=
→"myheader: asdasdasdasfasg"
```

Table 60: Request Fields

Name	Description	Data
url	Requested URL with query string	A URL String
cqhm	Http Method	e.g GET, POST
cqhv	Http Protocol Version	e.g. HTTP/1.1
rh	One or more of these key value pairs may exist in a logged event and are controlled by the configuration of the matched <i>Delivery Service</i>   Key/value pair of the format name : value	

Table 61: Response Fields

Name	Description
rurl	The resulting URL of the resource requested by the client

## DNS Specifics

57: Sample Message

```
144140678.000 qtype=DNS chi=192.168.10.11 rhi=- ttms=123 xn=65535
→fqdn=www.example.com. type=A class=IN ttl=12345 rcode=NOERROR
→rtype=CZ rloc="40.252611,58.439389" rdtl=- rerr="-" ans="192.168.1.2
→192.168.3.4 0:0:0:0:0:ffff:c0a8:102 0:0:0:0:0:ffff:c0a8:304"
```

Table 62: Request Fields

Name	Description	Data
xn	The ID from the client DNS request header	a whole number between 0 and 65535 (inclusive)
rhi	The IP address of the resolver when EDNS0 client subnet extensions are enabled.	An IPv4 or IPv6 string, or dash if request is for resolver only and no client subnet is present
fqdn	The qname field from the client DNS request message (i.e. the FQDN the client is requesting be	A series of DNS labels/domains separated by ‘.’ characters and ending with a ‘.’ character
type	The qtype field from the client DNS request message (i.e. the type of resolution that’s requested such as IPv4, IPv6)	Examples are A (IPv4), AAAA (IPv6), NS (Name Service), SOA, and CNAME, (see <a href="#">qtype</a> )
class	The qclass field from the client DNS request message (i.e. the class of resource being requested)	Either IN (Internet resource) or ANY (Traffic Router rejects requests with any other value of class)

Table 63: Response Fields

Name	Description	Data
ttl	The ‘time to live’ in seconds for the answer provided by Traffic Router (clients can reliably use this answer for this long without re-querying traffic router)	A whole number between 0 and 4294967295 (inclusive)
rcode	The result code for the DNS answer provided by Traffic Router	One of NOERROR (success), NOTIMP (request is not NOTIMP (request is not supported), REFUSED (request is refused to be answered), or NXDOMAIN (the domain/name requested does not exist)



## Deep Caching

### Overview

Deep Caching is a feature that enables clients to be routed to the closest possible “deep” Edge-tier *cache servers* on a per-*Delivery Service* basis. The term “deep” is used in the networking sense, meaning that the Edge-tier *cache servers* are located deep in the network where the number of network hops to a client is as minimal. This deep caching topology is desirable because storing content closer to the client gives better bandwidth savings, and sometimes the cost of bandwidth usage in the network outweighs the cost of adding storage. While it may not be feasible to cache an entire copy of the CDN’s contents in every deep location (for the best possible bandwidth savings), storing just a relatively small amount of the CDN’s most requested content can lead to very high bandwidth savings.

### What You Need

1. Edge cache deployed in “deep” locations and registered in Traffic Ops
2. A *Deep Coverage Zone File* mapping these deep cache hostnames to specific network prefixes
3. Deep caching *Parameters* in the Traffic Router *Profile*
  - `deepcoveragezone.polling.interval`
  - `deepcoveragezone.polling.url`

**See also:**

See *The Traffic Router Profile* for details.

4. Deep Caching enabled on one or more HTTP *Delivery Services* (i.e. ‘Deep Caching’ field on the *Delivery Service* details page (under *Advanced Options*) set to **ALWAYS**)

### How it Works

Deep Coverage Zone routing is very similar to that of regular Coverage Zone routing, except that the DCZF (Deep Coverage Zone File) is preferred over the regular CZF (Coverage Zone File) for *Delivery Services* with Deep Caching enabled. If the client requests a Deep Caching-enabled *Delivery Service* and their IP address gets a “hit” in the DCZF, Traffic Router will attempt to route that client to one of the available “deep” *cache servers* in the client’s corresponding zone. If there are no “deep” *cache servers* available for a client’s request, Traffic Router will fall back to the regular CZF and continue regular CZF routing from there.

## Steering Feature

### Overview

A Steering *Delivery Service* is a *Delivery Service* that is used to route a client to another *Delivery Service*. The *Type* of a Steering *Delivery Service* is either STEERING or CLIENT\_STEERING. A Steering *Delivery Service* will have target *Delivery Services* configured for it with weights assigned to them. Traffic Router uses the weights to make a consistent hash ring which it then uses to make sure that requests are routed to a target based on the configured weights. This consistent hash ring is separate from the consistent hash ring used in cache selection.

Special regular expressions - referred to as ‘filters’ - can also be configured for target *Delivery Services* to pin traffic to a specific *Delivery Service*. For example, if the filter `.*news/.*` for a target called `target-ds-1` is created, any requests to Traffic Router with “news” in them will be routed to `target-ds-1`. This will happen regardless of the configured weights.

### Some other points of interest

- Steering is currently only available for HTTP *Delivery Services* that are a part of the same CDN.
- A new role called STEERING has been added to the Traffic Ops database. Only users with the Steering *Role* or higher can modify steering assignments for a *Delivery Service*.
- Traffic Router uses the steering endpoints of the *Traffic Ops API* to poll for steering assignments, the assignments are then used when routing traffic.

A couple simple use-cases for Steering are:

- Migrating traffic from one *Delivery Service* to another over time.
- Trying out new functionality for a subset of traffic with an experimental *Delivery Service*.
- Load balancing between *Delivery Services*

### The Difference Between STEERING and CLIENT\_STEERING

The only difference between the STEERING and CLIENT\_STEERING *Delivery Service Types* is that CLIENT\_STEERING explicitly allows a client to bypass Steering by choosing a destination *Delivery Service*. A client can accomplish this by providing the X-TC-Steering-Option HTTP header with a value of the `xml_id` of the target *Delivery Service* to which they desire to be routed. When Traffic Router receives this header it will route to the requested target *Delivery Service* regardless of weight configuration. This header is ignored by STEERING *Delivery Services*.

## Configuration

The following needs to be completed for Steering to work correctly:

1. Two target *Delivery Services* are created in Traffic Ops. They must both be HTTP *Delivery Services* part of the same CDN.
2. A *Delivery Service* with type STEERING or CLIENT\_STEERING is created in Traffic Portal.
3. Target *Delivery Services* are assigned to the Steering *Delivery Service* using Traffic Portal.
4. A user with the role of Steering is created.
5. The Steering user assigns weights to the target *Delivery Services*.
6. If desired, the Steering user can create filters for the target *Delivery Services*.

**See also:**

For more information see *Configure Delivery Service Steering*.

## HTTPS for HTTP Delivery Services

New in version 1.7: Traffic Router now has the ability to allow HTTPS traffic between itself and clients on a per-HTTP *Delivery Service* basis.

---

**Note:** As of version 3.0 Traffic Router has been integrated with native OpenSSL. This makes establishing HTTPS connections to Traffic Router much less expensive than previous versions. However establishing an HTTPS connection is more computationally demanding than an HTTP connection. Since each client will in turn get redirected to an ATS instance, Traffic Router is most always creating a new HTTPS connection for all HTTPS traffic. It is likely to mean that an existing Traffic Router may have some decrease in performance if you wish to support a lot of HTTPS traffic. As noted for DNSSEC, you may need to plan to scale Traffic Router vertically and/or horizontally to handle the new load.

---

The HTTPS set up process is:

1. Select one of '1 - HTTPS', '2 - HTTP AND HTTPS', or '3 - HTTP TO HTTPS' for the *Delivery Service*
2. Generate private keys for the *Delivery Service* using a wildcard domain such as \*.my-delivery-service.my-cdn.example.com
3. Obtain and import signed certificate chain
4. Perform a CDN *Snapshot*

Clients may make HTTPS requests to *Delivery Services* only after the CDN *Snapshot* propagates to Traffic Router and it receives the certificate chain from Traffic Ops.

### Protocol Options

**HTTP** Any secure client will get an SSL handshake error. Non-secure clients will experience the same behavior as prior to 1.7

**HTTPS** Traffic Router will only redirect (send a 302 Found response) to clients communicating with a secure connection, all other clients will receive a 503 Service Unavailable response

**HTTP AND HTTPS** Traffic Router will redirect both secure and non-secure clients

**HTTP TO HTTPS** Traffic Router will redirect non-secure clients with a 302 Found response and a location that is secure (i.e. an `https://` URL instead of an `http://` URL), while secure clients will be redirected immediately to an appropriate target or *cache server*.

### Certificate Retrieval

**Warning:** If you have HTTPS *Delivery Services* in your CDN, Traffic Router will not accept **any** connections until it is able to fetch certificates from Traffic Ops and load them into memory. Traffic Router does not persist certificates to the Java Keystore or anywhere else.

Traffic Router fetches certificates into memory:

- At startup time
- When it receives a new CDN *Snapshot*
- Once an hour starting whenever the most recent of the last of the above occurred

---

**Note:** To adjust the frequency at which Traffic Router fetches certificates add the *Parameter* `certificates.polling.interval` with the ConfigFile “CRConfig.json” and set it to the desired duration in milliseconds.

---

---

**Note:** Taking a CDN *Snapshot* may be used at times to avoid waiting the entire polling cycle for a new set of certificates.

---

**Warning:** If a CDN *Snapshot* is taken that involves a *Delivery Service* missing its certificates, Traffic Router will ignore **ALL** changes in that CDN *Snapshot* until one of the following occurs:

- It receives certificates for that *Delivery Service*
- Another CDN *Snapshot* is taken and the *Delivery Service* without certificates is changed such that its HTTP protocol is set to ‘http’

## Certificate Chain Ordering

The ordering of certificates within the certificate bundle matters. It must be:

1. Primary Certificate (e.g. the one created for `*.my-delivery-service.my-cdn.example.com`)
2. Intermediate Certificate(s)
3. Root Certificate from a CA (optional)

**Warning:** If something is wrong with the certificate chain (e.g. the order of the certificates is backwards or for the wrong domain) the client will get an SSL handshake. Inspection of `/opt/tomcat/logs/catalina.log` is likely to yield information to reveal this.

To see the ordering of certificates you may have to manually split up your certificate chain and use `openssl(1ssl)` on each individual certificate

## Automatic Certificate Management Environment

Automatic Certificate Management Environment (ACME) is a protocol for automatically generating, renewing, and revoking SSL certificates. Currently, ACME can be used through [Let's Encrypt](#) or through [External Account Binding](#).

### External Account Binding

External account binding allows the user to use an existing account with an ACME provider to obtain, renew, and revoke SSL certificates. To use this functionality, fill in the fields in `cdn.conf` for the ACME provider with which the account is set up. The first time this is used for a specific ACME provider (defined by the `acme_provider` and `user_email` fields) the information will be used to get a private key and account URL from the ACME provider and register the account. These will be stored for later use. External account binding information can only be used once, so after the first time, the private key and URL will be used.

---

**Important:** The `acme_provider` and `user_email` combination must be unique. The `acme_provider` field must correlate to the `AuthType` field for each certificate to be renewed using that provider.

---

External account binding can be set up through `cdn.conf` by updating the following fields:

Table 64: Fields to update for external account binding using ACME protocol under *acme\_accounts*

Name	Type	Re- quired	Description
acme_provider	string	Yes	The certificate provider. This field needs to correlate to the AuthType field for each certificate so the renewal functionality knows which provider to use.
user_email	string	Yes	The email used to set up the account with the provider.
acme_url	string	Yes	The URL for the ACME.
kid	string	No	The key ID provided by the ACME provider for external account binding.
hmac_encoded	string	No	The HMAC key provided by the ACME provider for external account binding. This should be in Base64 URL encoded.

**Note:** The *kid* and *hmac\_encoded* fields are required unless the account has already been registered and the information has been stored in the Traffic Ops Database.

## Let's Encrypt

Let's Encrypt is a free, automated CA using ACME protocol. Let's Encrypt performs a domain validation before issuing or renewing a certificate. There are several options for domain validation but for this application the DNS challenge is used in order to receive wildcard certificates. Let's Encrypt sends a token to be used as a TXT record at `_acme-challenge.domain.example.com` and after verifying that the token is accessible there, will return the newly generated and signed certificate and key. The basic workflow implemented is:

1. POST to Let's Encrypt and receive the DNS challenge token.
2. Traffic Ops stores the DNS challenge.
3. Traffic Router has a watcher which checks with Traffic Ops for any new challenges or deleted challenges.
4. When a new record appears, Traffic Router temporarily adds a static route for the specified *Delivery Service* with the token from Let's Encrypt at `_acme-challenge.domain.example.com`.
5. Let's Encrypt continuously attempts to resolve it as a TXT record to verify ownership of the domain.

**Note:** DNSSEC should be turned on for any CDN using Let's Encrypt to guard against a 'Man in the Middle' interference with this transaction.

1. Let's Encrypt returns the signed certificate and key to Traffic Ops.

2. Traffic Ops stores the certificate and key in Traffic Vault and removes the DNS challenge record.
3. The Traffic Router watcher removes the TXT record.

Let's Encrypt can be set up through *cdn.conf* by updating the following fields:

Table 65: Fields to update for Let's Encrypt under *lets\_encrypt*

Name	Type	Re- quired	Description
user_email	string	Yes	Email to create account with Let's Encrypt or to receive expiration updates. If this is not included then <i>rate limits</i> may apply for the number of certificates.
send_expiration_email	boolean	No	Option to send email summarizing certificate expiration status
convert_self_signed	boolean	No	Option to convert self signed to Let's Encrypt certificates as they expire. Only works for certificates labeled as Self Signed in the Certificate Source field.
re-new_days_before_expiration	int	No	Number of days before expiration date to renew certificates
environment	string	No	Let's Encrypt environment to use. Options are 'staging' or 'production'. Defaults to 'production'.

## Automatic Certificate Renewal

If desired, an automated certificate renewal script is located at `/traffic_ops/etc/cron.d/autorenew_certs`. This job is setup to be run, but the file must be updated with the username and password for Traffic Ops in order to be run. In *cdn.conf* the following fields can be defined in order to alter the number of days in advance to renew and send a summary email after renewal.

---

**Note:** In order for this to work, the AuthType field for the certificate must match the ACME provider in the *cdn.conf*.

---



---

**Important:** After the automatic renewal script has run, a queue and snapshot must be run manually in order for the certificates to be used.

---

Table 66: Fields to update to run the automatic renewal script under *acme\_renewal*:

Name	Type	Re- quired	Description
summary_email	boolean	No	The email address to use for summarizing certificate expiration and renewal status. If it is blank, no email will be sent.
re-new_days_before_expiration	int	No	Number of days before expiration date to renew certificates. Default is 30 days.

Table 67: Fields to update for sending emails under *smtp*

Name	Type	Description
enabled	boolean	Enable sending emails through Simple Mail Transfer Protocol (SMTP)
user	string	User for SMTP server access
pass-word	string	Password for SMTP server access
address	string	SMTP server address including port

## Suggested Way of Setting up an HTTPS Delivery Service With Let's Encrypt Automation

Assuming you have already created a *Delivery Service* which you plan to modify to use HTTPS, do the following in Traffic Portal:

1. Select one of '1 - HTTPS', '2 - HTTP AND HTTPS', or '3 - HTTP TO HTTPS' for the protocol field of a *Delivery Service* and click the *Update* button
2. Go to *More* → *Manage SSL Keys*
3. Click on *More* → *Generate SSL Keys*
4. Click on the *Use Let's Encrypt* slider, click on the green *Generate Keys* button, then confirm that you want to make these changes
5. Take a new CDN *Snapshot*

Once this is done you should be able to verify that you are being correctly redirected by Traffic Router using e.g. `curl (1)` commands to HTTPS destinations on your *Delivery Service*.

## Suggested Way of Setting up an HTTPS Delivery Service With Certificate Authority

Assuming you have already created a *Delivery Service* which you plan to modify to use HTTPS, do the following in Traffic Portal:

1. Select one of '1 - HTTPS', '2 - HTTP AND HTTPS', or '3 - HTTP TO HTTPS' for the protocol field of a *Delivery Service* and click the *Update* button
2. Go to *More* → *Manage SSL Keys*
3. Click on *More* → *Generate SSL Keys*
4. Fill out the form and click on the green *Generate Keys* button, then confirm that you want to make these changes
5. Copy the contents of the Certificate Signing Request field and save it locally
6. Go back and select 'HTTP' for the protocol field of the *Delivery Service* and click *Save* (to avoid preventing other CDN *Snapshot* updates from being blocked by Traffic Router)
7. Follow your standard procedure for obtaining your signed certificate chain from a CA



8. After receiving your certificate chain import it into Traffic Ops
9. Edit the *Delivery Service*
10. Restore your original choice for the protocol field and click *Save*
11. Click *More* → *Manage SSL Keys*
12. Paste your key information into the appropriate fields
13. Click the green *Update Keys* button
14. Take a new CDN *Snapshot*

Once this is done you should be able to verify that you are being correctly redirected by Traffic Router using e.g. `curl (1)` commands to HTTPS destinations on your *Delivery Service*.

## Router Load Testing

The Traffic Router load testing tool is located in the [Traffic Control repository under test/router](#). It can be used to simulate a mix of HTTP and HTTPS traffic for a CDN by choosing the number of HTTP *Delivery Services* and the number HTTPS *Delivery Service* the test will exercise.

There are 2 parts to the load test:

- A web server that makes the actual requests and takes commands to fetch data from the CDN, start the test, and return current results.
- A web page that's used to run the test and see the results.

## Running the Load Tests

1. First, clone the [Traffic Control repository](#).
2. You will need to make sure you have a CA file on your machine
3. The web server is a Go program, set your GOPATH environment variable appropriately (we suggest \$HOME/go or \$HOME/src)
4. Open a terminal emulator and navigate to the `test/router/server` directory inside of the cloned repository
5. Execute the server binary by running `go run server.go`
6. Using your web browser of choice, open the file `test/router/index.html`
7. Authenticate against a Traffic Ops host - this should be a nearly instantaneous operation - you can watch the output from `server.go` for feedback
8. Enter the Traffic Ops host in the second form and click the button to get a list of CDN's
9. Wait for the web page to show a list of CDN's under the above form, this may take several seconds
10. The List of CDN's will display the number of HTTP- and HTTPS-capable *Delivery Services* that may be exercised

11. Choose the CDN you want to exercise from the drop-down menu
12. Fill out the rest of the form, enter appropriate numbers for each HTTP and HTTPS *Delivery Services*
13. Click *Run Test*
14. As the test runs the web page will occasionally report results including running time, latency, and throughput

### Tuning Recommendations

The following is an example of the command line parameters set in `/opt/traffic_router/conf/startup.properties` that has been tested on a multi-core server running under HTTPS load test requests. This is following the general recommendation to use the G1 garbage collector for JVM (Java Virtual Machine) applications running on multi-core machines. In addition to using the G1 garbage collector the `InitiatingHeapOccupancyPercent` was lowered to run garbage collection more frequently which improved overall throughput for Traffic Router and reduced ‘Stop the World’ garbage collection. Note that any environment variable settings in this file will override those set in `/lib/systemd/system/traffic_router.service`.

#### 58: Example CATALINA\_OPTS Configuration

```
CATALINA_OPTS="\
-server -Xms2g -Xmx8g \
-Dlog4j.configurationFile=$CATALINA_BASE/conf/log4j2.xml \
-Djava.library.path=/usr/lib64 \
-XX:+UseG1GC \
-XX:+UnlockExperimentalVMOptions \
-XX:InitiatingHeapOccupancyPercent=30"
```

### 3.1.6 Traffic Stats Administration

Traffic Stats consists of three separate components: Traffic Stats, InfluxDB, and Grafana. See below for information on installing and configuring each component as well as configuring the integration between the three and Traffic Ops.

#### Installation

##### Installing Traffic Stats

- See the [downloads](#) page for Traffic Control to get the latest release.
- Follow the instructions in *Building Traffic Control* to generate an RPM.
- Copy the RPM to your server
- Install the generated Traffic Stats RPM with `yum(8)` or `rpm(8)`

## Installing InfluxDB

In order to store Traffic Stats data you will need to install [InfluxDB](#). While not required, it is recommended to use some sort of high availability option like [Influx enterprise](#), [InfluxDB Relay](#), or another [high availability option](#).

## Installing Grafana

Grafana is used to display Traffic Stats/InfluxDB data in Traffic Ops. Grafana is typically run on the same server as Traffic Stats but this is not a requirement. Grafana can be installed on any server that can access InfluxDB and can be accessed by Traffic Ops. Documentation on installing Grafana can be found on the [Grafana website](#).

## Configuration

### Configuring Traffic Stats

Traffic Stats's configuration file can be found in `/opt/traffic_stats/conf/traffic_stats.cfg`. It is a JSON-format file with the following properties.

- toUser** The username of the user as whom to connect to Traffic Ops
- toPasswd** The password to use when authenticating with Traffic Ops
- toUrl** The URL of the Traffic Ops server used by Traffic Stats
- influxUser** Optionally specify the user to use when connecting to InfluxDB (if InfluxDB is not configured to require authentication, has no effect)
- influxPassword** Optionally specify the password to use when connecting to InfluxDB (if InfluxDB is not configured to require authentication, has no effect)
- pollingInterval** The interval in seconds for which Traffic Monitor will wait between polling for stats and storing them in InfluxDB
- statusToMon** The name of the *Status* of the Traffic Monitors to poll (e.g. ONLINE or OFFLINE)
- seelogConfig** Optionally specify the absolute path to a [seelog](#) configuration file. Has no effect if the `logs` property is present.  
  
Deprecated since version ATCv6.1: This will be removed in the future, configurations should be migrated to the new `logs` property.
- logs** This property is an object containing keys that specify locations for different log streams.  
  
New in version ATCv6.1.
  - error** The location of error-level logs. If omitted, `null`, an empty string (`""`), or the special path `/dev/null` (even on Windows), error-level logs will not be emitted. The special values `"stderr"`

and “stdout” cause logging to use STDOUT or STDERR, respectively<sup>1</sup>.

**warning** The location of warning-level logs. If omitted, `null`, an empty string (“”), or the special path `/dev/null` (even on Windows), warning-level logs will not be emitted. The special values “stderr” and “stdout” cause logging to use STDOUT or STDERR, respectively<sup>?</sup>.

**info** The location of info-level logs. If omitted, `null`, an empty string (“”), or the special path `/dev/null` (even on Windows), info-level logs will not be emitted. The special values “stderr” and “stdout” cause logging to use STDOUT or STDERR, respectively<sup>?</sup>.

**debug** The location of debug-level logs. If omitted, `null`, an empty string (“”), or the special path `/dev/null` (even on Windows), debug-level logs will not be emitted. The special values “stderr” and “stdout” cause logging to use STDOUT or STDERR, respectively<sup>?</sup>.

**event** The location of event-level logs. If omitted, `null`, an empty string (“”), or the special path `/dev/null` (even on Windows), event-level logs will not be emitted. The special values “stderr” and “stdout” cause logging to use STDOUT or STDERR, respectively<sup>?</sup>.

---

**Note:** At the time of this writing, Traffic Stats does not make use of the “event” log level.

---

**dailySummaryPollingInterval** The interval, in seconds, on which Traffic Stats checks to see if daily stats need to be computed and stored.

**cacheRetentionPolicy** The default retention policy for cache stats

**dsRetentionPolicy** The default retention policy for *Delivery Service* statistics

**dailySummaryRetentionPolicy** The retention policy to be used for the daily statistics

**influxUrls** An array of InfluxDB hosts for Traffic Stats to write stats to.

---

<sup>1</sup> To log to files named literally “stdout” or “stderr”, use an absolute or relative file path e.g. “./stdout” or “/path/to/stderr”.

## Configuring InfluxDB

As mentioned above, it is recommended that InfluxDB be running in some sort of high availability configuration. There are several ways to achieve high availability so it is best to consult the high availability options on the [InfluxDB website](#).

Once InfluxDB is installed and configured, databases and retention policies need to be created. Traffic Stats writes to three different databases: `cache_stats`, `deliveryservice_stats`, and `daily_stats`. More information about the databases and what data is stored in each can be found in the [Traffic Stats Overview](#).

To easily create databases, retention policies, and continuous queries, run **create\_ts\_databases** from the `/opt/traffic_stats/influxdb_tools` directory on your Traffic Stats server. See the [InfluxDB Tools](#) section for more information.

## Configuring Grafana

In Traffic Portal the *Other* → *Grafana* menu item can be configured to display Grafana graphs using InfluxDB data (when not configured, this menu item will not appear). In order for this to work correctly, you will need two things:

1. A [Parameter](#) with the graph URL (more information below)
2. The graphs created in Grafana. See below for how to create some simple graphs in Grafana. These instructions assume that InfluxDB has been configured and that data has been written to it. If this is not true, you will not see any graphs.

To create a graph in Grafana, you can follow these basic steps:

1. Login to Grafana as an administrative user
2. Click on *Data Sources* → *Add New*
3. Enter the necessary information to configure your data source
4. Click on *Home* → *New* at the bottom
5. Click on “*Collapsed Menu Icon*” *Button* → *Add Panel* → *Graph*
6. Where it says *No Title (click here)* click and choose edit
7. Choose your data source at the bottom
8. You can have Grafana help you create a query, or you can create your own.

### 59: Sample Query

```
SELECT sum(value)*1000 FROM "monthly"."bandwidth.cdn.1min"
GROUP BY time(60s), cdn;
```

9. Once you have the graph the way you want it, click the *Save Dashboard* button at the top
10. You should now have a new saved graph

In order for Traffic Portal users to see Grafana graphs, Grafana will need to allow anonymous access. Information on how to configure anonymous access can be found on the configuration page of the [Grafana Website](#).

Traffic Portal uses custom dashboards to display information about individual *Delivery Services* or *Cache Groups*. In order for the custom graphs to display correctly, the Javascript files in `traffic_stats/grafana/` need to be in the `/usr/share/grafana/public/dashboards/` directory on the Grafana server. If your Grafana server is the same as your Traffic Stats server the RPM install process will take care of putting the files in place. If your Grafana server is different from your Traffic Stats server, you will need to manually copy the files to the correct directory.

### See also:

More information on custom scripted graphs can be found in the [scripted dashboards](#) section of the Grafana documentation.

## Configuring Traffic Portal for Traffic Stats

- The InfluxDB servers need to be added to Traffic Portal with a *Profile* that has the *Type* InfluxDB. Make sure to use port 8086 in the configuration.
- The traffic stats server should be added to Traffic Ops with a *Profile* that has the *Type* TRAFFIC\_STATS.
- *Parameters* for which stats will be collected are added with the release, but any changes can be made via *Parameters* that are assigned to the Traffic Stats *Profile*.

## Configuring Traffic Portal to use Grafana Dashboards

To configure Traffic Portal to use Grafana Dashboards, you need to enter the following *Parameters* and assign them to the special GLOBAL *Profile*. This assumes you followed instructions in the *Installation*, *Configuring Traffic Stats*, *Configuring InfluxDB*, and *Configuring Grafana* sections.

Table 68: Traffic Stats Parameters

parameter name	parameter value
all_graph_url	<code>https://grafanaHost/dashboard/db/deliveryservice-stats-dashboard</code>
cachegroup_graph_url	<code>https://grafanaHost/dashboard/script/traffic_ops_cachegroup.js?which=</code>
deliveryservice_graph_url	<code>https://grafanaHost/dashboard/script/traffic_ops_deliveryservice.js?which=</code>
server_graph_url	<code>https://grafanaHost/dashboard/script/traffic_ops_server.js?which=</code>
visual_status_panel_cdn	<code>https://grafanaHost/dashboard-solo/db/cdn-stats-dashboard?panelId=2&amp;fullscreen&amp;from=now-24h&amp;to=now-60s</code>
visual_status_panel_delivery	<code>https://grafanaHost/dashboard-solo/db/cdn-stats-dashboard?panelId=1&amp;fullscreen&amp;from=now-24h&amp;to=now-60s</code>
daily_bw_url	<code>https://grafanaHost/dashboard-solo/db/daily-summary-dashboard?panelId=1&amp;fullscreen&amp;from=now-3y&amp;to=now</code>
daily_served_url	<code>https://grafanaHost/dashboard-solo/db/daily-summary-dashboard?panelId=2&amp;fullscreen&amp;from=now-3y&amp;to=now</code>

where

**grafanaHost** is the FQDN of the Grafana server (again, usually the same as the Traffic Stats server),

**cdn-stats-dashboard** is the name of the Dashboard providing CDN-level statistics,

**deliveryservice-stats-dashboard** is the name of the Dashboard providing *Delivery Service*-level statistics, and

**daily-summary-dashboard** is the name of the Dashboard providing a daily summary of general statistics that would be of interest to administrators using Traffic Portal

## InfluxDB Tools

Under the Traffic Stats source directory there is a directory called `influxdb_tools`. These tools are meant to be used as one-off scripts to help a user quickly get new databases and continuous queries setup in InfluxDB. They are specific for Traffic Stats and are not meant to be generic to InfluxDB. Below is a brief description of each script along with how to use it.

### create/create\_ts\_databases.go

This program creates all [databases](#), [retention policies](#), and [continuous queries](#) required by Traffic Stats.

#### Pre-Requisites

- Go 1.7 or later
- Configured `$GOPATH` environment variable

#### Options and Arguments

##### **--help**

(Optional) Print usage information and exit (with a failure exit code for some reason)

##### **--password** password

The password that will be used by the user defined by **--user** to authenticate.

##### **--replication** N

(Optional) The number of nodes in the cluster (default: 3)

##### **--url** URL

The InfluxDB server's root URL - including port number, if required (default: `http://localhost:8086`)

##### **--user** username

The name of the user to use when connecting to InfluxDB

### sync/sync\_ts\_databases.go

This program is used to sync one InfluxDB environment to another. Only data from continuous queries is synced as it is down-sampled data and much smaller in size than syncing raw data. Possible use cases are syncing from production to development or syncing a new cluster once brought online.

#### Pre-Requisites

- Go 1.7 or later
- Configured `$GOPATH` environment variable



## Options and Arguments

- database** database\_name  
(Optional) Specify the name of a specific database to sync (default: all databases)
- days** N  
The number of days in the past to sync. 0 means ‘all’
- help**  
(Optional) Print usage information and exit
- source-password** password  
The password of the user named by *--source-user*
- source-url** URL  
(Optional) The URL of the InfluxDB instance *\_from\_* which data will be copied (default: `http://localhost:8086`)
- source-user** username  
The name of the user as whom the utility will connect to the source InfluxDB instance
- target-password** password  
The password of the user named by *--target-user*
- target-url** URL  
(Optional) The URL of the InfluxDB instance *\_to\_* which data will be copied (default: `http://localhost:8086`)
- target-user** username  
The name of the user as whom the utility will connect to the target InfluxDB instance

## 3.1.7 Traffic Server Administration

### Installing Traffic Server

1. Build the Traffic Server RPM. The best way to do this is to follow the [Apache Traffic Server documentation](#).
2. Build the astats RPM using the appropriate version number - ours are built [here](#)

---

**Note:** The astats plugin is bundled as a part of Apache Traffic Server as of version 7.2.

---

3. Install Traffic Server and astats

60: Apache Traffic Server Installation Using *yum(8)*

```
yum -y install trafficserver-*.rpm astats_over_http*.rpm
```

4. Add the server using the Traffic Portal UI:

1. Go to *Configure* → *Servers*
2. Click on the + button at the top of the page.
3. Complete the form. Be sure to fill out all fields marked ‘Required’
  - Set ‘Interface Name’ to the name of the network interface device from which Apache Traffic Server delivers content.
  - Set ‘Type’ to ‘MID’ or ‘EDGE’.
  - If you wish for the server to immediately be polled by the *Health Protocol*, set ‘Status’ to ‘REPORTED’.
4. Click on the *Create* button to submit the form.
5. Verify that the server status is now listed as **Reported**
5. Install the *t3c* script and run it in ‘badass’ mode to create the initial configuration

**See also:**

*Configuring Traffic Server*

6. Start ATS

61: Starting ATS with *systemd*(1)

```
systemctl start trafficserver
```

7. (Optional) Configure ATS to start automatically when the system powers on

62: Configuring ATS to Start Automatically Using *systemd*(1)

```
systemctl enable trafficserver
```

8. Verify that the installation is working
  1. Make sure that the service is running

### 63: Checking that ATS is Running Using *systemd*(1)

```
systemctl status trafficserver
```

2. Assuming a Traffic Monitor is already installed somewhere, check the “Cache States” table in its Web UI to verify that the ATS server appears.

## Configuring Traffic Server

All of the ATS application configuration files are generated by Traffic Ops and installed by *t3c*. The *t3c* file should be installed on all *cache servers* (See *Installing the Cache Config Tool*), usually in `/opt/ort`. It is used to do the initial install of the configuration files when the *cache server* is being deployed, and to keep the configuration files up-to-date when the *cache server* is already in service.

## Cache Config File Generation

In the past, ATS config files were generated by Traffic Ops. Cache config generation is now performed on cache servers themselves, via a library for generic use, and an application which uses the library and resides on the cache server. That application, *t3c*, is in the process of being rewritten as many small POSIX-style applications which will behave similar to the *git* ecosystem.

The library, `lib/go-atscfg`, allows users to write their own applications and servers, if they wish to generate ATS configuration files and deploy them to caches via other means. For example, if you wish to generate config files with an additional service, or continue generating config files on Traffic Ops itself via a plugin or local service.

The app `atstccfg` is installed by the cache config RPM alongside *t3c*. This app is a helper to *t3c* and does not have a stable interface, and should never need to be called directly by Traffic Control administrators.

The cache-side config generation provides several benefits. Primarily, it reduces the overhead and risk of the monolithic Traffic Ops installation and upgrade process, and allows operators to canary-test config changes one cache at a time, and in the event of an error, only rolling back a few canary caches rather than the entire Traffic Ops instance.

## Installing the Cache Config Tool

1. Build the RPM by following the instructions in *Building Traffic Control* and install it with *rpm*(8) or *yum*(8).
2. Install modules required by the RPM if needed
3. For initial configuration or when major changes (like a *Profile* change) need to be made, run the script in “badass” mode. All required RPM packages will be installed, all ATS

configuration files will be fetched and installed, and (if needed) the ATS service will be restarted.

---

**Note:** The first run gives a lot of state errors that are expected. The “badass” mode fixes these issues. If you run it a second time, this should be cleaner. Also, note that many “ERROR” messages emitted by *t3c* are actually information messages. Do not panic. We are in the process of converting legacy “ERROR” messages to informational logs.

---

4. Create a *cron(8)* entry for running *t3c* in “syncds” mode every 15 minutes. This makes Traffic Control check periodically if the server has updates pending, and if so get the updated configuration.

---

**Note:** By default, running *t3c* on an Edge-tier *cache server* will cause it to first wait for its parents (usually Mid-tier *cache server* s) to download their configuration before downloading its own configuration. Because of this, scheduling *t3c* for running every 15 minutes (with 5 minutes default dispersion) means that it might take up to ~35 minutes for queued updates to affect all *cache server* s. To customize this dispersion time, use the command line option `--dispersion=x` where *x* is the number of seconds for the dispersion period. Servers will select a random number from within this dispersion period to being downloading configuration files from Traffic Ops. Another option, `--login_dispersion=x` can be used to create a dispersion period after the job begins during which *t3c* will wait before logging in and checking Traffic Ops for updates to the server. This defaults to 0. If `use_reval_pending`, a.k.a. “Rapid Revalidate” is enabled, Edge-tier *cache servers* will **not** wait for their parents to download their configuration before downloading their own.

---

---

**Note:** In “syncds” mode, the *t3c* app updates only configurations that might be changed as part of normal operations, such as:

- *Delivery Services*
  - SSL certificates
  - Traffic Monitor IP addresses
  - Logging configuration
  - Revalidation requests (By default - if “Rapid Revalidate” is enabled, this will only be checked by using a separate revalidate command in *t3c*)
- 

5. If “Rapid Revalidate” is enabled in Traffic Ops, create a second *cron(8)* job for revalidation checks by running *t3c* in “revalidate” mode. *t3c* will not check revalidation files if “Rapid Revalidate” is enabled. This setting allows for a separate check to be performed every 60 seconds to verify if a revalidation update has been made.

## Strings with Special Meaning to t3c

When processing configuration files, if `traffic_ops_ort` encounters any of the strings in the *Replacement Strings* table it will perform the indicated replacement. This means that these strings can be used to create templates in *Profile Parameters* and certain *Delivery Service* configuration fields.

Table 69: Replacement Strings

String	Replaced With
<code>__CACHE_IPv4</code>	IPv4 address of the <i>cache server</i> on which t3c is running.
<code>__FULL_HOSTNAME</code>	hostname (i.e. including the full domain to which it belongs) of the <i>cache server</i> on which t3c is running.
<code>__HOSTNAME</code>	The (short) hostname of the <i>cache server</i> on which t3c is running.
<code>__RETURN</code>	A newline character ( <code>\n</code> ).
<code>__SERVER_TCP_PORT</code>	The <i>server</i> on which t3c is being run has a TCP port configured to something besides 80, this will be replaced with that TCP port value. <i>If it is set to 80, this string will simply be removed, NOT replaced with ANYTHING.</i>
<code>##OVERRIDE##</code>	This string is only valid in the content of files named “remap.config”. It is further described in <i>Remap Override</i>

Deprecated since version ATCv4.0: The use of `__RETURN` in lieu of a true newline character is (finally) no longer necessary, and the ability to do so will be removed in the future.

**Note:** There is currently no way to indicate that a server’s IPv6 address should be inserted - only IPv4 is supported.

## Remap Override

**Warning:** The ANY\_MAP `##OVERRIDE##` special string is a temporary solution and will be deprecated once Delivery Service Versioning is implemented. For this reason it is suggested that it not be used unless absolutely necessary.

The `##OVERRIDE##` template string allows the *Delivery Service Raw Remap Text* field to override to fully override the *Delivery Service*’s line in the `remap.config` ATS configuration file, generated by Traffic Ops. The end result is the original, generated line commented out, prepended with `##OVERRIDDEN##` and the `##OVERRIDE##` rule is activated in its place. This behavior is used to incrementally deploy plugins used in this configuration file. Normally, this entails cloning the *Delivery Service* that will have the plugin, ensuring it is assigned to a subset of the *cache servers* that serve the *Delivery Service* content, then using this `##OVERRIDE##` rule to create a `remap.config` rule that will use the plugin, overriding the normal rule. Simply grow the subset over time at the desired rate to slowly deploy the plugin. When it encompasses all *cache servers* that serve the original *Delivery Service*’s content, the “override *Delivery Service*” can be deleted and the original can use a non-`##OVERRIDE##` *Raw Remap Text* to add the plugin.

#### 64: Example of Remap Override

```
# This is the original line as generated by Traffic Ops
map http://from.example.com/ http://to.example.com/

# This is the raw remap text as configured on the delivery service
##OVERRIDE## map http://from.example.com/ http://to.example.com/ some_
→plugin.so

# The resulting content is what actually winds up in the remap.config_
→file:
##OVERRIDE##
map http://from.example.com/ http://to.example.com/ some_plugin.so
##OVERRIDDEN## map http://from.example.com/ http://to.example.com/
```

**Warning:** The “from” URL must exactly match for this to properly work (e.g. including trailing URL ‘/’), otherwise ATS may fail to initialize or reload while processing remap.config.

---

**Tip:** To assist in troubleshooting, it is strongly recommended that any ##OVERRIDE## rules in use should be documented on the original *Delivery Service*.

---

### 3.1.8 t3c

*t3c* is the tool used to deploy configuration to *cache servers* used by ATC. It stands for “Traffic Control Cache Configuration”.

### 3.1.9 Traffic Vault Administration

Currently, the supported backends for Traffic Vault are PostgreSQL and Riak, but Riak support is deprecated and may be removed in a future release. More backends may be supported in the future.

## PostgreSQL

In order to use the PostgreSQL backend for Traffic Vault, you will need to set the `traffic_vault_backend` option to "postgres" and include the necessary configuration in the `traffic_vault_config` section in `cdn.conf`. The `traffic_vault_config` options for the PostgreSQL backend are as follows:

**dbname** The name of the database to use

**hostname** The hostname of the database server to connect to

**password** The password to use when connecting to the database

**port** The port number that the database listens for new connections on (NOTE: the PostgreSQL default is 5432)

**user** The username to use when connecting to the database

**aes\_key\_location** The location on-disk for a base64-encoded AES key used to encrypt secrets before they are stored. It is highly recommended to backup this key to a safe, secure storage location, because if it is lost, you will lose access to all your Traffic Vault data. Either this option or `hashicorp_vault` must be used.

**hashicorp\_vault** This group of configuration options is for fetching the base64-encoded AES key from [HashiCorp Vault](#). This uses the [AppRole authentication method](#).

**address** The address of the HashiCorp Vault server, e.g. `http://localhost:8200`

**role\_id** The RoleID of the AppRole.

**secret\_id** The SecretID issued against the AppRole.

**secret\_path** The URI path where the secret AES key is located, e.g. `/v1/secret/data/trafficvault`. The secret should be stored using the [KV Secrets Engine](#) with a key of `traffic_vault_key` and value of a base64-encoded AES key, e.g. `traffic_vault_key='WoFc86CisM1aXo8D5GvDnq2h9kjULuIP4upaqX15SRc='`.

**login\_path** Optional. The URI path used to login with the AppRole method. Default: `/v1/auth/approle/login`

**timeout\_sec** Optional. The timeout (in seconds) for requests. Default: 30

**insecure** Optional. Disable server certificate verification. This should only be used for testing purposes. Default: false

**conn\_max\_lifetime\_seconds** Optional. The maximum amount of time (in seconds) a connection may be reused. If negative, connections are not closed due to a connection's age. If 0 or unset, the default of 60 is used.

**max\_connections** Optional. The maximum number of open connections to the database. Default: 0 (unlimited)

**max\_idle\_connections** Optional. The maximum number of connections in the idle connection pool. If negative, no idle connections are retained. If 0 or unset, the default of 30 is used.

**query\_timeout\_seconds** Optional. The duration (in seconds) after which database queries will time out and be cancelled. Default: 30

**ssl** Optional. Whether or not to use SSL to connect to the database. Default: false

### Example `cdn.conf` snippet:

```
{
  "traffic_ops_golang": {
    "traffic_vault_backend": "postgres",
    "traffic_vault_config": {
      "dbname": "tv_development",
      "hostname": "localhost",
      "user": "traffic_vault",
      "password": "twelve",
      "port": 5432,
      "ssl": false,
      "conn_max_lifetime_seconds": 60,
      "max_connections": 500,
      "max_idle_connections": 30,
      "query_timeout_seconds": 10,
      "aes_key_location": "/opt/traffic_ops/app/conf/tv.key"
    }
  }
}
```

## Administration of the PostgreSQL database for Traffic Vault

Similar to administering the Traffic Ops database, the *admin* tool should be used for administering the PostgreSQL Traffic Vault backend.

### `app/db/reencrypt/reencrypt`

The **reencrypt** binary is used to re-encrypt all data in the Postgres Traffic Vault with a new base64-encoded AES key.

---

**Note:** For proper resolution of configuration files, it's recommended that this binary be run from the `app/db/reencrypt` directory.

---



## Usage

```
./reencrypt [options]
```

## Options and Arguments

**--new-key** NEW\_KEY

(Optional) The file path for the new base64-encoded AES key. Default is `/opt/traffic_ops/app/conf/new.key`.

**--previous-key** PREVIOUS\_KEY

(Optional) The file path for the previous base64-encoded AES key. Default is `/opt/traffic_ops/app/conf/aes.key`.

**--cfg** CONFIG\_FILE

(Optional) The path for the configuration file. Default is `./reencrypt.conf`.

**--help**

(Optional) Print usage information and exit.

### 65: Example Usage

```
./reencrypt --new-key ~/exampleNewKey.txt --previous-key ~/
↪exampleOldKey.txt
```

## reencrypt.conf

This file deals with configuration of the Traffic Vault Database to be used with the **reencrypt** tool.

**dbname** The name of the PostgreSQL database used.

**hostname** The hostname (FQDN) of the server that runs the Traffic Vault Database.

**password** The password to use when authenticating with the Traffic Vault database.

**port** The port number on which the Traffic Vault Database is listening for incoming connections (NOTE: the PostgreSQL default is 5432).

**ssl** A boolean that sets whether or not the Traffic Vault Database encrypts its connections with SSL.

**user** The name of the user as whom to connect to the database.

### Riak (deprecated)

Deprecated since version ATCv6: The Riak Traffic Vault backend is deprecated and support may be removed in a future release. It is highly recommended to use the PostgreSQL Traffic Vault backend instead.

In order to use the Riak backend for Traffic Vault, you will need to set the `traffic_vault_backend` option to `"riak"` and include the necessary configuration in the `traffic_vault_config` section in `cdn.conf`. The `traffic_vault_config` options for the Riak backend are as follows:

**password** The password to use when authenticating with Riak

**user** The username to use when authenticating with Riak

**port** The Riak protobuf port to connect to. Default: 8087

**tlsConfig** Optional. Certain TLS options from [the `tls.Config` struct options](#) may be included here, such as `insecureSkipVerify: true` to disable certificate validation in order to use self-signed certificates for test/development purposes.

**MaxTLSVersion** Optional. This is the highest TLS version that Traffic Ops is allowed to use to connect to Traffic Vault. Valid values are `"1.0"`, `"1.1"`, `"1.2"`, and `"1.3"`. The default is `"1.1"`.

---

**Note:** Enabling TLS 1.1 in Riak itself is required for Traffic Ops to communicate with Riak. See [Enabling TLS 1.1](#) for details.

---

### Example `cdn.conf` snippet:

```
{
  "traffic_ops_golang": {
    "traffic_vault_backend": "riak",
    "traffic_vault_config": {
      "user": "riakuser",
      "password": "password",
      "MaxTLSVersion": "1.1",
      "port": 8087
    }
  }
}
```

## Installing the Riak backend for Traffic Vault

In order to successfully store private keys you will need to install Riak. The latest version of Riak can be downloaded on [the Riak website](#). The installation instructions for Riak can be found [here](#). Based on experience, version 2.0.5 of Riak is recommended, but the latest version should suffice.

## Configuring Riak

Follow these steps to configure Riak in a production environment.

### Self Signed Certificate configuration

---

**Note:** Self-signed certificates are not recommended for production use. Intended for development or learning purposes only. Modify subject as necessary.

---

#### 66: Self-Signed Certificate Configuration

```
cd ~
mkdir certs
cd certs
openssl genrsa -out ca-bundle.key 2048
openssl req -new -key ca-bundle.key -out ca-bundle.csr -subj "/C=US/
→ST=CO/L=DEN/O=somecompany/OU=CDN/CN=somecompany.net/
→emailAddress=someuser@somecompany.net"
openssl x509 -req -days 365 -in ca-bundle.csr -signkey ca-bundle.key -
→out ca-bundle.crt
openssl genrsa -out server.key 2048
openssl req -new -key server.key -out server.csr -subj "/C=US/ST=CO/
→L=DEN/O=somecompany/OU=CDN/CN=somecompany.net/
→emailAddress=someuser@somecompany.net"
openssl x509 -req -days 365 -in server.csr -CA ca-bundle.crt -CAkey ca-
→bundle.key -CAcreateserial -out server.crt
mkdir /etc/riak/certs
mv -f server.crt /etc/riak/certs/.
mv -f server.key /etc/riak/certs/.
mv -f ca-bundle.crt /etc/pki/tls/certs/.
```

### Riak Configuration File

The following steps need to be performed on each Riak server in the cluster:

1. Log into Riak server as root
2. Update the following in `riak.conf` to reflect your IP, hostname, and CDN domains and sub-domains:
  - `nodename = riak@a-host.sys.kabletown.net`
  - `listener.http.internal = a-host.sys.kabletown.net:8098` (port can be 80 - This endpoint will not work over HTTPS)
  - `listener.protobuf.internal = a-host.sys.kabletown.net:8087` (can be different port if you want)
  - `listener.https.internal = a-host.sys.kabletown.net:8088` (port can be 443)
3. Update the following in `riak.conf` file to point to your SSL certificate files
  - `ssl.certfile = /etc/riak/certs/server.crt`
  - `ssl.keyfile = /etc/riak/certs/server.key`
  - `ssl.cacertfile = /etc/pki/tls/certs/ca-bundle.crt`

### Enabling TLS 1.1 (required)

1. Add a line at the bottom of the `riak.conf` for TLSv1.1 by setting `tls_protocols.tls1.1 = on`
2. Once the configuration file has been updated restart Riak
3. Consult the [Riak documentation](#) for instructions on how to verify the installed service

### riak-admin Configuration

`riak-admin` is a command line utility used to configure Riak that needs to be run as root on a server in the Riak cluster.

#### See also:

[The riak-admin documentation](#)

#### 67: Traffic Vault Setup with `riak-admin`

```
# This script need only be run on any *one* Riak server in the cluster

# Enable security and secure access groups
riak-admin security enable
riak-admin security add-group admins
```

(continues on next page)

(continued from previous page)

```
riak-admin security add-group keysusers

# User name and password should be stored in the traffic_vault_config_
→section in
# /opt/traffic_ops/app/conf/cdn.conf on the Traffic Ops server (with_
→traffic_vault_backend = riak)
# In this example, we assume the usernames 'admin' and 'riakuser' with
# respective passwords stored in the ADMIN_PASSWORD and RIAK_USER_
→PASSWORD
# environment variables
riak-admin security add-user admin password=$ADMIN_PASSWORD_
→groups=admins
riak-admin security add-user riakuser password=$RIAK_USER_PASSWORD_
→groups=keysusers
riak-admin security add-source riakuser 0.0.0.0/0 password
riak-admin security add-source admin 0.0.0.0/0 password

# Grant privileges to the admins group for everything
riak-admin security grant riak_kv.list_buckets,riak_kv.list_keys,riak_
→kv.get,riak_kv.put,riak_kv.delete on any to admins

# Grant privileges to keysusers group for SSL, DNSSEC, and url_sig_
→keys buckets only
riak-admin security grant riak_kv.get,riak_kv.put,riak_kv.delete on_
→default ssl to keysusers
riak-admin security grant riak_kv.get,riak_kv.put,riak_kv.delete on_
→default dnssec to keysusers
riak-admin security grant riak_kv.get,riak_kv.put,riak_kv.delete on_
→default url_sig_keys to keysusers
riak-admin security grant riak_kv.get,riak_kv.put,riak_kv.delete on_
→default cdn_uri_sig_keys to keysusers
```

#### See also:

For more information on security in Riak, see the [Riak Security documentation](#).

## Traffic Ops Configuration

Before a fully set-up Riak instance may be used as the Traffic Vault backend, it must be added as a server to Traffic Ops. The easiest way to accomplish this is via Traffic Portal at *Configure* → *Servers*, though *servers* may also be used by low-level tools and/or scripts. The Traffic Ops configuration file `/opt/traffic_ops/app/conf/cdn.conf` must be updated to set `traffic_vault_backend` to "riak" and the `traffic_vault_config` to include the correct username and password for accessing the Riak database.

## Configuring Riak Search

In order to more effectively support retrieval of SSL certificates by Traffic Router and *ORT*, the Riak backend for Traffic Vault uses *Riak search*. Riak Search uses *Apache Solr* for indexing and searching of records. This section explains how to enable, configure, and validate Riak Search.

### Riak Configuration

On each Traffic Vault server follow these steps.

1. If Java (JDKv1.8+) is not already installed on your Riak server, install Java

68: Check if Java is Installed, Then Install if Needed

```
# Ensure that this outputs a Java version that is at least 1.8
java -version

# If it didn't, or produced an error because `java` doesn't exist,
# install the correct version
# (OpenJDK is used here because of its permissive license, though OracleJDK
# should work with some tinkering)

# On CentOS/RedHat/Fedora (recommended)
yum install -y java-1.8.0-openjdk java-1.8.0-openjdk-devel

# On Ubuntu/Debian/Linux Mint
apt install -y openjdk-8-jdk

# Arch/Manjaro
pacman -Sy jdk8-openjdk
```

2. Enable search in `riak.conf` by changing the `search = off` setting to `search = on`
3. Restart Riak to propagate configuration changes

## 69: Restarting Riak on *systemd(1)* Systems

```
systemctl restart riak
```

## One-time Configuration

After Riak has been configured to use Riak Search, permissions still need need to be updated to allow users to utilize this feature. Unlike actually setting up Riak Search, the permissions step need only be done on any *one* of the Riak servers in the cluster.

1. Use `riak-admin` to grant `search.admin` permissions to the “admin” user and `search.query` permissions to **both** the “admin” user and the “riakuser” user. The “admin” user will also require `search.admin` permissions on the `schema` (in addition to `index`) and `riak_core.set_bucket` permissions on any.

## 70: Setting up Riak Search Permissions

```
riak-admin security grant search.admin on schema to admin
riak-admin security grant search.admin on index to admin
riak-admin security grant search.query on index to admin
riak-admin security grant search.query on index sslkeys to ↵
↵admin
riak-admin security grant search.query on index to riakuser
riak-admin security grant search.query on index sslkeys to ↵
↵riakuser
riak-admin security grant riak_core.set_bucket on any to ↵
↵admin
```

2. Add the search schema to Riak. This schema is a simple Apache Solr configuration file which will index all records on CDN, hostname, and *Delivery Service*. The file can be found at `traffic_ops/app/config/misc/riak_search/sslkeys.xml` in the Traffic Control repository.

## 71: Adding the GitHub-hosted Search Schema to Riak

```
# Obtain the configuration file - in this example by ↵
↵downloading it from GitHub
wget https://raw.githubusercontent.com/apache/
↵trafficcontrol/master/traffic_ops/app/conf/misc/riak_
↵search/sslkeys.xml

# Upload the schema to the Riak server using its API
# Note that the assumptions made here are that the "admin" ↵
↵user's password is "pass"
# and the server is accessible at port 8088 on the ↵
↵hostname "trafficvault.infra.ciab.test"
curl --tlsv1.1 --tls-max 1.1 -kvsX PUT "https://
↵admin:pass@trafficvault.infra.ciab.test:8088/index/
↵schema/sslkeys" -H "Content-Type: application/xml" -d ↵
```

(Continues on next page)

(continued from previous page)

3. Add the search index to Riak.

#### 72: Adding the Search Index to Riak Via its API

```
# Note that the assumptions made here are that the "admin"
↪user's password is "pass"
# and the server is accessible at port 8088 on the
↪hostname "trafficvault.infra.ciab.test"
curl --tlsv1.1 --tls-max 1.1 -kvsX PUT "https://
↪admin:pass@trafficvault.infra.ciab.test:8088/search/
↪index/sslkeys" -H 'Content-Type: application/json' -d '{
↪"schema":"sslkeys"}'
```

4. Associate the sslkeys index to the ssl bucket in Riak

#### 73: Using the Riak API to Create an Index-to-Bucket Association for sslkeys

```
# Note that the assumptions made here are that the "admin"
↪user's password is "pass"
# and the server is accessible at port 8088 on the
↪hostname "trafficvault.infra.ciab.test"
curl --tlsv1.1 --tls-max 1.1 -kvs -XPUT "https://
↪admin:pass@trafficvault.infra.ciab.test:8088/buckets/ssl/
↪props" -H'content-type:application/json' -d'{"props":{"
↪"search_index":"sslkeys"}}'
```

5. To validate the search is working run a query against the Riak database server, or use the Traffic Ops API endpoint: `cdns/name/{{name}}/sslkeys`

#### 74: Validate Riak Search is Working

```
# Note that the assumptions made here are that the "admin"
↪user's password is
# "pass", the Traffic Vault server's Riak database is
↪accessible at port 8088 on
# the hostname "trafficvault.infra.ciab.test", $COOKIE
↪contains a valid
# Mojolicious cookie for a Traffic Ops user with proper
↪permissions, and the
# Traffic Ops server is available at the hostname
↪"trafficops.infra.ciab.test"

# Verify by querying Riak directly
curl --tlsv1.1 --tls-max 1.1 -kvs "https://
↪admin:password@trafficvault.infra.ciab.test:8088/search/
↪query/sslkeys?wt=json&q=cdn:CDN-in-a-Box" (continues on next page)
```



(continued from previous page)

```
# Verify using the Traffic Ops API
curl -Lvs -H "Cookie: $COOKIE" https://trafficops.infra.
ciab.test/api/4.0/cdns/name/mycdn/sslkeys
```

### 3.1.10 Quick How To Guides

Traffic Control is a complicated system, and documenting it is not trivial. Sometimes a picture says more than a thousand words, so here are some screenshot-based tutorials on how to use some of the more involved features.

#### Configure Anonymous Blocking

**Note:** Anonymous Blocking is only supported for HTTP delivery services. You will need access to a database that provides anonymous IP statistics ([Maxmind's database](#) is recommended, as this functionality was built specifically to work with it.)

1. Prepare the Anonymous Blocking configuration file. Anonymous Blocking uses a configuration file in JSON format to define blocking rules for *Delivery Services*. The file needs to be put on an HTTP server accessible to Traffic Router.

#### 75: Example Configuration JSON

```
{
  "customer": "YourCompany",
  "version": "1",
  "date" : "2017-05-23 03:28:25",
  "name": "Anonymous IP Blocking Policy",

  "anonymousIp": { "blockAnonymousVPN": true,
    "blockHostingProvider": true,
    "blockPublicProxy": true,
    "blockTorExitNode": true},

  "ip4Whitelist": ["192.168.30.0/24", "10.0.2.0/24", "10.
1.1.1/32"],
  "ip6Whitelist": ["2001:550:90a::/48", "::1/128"],
  "redirectUrl": "http://youvebeenblocked.com"
}
```

**anonymousIp** Contains the types of IPs which can be checked against the Anonymous IP Database. There are 4 types of IPs which can be checked: VPN (Virtual Private Network)s, Hosting Providers, Public Proxies, and TOR “Exit Nodes”. Each type of IP can be enabled or disabled. If the

value is true, IPs matching this type will be blocked when the feature is enabled in the *Delivery Service*. If the value is false, IPs which match this type will not be blocked. If an IP matches more than 1 type and any type is enabled, the IP will be blocked.

**redirectUrl** The URL that will be returned to the blocked clients. Without a *redirectUrl*, the clients will receive an HTTP response code 403 Forbidden. With a *redirectUrl*, the clients will be redirected with an HTTP response code 302 Found.

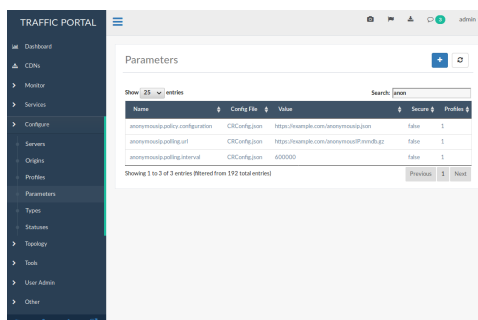
**ipWhiteList** An optional element. It includes a list of CIDR blocks indicating the IPv4 and IPv6 subnets that are allowed by the rule. If this list exists and the value is not null, client IPs will be matched against the CIDR list, and if there is any match, the request will be allowed. If there is no match in the white list, further anonymous blocking logic will continue.

2. Add the following three Anonymous Blocking *Parameters* in Traffic Portal with the “CR-Config.json” *Config File*, and ensure they are assigned to all of the Traffic Routers that should perform Anonymous Blocking:

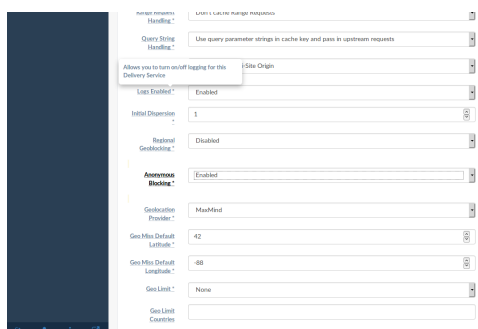
**anonymousip.policy.configuration** The URL of the Anonymous Blocking configuration file. Traffic Router will fetch the file from this URL.

**anonymousip.polling.url** The URL of the Anonymous IP Database. Traffic Router will fetch the file from this URL.

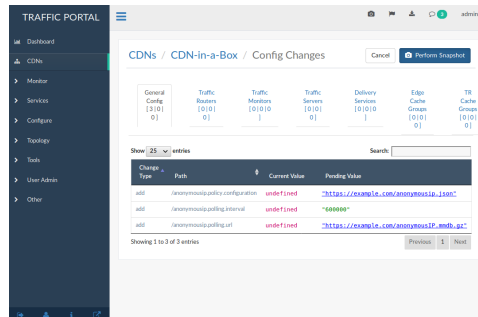
**anonymousip.polling.interval** The interval that Traffic Router polls the Anonymous Blocking configuration file and Anonymous IP Database.



3. Enable Anonymous Blocking for a *Delivery Service* using the *Delivery Services view* in *Traffic Portal* (don't forget to save changes!)



4. Go to *the Traffic Portal CDNs view*, click on *Diff CDN Config Snapshot*, and click *Perform Snapshot*.



## Traffic Router Access Log

Anonymous Blocking extends the field of `rtype` and adds a new field `ANON_BLOCK` in the Traffic Router `access.log` file to help monitor this feature. If the `rtype` in an access log is `ANON_BLOCK` then the client's IP was found in the Anonymous IP Database and was blocked.

**See also:**

*Troubleshooting and log files*

## Configure Cache Group Fallbacks

**See also:**

*Cache Groups*

1. Go to 'Topology', click on *Cache Groups*, and click on your desired *Cache Group* or click the + button to create a new *Cache Group*.

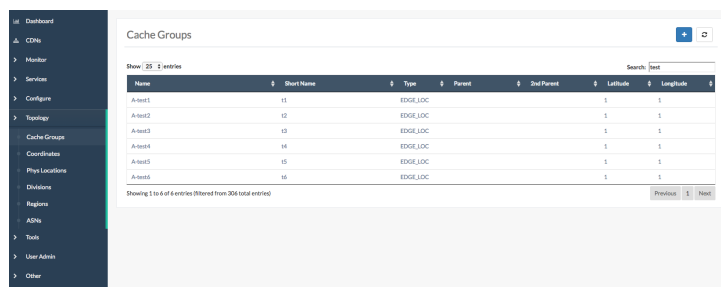


Fig. 22: Cache Groups Page

2. Verify that the *Cache Group* is of type `EDGE_LOC`. *Cache Group* Failovers only apply to `EDGE_LOC` *Cache Groups*.

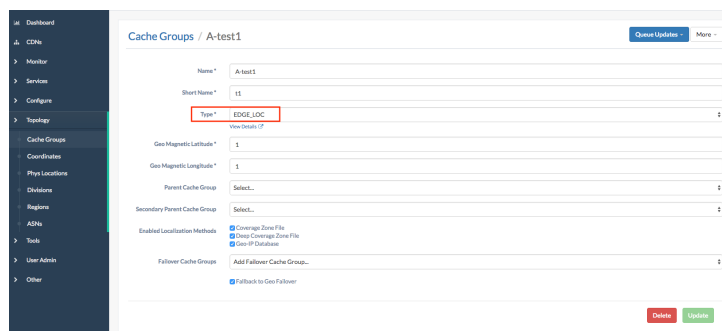


Fig. 23: Cache Group Details Page

- Once `EDGE_LOC` is selected, the Failover Cache Groups section will appear at the bottom of the page. If you are editing an existing *Cache Group*, then the current Failovers will be listed. If creating a new *Cache Group*, the Fallback to Geo Failover box will default to be checked.

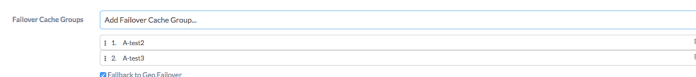


Fig. 24: Failover Cache Groups Section of Cache Group Details Page

- To add a new Failover to the list, select the “Add Failover *Cache Group*” drop down and choose which *Cache Group* you would like. While in the drop down, you can also type in order to search.

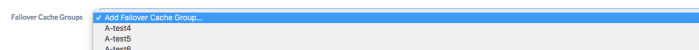


Fig. 25: Add New Failover Cache Group Section of Cache Group Details Page

- The order of the Failovers is important. If you want to reorder the Failovers, you can drag and drop them into a new position. A red line will appear to show where the Failover will be dropped.
- To remove a Failover, click the trash can symbol on the right hand side of the list.
- Click the *Update* button (if editing existing *Cache Group*) or the *Create* button (if creating new *Cache Group*) in order to save the Failovers to the *Cache Group*.

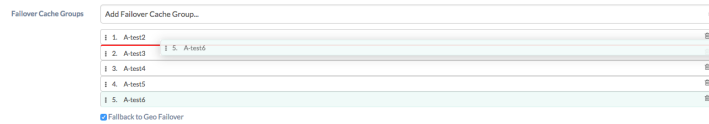


Fig. 26: Failover Cache Groups Section Drag and Drop Functionality

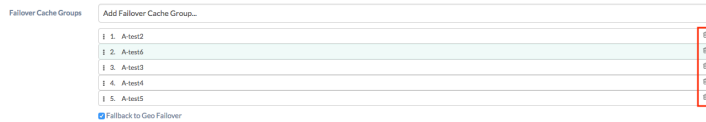


Fig. 27: Failover Cache Groups Delete

## Forcing Content Invalidation

Invalidating content on the CDN is sometimes necessary when the *Origin* was mis-configured and something is cached in the CDN that needs to be removed.

### Implementation Detail

Given the size of a typical Traffic Control CDN and the amount of content that can be cached in it, removing the content from all the caches may take a long time. To speed up content invalidation, Traffic Control does not try to remove the content from the caches, but it makes the content inaccessible using the `regex_revalidate` plugin for Apache Traffic Server. This forces a “re-validation” of the content.

To invalidate content for a specific *Delivery Service*, follow these steps:

1. Select the desired *Delivery Service* from the *Delivery Services* view of Traffic Portal

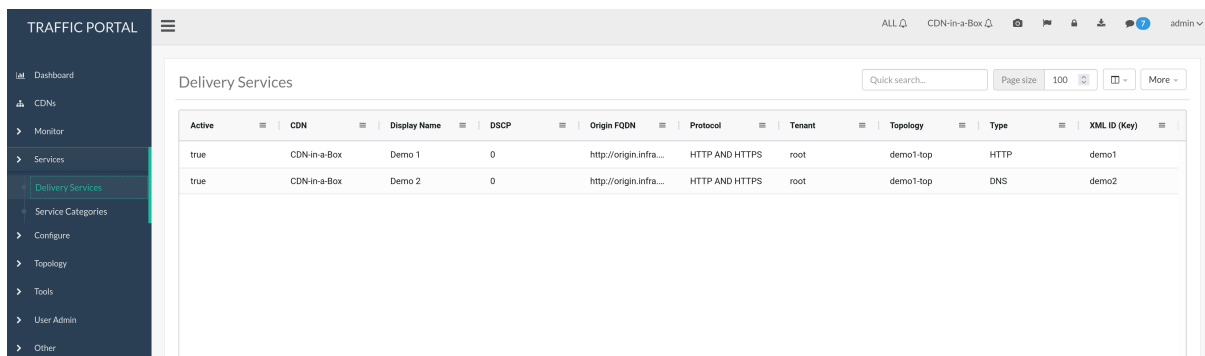


Fig. 28: The Traffic Portal Delivery Services view

2. From the *More* drop-down menu, select *Manage Invalidation Requests*
3. From the *More* drop-down menu on this page, select *Create Invalidation Request*

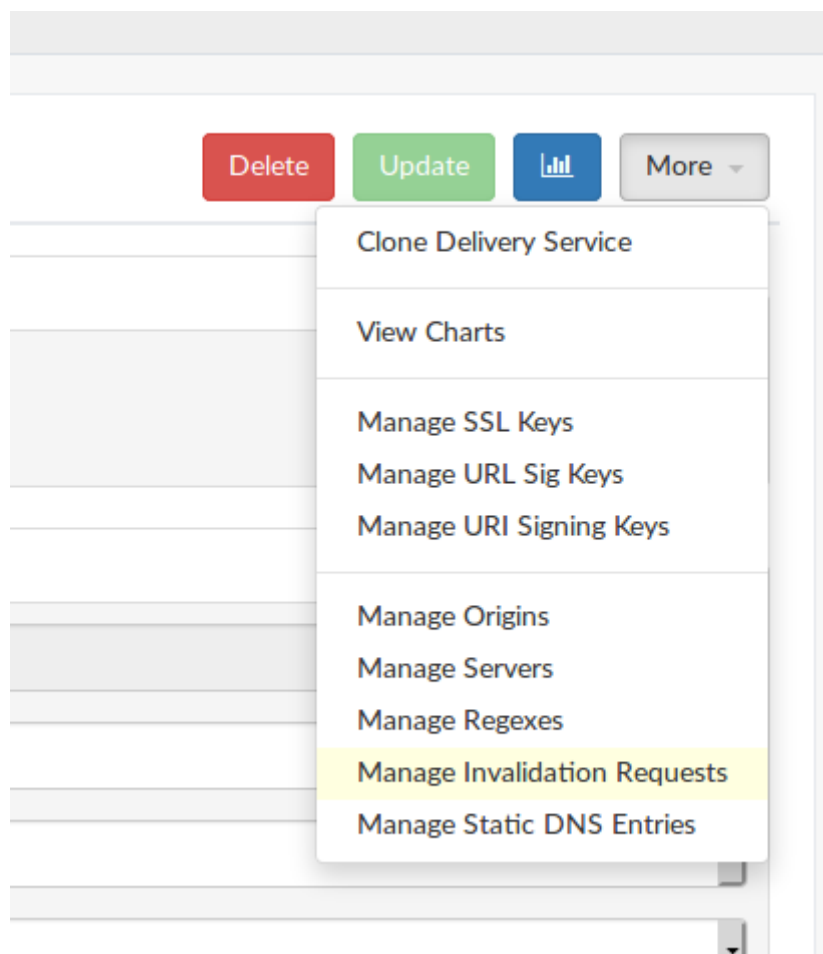


Fig. 29: Select ‘Manage Invalidation Requests’

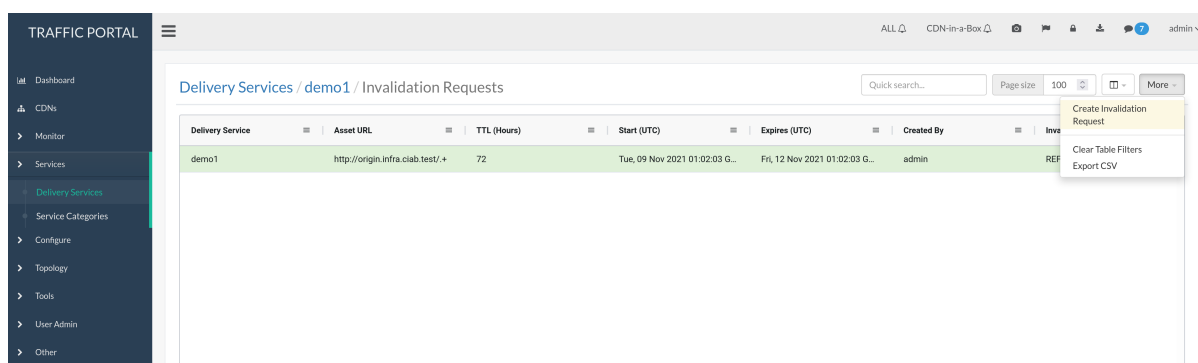


Fig. 30: Select ‘Create Invalidation Request’

4. Fill out this form. The “Path Regex” field should be a [PCRE](#)-compatible regular expression that matches all content that must be invalidated - and should **not** match any content that must *not* be invalidated. “TTL (hours)” specifies the number of hours for which the invalidation should remain active. Best practice is to set this to the same as the content’s cache lifetime (typically set in the *Origin*’s Cache-Control response header). *Invalidation Type* describes how content will be invalidated.

The screenshot shows the 'Traffic Portal' interface. On the left is a dark sidebar with a menu including 'Dashboard', 'CDNs', 'Monitor', 'Services' (highlighted), 'Service Categories', 'Configure', 'Topology', 'Tools', 'User Admin', and 'Other'. The main content area has a breadcrumb trail: 'Delivery Services / demo1 / Invalidation Requests / New'. Below this is a form with three input fields: 'Path Regex \*' containing '/.\*\jpg', 'TTL (hours) \*' containing '1', and 'Invalidation Type \*' set to 'REFRESH'. A green 'Create' button is positioned at the bottom right of the form.

Fig. 31: The ‘new content invalidation submission’ Form

5. Click on the *Create* button to finalize the content invalidation.

## Configure DNSSEC

In order to support [DNSSEC](#) in Traffic Router, Traffic Portal provides some actions for managing DNSSEC keys for a CDN and associated *Delivery Services*. DNSSEC keys consist of a KSK which is used to sign other DNSKEY records as well as a ZSK which is used to sign other records. DNSSEC keys are stored in Traffic Vault and should only be accessible to Traffic Ops. Other applications needing access to this data, such as Traffic Router, must use the *Traffic Ops API* to retrieve this information.

**See also:**

[DNSSEC](#)

---

**Note:** DNSSEC is only supported for DNS-Routed *Delivery Services*.

---

## Creating Keys

**Note:** In order for Traffic Ops to successfully store keys in Traffic Vault, at least one Traffic Vault server needs to be configured in Traffic Ops.

1. Go to *CDNs* and click on the desired CDN.

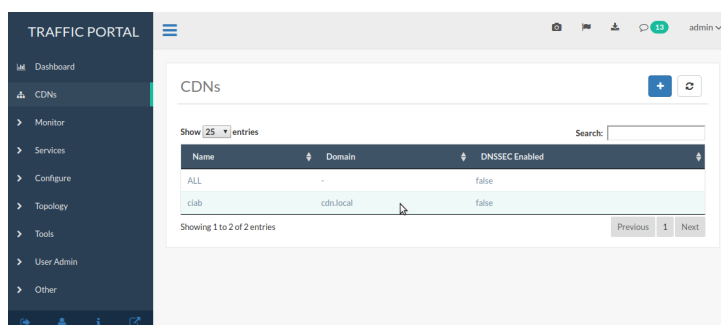


Fig. 32: CDNs Page

2. Click on *More* → *Manage DNSSEC Keys*.

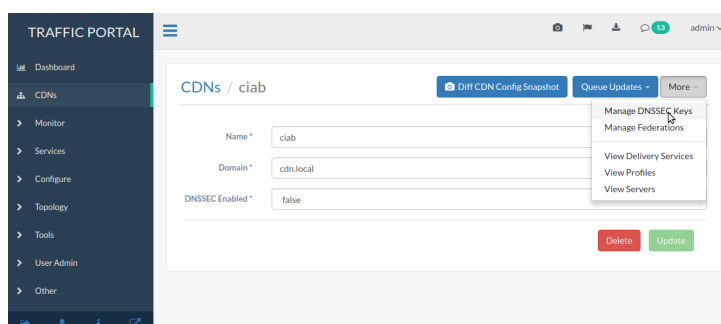


Fig. 33: CDN Details Page

3. Click on the *Generate DNSSEC Keys* button.
4. A modal will pop up asking you to confirm that you want to proceed.
5. Input the required information (reasonable defaults should be generated for you). When done, click on the green *Generate* button.

**Note:** Depending upon the number of *Delivery Services* in the CDN, generating DNSSEC keys may take several seconds.



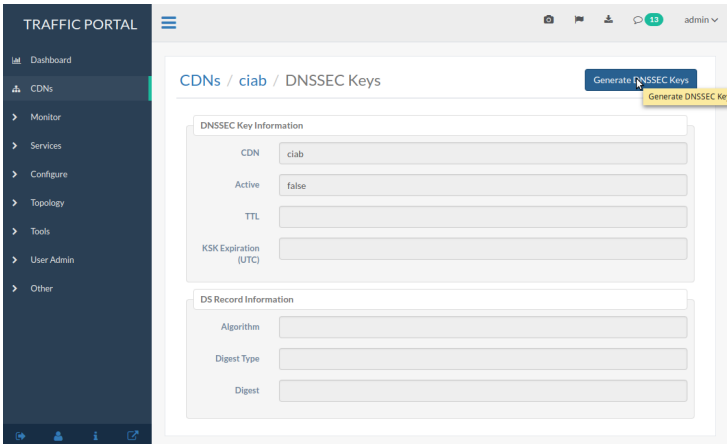


Fig. 34: DNSSEC Key Management Page

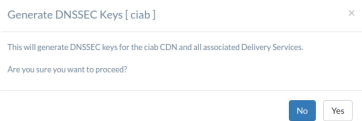


Fig. 35: Confirmation Modal

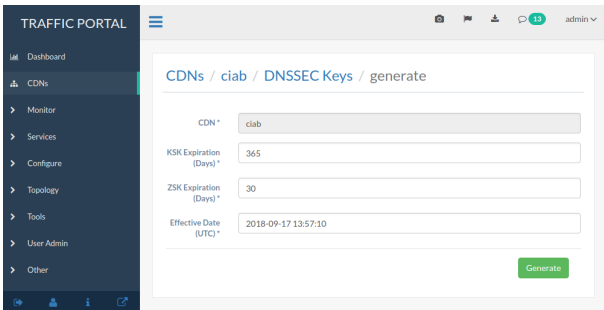


Fig. 36: DNSSEC Key Generation Page

6. You will be prompted to confirm the changes by typing the name of the CDN into a text box. After doing so, click on the red *Confirm* button.

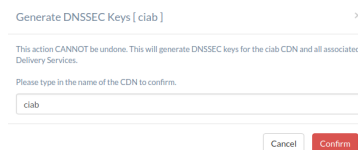


Fig. 37: DNSSEC Key Change Confirmation

7. In order for DNSSEC to work properly, the DS (Delegation of Signing) Record information needs to be added to the parent zone of the CDN's domain (e.g. If the CDN's domain is 'ciab.cdn.local' the parent zone is 'cdn.local'). If you control your parent zone you can enter this information yourself, otherwise you will need to work with your DNS team to get the DS Record added to the parent zone.

### Enabling and Disabling DNSSEC on a CDN

1. Once DS Record information has been added to the parent zone, DNSSEC needs to be activated for the CDN so that Traffic Router will sign responses. Go to the CDN details page for this CDN, and set the 'DNSSEC Enabled' field to 'true' (or 'false' to disable DNSSEC), then click the green *Update* button.

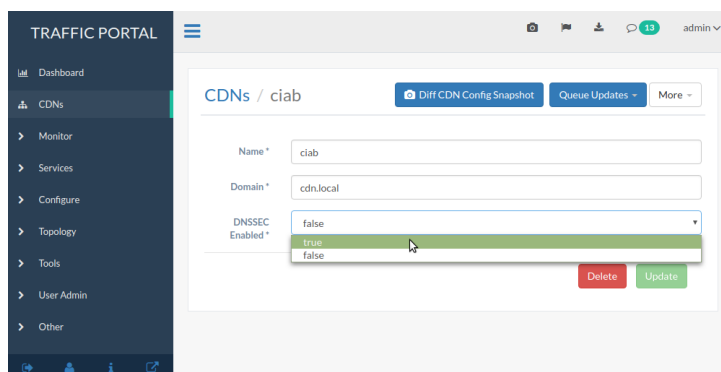


Fig. 38: Change 'DNSSEC Enabled' to 'true'

2. DNSSEC should now be active (or inactive, if disabled) on your CDN and Traffic Router should be signing responses. This should be tested e.g. with this *dig(1)* command: `dig edge.cdn.local. +dnssec`.

## Regenerating KSKs

When KSK expiration is approaching (default 365 days), it is necessary to manually generate a new KSK for the TLD and add the DS Record to the parent zone. In order to avoid signing errors, it is suggested that an effective date is chosen which allows time for the DS Record to be added to the parent zone before the new KSK becomes active.

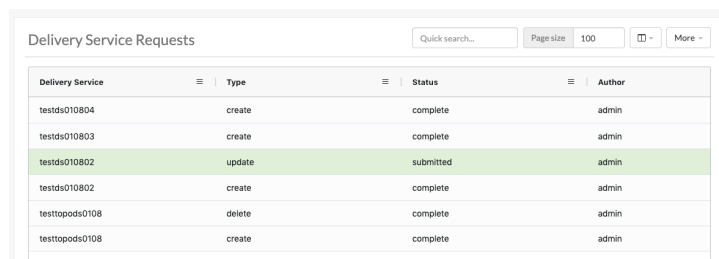
## Regenerating Keys

It's also possible to re-do the generation of DNSSEC keys once they have already been generated. This can be done at any point after [Creating Keys](#) by clicking on the then-enabled *Regenerate Keys* button on the CDN's details page.

## Delivery Service Requests

When enabled in `traffic_portal_properties.json`, Delivery Service Requests are created when *all* users attempt to create, update or delete a [Delivery Service](#). This allows users with higher level permissions (“operations” or “admin”) to review the changes for completeness and accuracy before deploying the changes. In addition, most [Delivery Service](#) changes require configuration updates (i.e. [Queue Updates](#)) and/or a CDN [Snapshot](#). Both of these actions are reserved for users with elevated permissions.

A list of the Delivery Service requests associated with your [Tenant](#) can be found under *Services* → *Delivery Service Requests*



Delivery Service	Type	Status	Author
testds010804	create	complete	admin
testds010803	create	complete	admin
testds010802	update	submitted	admin
testds010802	create	complete	admin
testtopods0108	delete	complete	admin
testtopods0108	create	complete	admin

Fig. 39: Example Delivery Service Request Listing

## Who Can Create a Delivery Service Request and How?

Users with the Portal [Role](#) (or above) can create Delivery Service Requests by doing one of three things:

- Creating a new [Delivery Service](#)
- Updating an existing [Delivery Service](#)
- Deleting an exiting [Delivery Service](#)

By performing one of these actions, a Delivery Service Request will be created for you with a status of ‘draft’ or ‘submitted’. You determine the status of your request upon submission.

Only change the status of your request to ‘submitted’ once the request is ready for review and deployment.

### Who Can Fulfill a Delivery Service Request and How?

Users with elevated permissions (Operations or above) can fulfill (apply the changes) or reject the Delivery Service Request. In fact, they can do all of the following:

**Update the contents of the Delivery Service Request** This will update the “Last Edited By” field to indicate who last updated the request.

**Assign or Unassign the Delivery Service Request** Assignment is currently limited to current user. This is optional as fulfillment will auto-assign the request to the user doing the fulfillment.

**Reject the Delivery Service Request** Rejecting a Delivery Service Request will set status to ‘rejected’ and the request can no longer be modified. This will auto-assign the request to the user doing the rejection.

**Fulfill the Delivery Service Request** Fulfilling a Delivery Service Request will show the requested changes and, once committed, will apply the desired changes and set status to ‘pending’. The request is pending because many types of changes will require *cache server* configuration updates (i.e. *Queue Updates*) and/or a CDN *Snapshot*. Once *Queue Updates* and/or CDN *Snapshot* is complete, the request should be marked ‘complete’.

**Complete the Delivery Service Request** Only after the Delivery Service Request has been fulfilled and the changes have been applied can a Delivery Service Request be marked as ‘complete’. Marking a Delivery Service Request as ‘complete’ is currently a manual step because some changes require *cache server* configuration updates (i.e. *Queue Updates*) and/or a CDN *Snapshot*. Once that is done and the changes have been deployed, the request status should be changed from ‘pending’ to ‘complete’.

---

**Note:** Only the user that fulfilled the delivery service request can mark a delivery service as ‘complete’. This prevents other users from interfering in the process and marking delivery services as ‘complete’ when further action is required for the changes to truly be deployed. However, in `traffic_portal_properties.json`, users with the ‘overrideRole’ are given the ability to mark any delivery service requests as ‘complete’.

---

**Delete the Delivery Service request** Delivery Service Requests with a status of ‘draft’ or ‘submitted’ can always be deleted entirely if appropriate.

## Configure Federations

1. Create a user with a federations role (*User Admin* → *Users* → '+' button). This user will need the ability to:
  - Create/edit/delete federations
  - Add IPV4 resolvers
  - Add IPV6 resolvers

The screenshot shows a 'Add User' form with the following fields and values:

Field	Value
* Full Name	Federation User
* Username	federation_user1
* Email	federation_user@mega-cdn.net
* Password	.....
* Confirm Password	.....
Role	federation
Phone Number	
Company	
Address Line 1	
Address Line 2	
City	
State or Province	
Postal Code	
Country	
UID	
GID	
Last Updated	

Buttons: Add User, Close

2. As a user with administrative privileges, create a Federation Mapping by going to *Services* → *:term: 'Delivery Services' → More → Federations`* and then clicking *Add Federation Mapping*.
3. Choose the *Delivery Service* to which the federation will be mapped and assign it to the Federation-role user; click *Add*.
4. After the Federation is added, Traffic Ops will display the Federation. Changes can be made at this time or the Federation can be deleted. Notice that no resolvers have been added to the Federation yet. This can only be done by the Federation-role user to whom the Federated *Delivery Service* was assigned. If no further action is necessary, the *Close* button will close the window and display the list of all Federations.

**Add Federation Mapping**

\* CNAME:

Description:

\* TTL:

\* Role:

\* Delivery Services:

\* User:

5. The federation user logs into either the Traffic Ops API or the Traffic Portal UI and stores the Mojolicious cookie. The Mojolicious cookie can be obtained manually using the debug tools on a web browser or via a command line utility like *curl* (1).

#### 76: Example cURL Command

```
curl -i -XPOST "http://localhost:3000/api/4.0/user/login" -
-H "Content-Type: application/json" -d '{ "u":
"federation_user1", "p": "password" }'
```

Federation Mapping

Successfully added Federation!

\* CNAME

img.mega-cdn.net.

Description

Federation Mapping for img.mega-cdn.

\* TTL

30

\* Role

federation

\* Delivery Services

images-c1

\* User

Federation User (federation\_user1) -

Resolvers

Delete

Save

Close

77: Example API Response

```
HTTP/1.1 200 OK
Date: Wed, 02 Dec 2015 21:12:06 GMT
Content-Length: 65
Access-Control-Allow-Credentials: true
Content-Type: application/json
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov
↳2019 17:40:54 GMT; Max-Age=3600; HttpOnly
X-Server-Name: traffic_ops_golang/
Access-Control-Allow-Headers: Origin, X-Requested-With,
↳Content-Type, Accept
Cache-Control: no-cache, no-store, max-age=0, must-
↳revalidate
Connection: keep-alive
Access-Control-Allow-Origin: http://localhost:8080

{"alerts":[{"level":"success","text":"Successfully logged
↳in."}]}
```

6. The federation user sends a request to Traffic Ops to add IPV4 and/or IPV6 resolvers

78: Example cURL Command

```
curl -ki -H "Cookie:
↳mojolicious=eyJleHBpcmVzIjoxNDQ5MTA1MTI2LCJhdXRoX2RhdGEiOiJmZWRLcmF0aW9uX
↳--06b4f870d809d82a91433e92eae8320875c3e8b0;" -XPUT
↳'http://localhost:3000/api/4.0/federations' -d '
{"federations": [
  { "deliveryService": "images-c1",
    "mappings":
      { "resolve4": [ "8.8.8.8/32", "8.8.4.4/32" ],
        "resolve6": ["2001:4860:4860::8888/128",
↳"2001:4860:4860::8844"]
      }
    }
}]}'
```

79: Example API Response

```
HTTP/1.1 200 OK
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov
↳2019 17:40:54 GMT; Max-Age=3600; HttpOnly
X-Server-Name: traffic_ops_golang/
Date: Wed, 02 Dec 2015 21:25:42 GMT
Content-Length: 74
```

(continues on next page)



(continued from previous page)

```
Access-Control-Allow-Credentials: true
Content-Type: application/json
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Cache-Control: no-cache, no-store, max-age=0, must-
    ↪revalidate
Access-Control-Allow-Origin: http://localhost:8080
Connection: keep-alive
Access-Control-Allow-Headers: Origin, X-Requested-With,
    ↪Content-Type, Accept

{"response": "federation_user1 successfully created,
    ↪federation resolvers."}
```

7. The resolvers added by the Federation-user will now visible in Traffic Portal.

**Federation Mapping**

\* CNAME:

Description:

\* TTL:

\* Role:

\* Delivery Services:

\* User:

**Resolvers**

- ▼ resolve4
  - 8.8.8.8/32
  - 8.8.4.4/32
- ▼ resolve6
  - 2001:4860:4860::8888/128
  - 2001:4860:4860::8844/128

Buttons: Delete, Save, Close

Any requests made from a client that resolves to one of the federation resolvers will now be given a CNAME Record from Traffic Router.

### 80: Example DNS request (via dig)

```
dig @tr.kabletown.net foo.images-c1.kabletown.net
```

### 81: Example Resolver Response

```
; <<>> DiG 9.7.3-RedHat-9.7.3-2.el6 <<>> @tr.kabletown.net.
; →foo.images-c1.kabletown.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45110
;; flags: qr rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
; →ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;foo.images-c1.kabletown.net.    IN A

;; ANSWER SECTION:
foo.images-c1.kabletown.net.    30 IN CNAME img.mega-cdn.net.

;; Query time: 9 msec
;; SERVER: 10.10.10.10#53(10.10.10.10)
;; WHEN: Wed Dec  2 22:05:26 2015
;; MSG SIZE rcvd: 84
```

## Creating the CentOS Kickstart File

The Kickstart file is a text file, containing a list of items, each identified by a keyword. This file can be generated using the [Red Hat Kickstart Configurator application](#), or it can be written from scratch. The Red Hat Enterprise Linux installation program also creates a sample Kickstart file based on the options selected during installation. It is written to the file `/root/anaconda-ks.cfg` in this case. This file is editable using most text editors.

## Generating a System Image

1. Create a Kickstart file.
2. Create a boot media with the Kickstart file or make the Kickstart file available on the network.
3. Make the installation tree available.
4. Start the Kickstart installation.

## 82: Creating a New System Image Definition Tree from an Existing One

```
# Starting from the Kickstart root directory (`/var/www/files` by
↳ default)
mkdir newdir
cd newdir/

# In this example, the pre-existing system image definition tree is
↳ for CentOS 7.4 located in `centos74`
cp -r ../centos74/* .
vim ks.src
vim isolinux/isolinux.cfg
cd ..
vim osversions.json
```

`ks.src` is a standard, Kickstart-formatted file that will be used to create the Kickstart (`ks.cfg`) file for the install whenever a system image is generated from the source tree. `ks.src` is a template - it will be overwritten by any information set in the form submitted from *Tools* → *Generate ISO* in Traffic Portal. Ultimately, the two are combined to create the final Kickstart file (`ks.cfg`).

---

**Note:** It is highly recommended for ease of use that the system image source trees be kept under 1GB in size.

---

### See also:

For in-depth instructions, please see [Kickstart Installation](#) in the Red Hat documentation.

### kickstart.files.location

The Kickstart root directory used by *Traffic Ops* (`/var/www/files` by default) can be changed by setting the `kickstart.files.location` *Parameter*.

## Configure Multi-Site Origin

The following steps will take you through the procedure of setting up an MSO. #. Create *Cache Groups* for the origin locations, and assign the appropriate parent-child relationship between the Mid-tier *Cache Group*(s) and origin *Cache Groups*. Each Mid-tier *Cache Group* can be assigned a primary and secondary origin parent *Cache Group*. When the *Cache Group* parent configuration is generated, origins in the primary *Cache Groups* will be listed first, followed by origins in the secondary *Cache Group*. Origin servers assigned to the *Delivery Service* that are assigned to neither the primary nor secondary *Cache Groups* will be listed last.

Health	Delivery Services	Servers	Parameters	Tools	Misc	Change Log	Help	UTC: 20:52:58
Search: <input type="text" value="orig"/>								
Name	Short Name	Type	Magnet Latitude	Magnet Longitude	Parent Loc	Last Updated	Cache Groups	
mid-east	east	MID_LOC	0	0	origin-east	2015-06-25 20:52:37		
mid-west	west	MID_LOC	0	0	origin-west	2015-06-25 20:52:37		
origin-east	org-east	ORG_LOC	0	0		2015-06-25 20:52:37		
origin-west	org-west	ORG_LOC	0	0		2015-06-25 20:52:37		

Showing 1 to 4 of 4 entries (Filtered from 12 total entries)

[Add Cache Group](#)

1. Create a profile to assign to each of the origins:

Health	Delivery Services	Servers	Parameters	Tools	Misc	Change Log	Help	UTC: 20:53:46
Search: <input type="text" value="ORG"/>								
Profile name	Parameter name	Config file name	Value					
ORG1_CDN1	domain_name	CRConfig.xml	cdn1.kabletown.net					
ORG1_CDN1	GeoLocationURL	CRConfig.xml	http://www.cdnlab.kabletown.net:8080/GeoLocCity.dat.gz					
ORG1_CDN1	weight	parent.config	1.0					
ORG2_CDN1	domain_name	CRConfig.xml	cdn1.kabletown.net					
ORG2_CDN1	GeoLocationURL	CRConfig.xml	http://www.cdnlab.kabletown.net:8080/GeoLocCity.dat.gz					
ORG2_CDN1	weight	parent.config	1.0					

Showing 1 to 6 of 6 entries (Filtered from 3,996 total entries)

[Add Parameter](#)

2. Create server entries for the origination vips:

Health	Delivery Services	Servers	Parameters	Tools	Misc	Change Log	Help	UTC: 20:54:14
Search: <input type="text" value="ORG"/>								
Host Name	Domain Name	Cache group	Location	IPAddr	IPv6 Addr	Status	Profile	IL0
org1	kabletown.net	origin-east	plocation-nyc-1	10.11.10.2		CCR_IGNORE	ORG1_CDN1	
org2	kabletown.net	origin-west	plocation-nyc-1	10.11.12.2		CCR_IGNORE	ORG2_CDN1	

Showing 1 to 2 of 2 entries (Filtered from 83 total entries)

[Add Server](#)

3. Check the multi-site check box in the *Delivery Service* screen:
4. Assign the org servers to the *Delivery Service* that will have the multi site feature. Origin servers assigned to a *Delivery Service* with multi-site checked will be assigned to be the origin servers for this *Delivery Service*.

**Note:** “Origin Server Base URL” uniqueness: In order to enable Mid-tier *Cache Group* to distinguish *Delivery Services* by different MSO algorithms while performing parent fail-over, it requires that OSBU (Origin Server Base URL) for each MSO-enabled *Delivery Service* is unique. This means that the OSBU of an MSO-enabled *Delivery Service* should be different from the OSBUs of any other *Delivery Service*, regardless of whether they are MSO-enabled or not. The exceptions to this rule are:

- If there are multiple CDNs created on the same Traffic Ops, *Delivery Services* across different CDNs may have the same OSBU configured.
- If several *Delivery Services* in the same CDN have the same MSO algorithm configured, they may share the same OSBU.
- If delivery services are assigned with different Mid-tier *Cache Groups* respectively, they can share the same OSBU.
- This OSBU must be valid - ATS will perform a DNS lookup on this FQDN even if IPs, not DNS, are used in the `parent.config`.

* CDN	cdn1
* Content Routing Type	HTTP
* Protocol	http
* DSCP Tag	0 - Best Effort
Signed URLs	No
Query String Handling	0 - use qstring in cache key, and pass up
Geo Limit?	None
Bypass FQDN	
Initial Dispersion	1 - Off
IPv6 Routing Enabled?	No
Range Request Handling	0 - Don't cache
Delivery Service DNS TTL	3600
* Origin Server Base URL	http://movies.origin.kabletown.net
* Use Multi Site Origin Feature	Yes

Delivery Service Server Assignments for movies-c1 (http://movies.origin.kabletown.net)

EDGE1\_CDN1\_421\_SSL

us-ca-losangeles

us-co-denver

us-il-chicago

atsec-chi-00

atsec-chi-01

atsec-chi-02

atsec-chi-03

good-host

us-ny-newyork

us-pa-philadelphia

us-tx-houston

Save

Close

- The OSBU entered as the “Origin Server Base URL” will be sent to the origins as a host header. All origins must be configured to respond to this host.

5. Create a delivery service profile. This must be done to set the MSO algorithm. Also, as of ATS 6.x, multi-site options must be set as parameters within the `parent.config`. Header rewrite parameters will be ignored. See [ATS parent.config](#) for more details. These *Parameters* are now handled by the creation of a *Delivery Service Profile*.

- a) Create a *Profile* of the *Type* DS\_PROFILE for the *Delivery Service* in question.

- b) Click *Show profile parameters* to bring up the *Parameters* screen for the *Profile*. Create the following *Parameters*:

Parameter Name	Config File Name	Value	ATS parent.config value
mso.algorithm	parent.config	true, false, strict, consistent_hash	round_robin
mso.parent_retry	parent.config	simple_retry, both, unavailable_server_retry	parent_retry
mso.unavailable_server_retry_response_codes	parent.config	list of response codes, eg “500,502,503”	defaults to the value in records.config when unused.
mso.max_simple_retries	parent.config	Number of retries made after a 4xx error	defaults to the value in records.config when unused.
mso.max_unavailable_server_retries	parent.config	Number of retries made after a 5xx error	defaults to the value in records.config when unused.

- c) In the *Delivery Service* page, select the newly created DS\_PROFILE and save the *Delivery Service*.

6. Turn on `parent_proxy_routing` in the MID *Profile*.

Profile name	Parameter name	Config file name	Value	Secure
DS_example	ms.algorithm	parent.config	false	no
DS_example	ms.parent_retry	parent.config	both	no
DS_example	ms.unavailable_server_retry_responses	parent.config	"500,502,503"	no

**Note:** Support for multisite configurations with single-layer CDNs is now available. If a *Cache Groups* defined parents are either blank or of the type ORG\_LOC, that *cache server's* parent.config will be generated as a top layer cache, even if it is an edge. In the past, parent.config generation was strictly determined by cache type. The new method examines the parent *Cache Group* definitions and generates the parent.config accordingly.

## Configure OAuth Login

An opt-in configuration for SSO using OAuth is supported and can be configured through the `/opt/traffic_portal/public/traffic_portal_properties.json` and `/opt/traffic_ops/app/conf/cdn.conf` files. OAuth uses a third party provider to authenticate the user. If enabled, the Traffic Portal Login page will still accept username and password for local accounts but will also allow authentication using OAuth. This will redirect to the `oAuthUrl` from `/opt/traffic_portal/public/traffic_portal_properties.json` which will authenticate the user then redirect to the new `/sso` page with an authorization code. The new `/sso` page will then construct the full URL to exchange the authorization code for a JSON Web Token, and POST this information to the `user/login/oauth` API endpoint. The `user/login/oauth` API endpoint will POST to the URL provided and receive JSON Web Token. The `user/login/oauth` API endpoint will decode the token, validate that it is between the issued time and the expiration time, and validate that the public key set URL is allowed by the list of whitelisted URLs read from `/opt/traffic_ops/app/conf/cdn.conf`. It will then authorize the user from the database and return a mojolicious cookie as per the normal login workflow.

**Note:** Ensure that the user names in the Traffic Ops database match the value returned in the `sub` field in the response from the OAuth provider when setting up with the OAuth provider. The `sub` field is used to reference the roles in the Traffic Ops database in order to authorize the user.

**Note:** OAuth providers sometimes do not return the public key set URL but instead require a locally stored key. This functionality is not currently supported and will require further development.

**Note:** The POST from the API to the OAuth provider to exchange the code for a token expects the response to have the token in JSON format with `access_token` as the desired field (and can include other fields). It also supports a response with just the token itself as the body. Further development work will need to be done to allow other response forms or other response fields.



**Note:** Users must exist in both Traffic Ops as well as in the OAuth provider's system. The user's rights are defined by the *Role* assigned to the user.

To configure OAuth login:

- Set up authentication with a third party OAuth provider.
- Update `/opt/traffic_portal/public/traffic_portal_properties.json` and ensure the following properties are set up correctly:

Table 70: OAuth Configuration Property Definitions In `traffic_portal_properties.json`

Name	Type	Description
<code>enabled</code>	boolean	Allow OAuth SSO login
<code>oAuthUrl</code>	string	URL to your OAuth provider
<code>redirectUri-ParameterOverride</code>	string	Query parameter override if the oAuth provider requires a different key for the <code>redirect_uri</code> parameter, defaults to <code>redirect_uri</code>
<code>clientId</code>	string	Client id registered with OAuth provider, passed in with <code>client_id</code> parameter
<code>oAuth-CodeTo-tokenUrl</code>	string	URL to your OAuth provider's endpoint for exchanging the code (from <code>oAuthUrl</code> ) for a token

83: Example OAuth Configuration Properties In `traffic_portal_properties.json`

```
{
  "oAuth": {
    "_comment": "Opt-in OAuth properties for SSO login.
    ↪ See http://traffic-control-cdn.readthedocs.io/en/
    ↪ release-4.0.0/admin/quick_howto/oauth_login.html for
    ↪ more details. redirectUriParameterOverride defaults to
    ↪ redirect_uri if left blank.",
    "enabled": true,
    "oAuthUrl": "example.oauth.com",
    "redirectUriParameterOverride": "",
    "clientId": "",
    "oAuthCodeTokenUrl": "example.oauth.com/oauth/token
    ↪ "
  }
}
```

- Update `/opt/traffic_ops/app/conf/cdn.conf` property `traffic_ops_golang.whitelisted_oauth_urls` to contain all allowed domains for the JSON key set (Use `*` for wildcard):



Table 71: OAuth Configuration Property Definitions In cdn.conf

Name	Type	Description
whitelisted_oauth_urls	Array of strings	List of whitelisted URLs for the JSON public key set returned by OAuth provider. Can contain * wild-cards.
oauth_client_secret	string	Client secret registered with OAuth provider to verify client, passed in with <i>client_secret</i> parameter

84: Example OAuth Configuration Properties In cdn.conf

```
{
  "traffic_ops_golang": {
    "whitelisted_oauth_urls": [
      "oauth.example.com",
      ".*example.com"
    ],
    "oauth_client_secret": "secret"
  }
}
```

## Compare Profiles

In Traffic Portal *all* users can compare the *Parameters* of any 2 *Profiles* side-by-side, and users with a higher level *Role* (“operations” or “admin”) can easily add or remove parameters from each profile as necessary.

The ability to compare 2 profiles can be found under *Configure* → *Profiles* → *More* → *Compare Profiles*

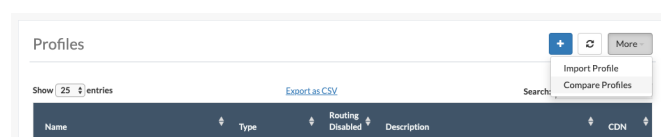


Fig. 40: The “Compare Profiles” menu item

Once you have selected the *Compare Profiles* menu item, you will be asked to choose 2 profiles to compare.

All parameters exclusively assigned to one profile but not the other will be displayed by default with their profile membership displayed side-by-side. In addition, by selecting the *Show Shared Params* link, the user can see a superset of all parameters (shared and not shared) across the 2 profiles. Both views provide users with higher level permissions (“operations” or “admin”) the ability to easily remove or add parameters for each profile and persist the final state of both profiles (or restore the original state and discard changes). As the user makes changes, a blue shadow is added to all modified checkboxes.

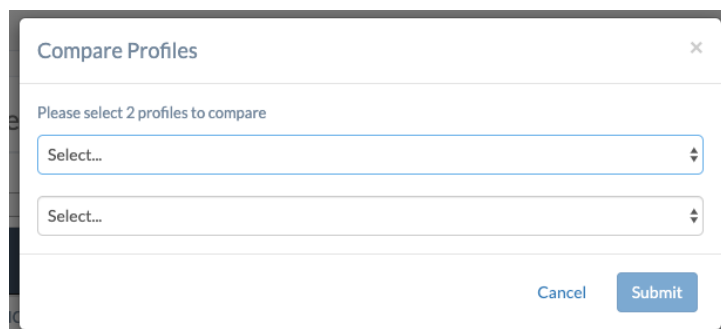


Fig. 41: The “Compare Profiles” dialog

Name	Config File	Value	PROFILE_1	PROFILE_2
edge.http.routing	CRCConfig.json	true	<input checked="" type="checkbox"/>	<input type="checkbox"/>
edge.http.limit	CRCConfig.json	12	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CONFIG.proxy.config.temp_dir	records.config	STRING /tmp	<input type="checkbox"/>	<input checked="" type="checkbox"/>
neustar.polling.interval	CRCConfig.json	86400000	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Fig. 42: The “Compare Profiles” table

## Configure Regional Geo-blocking (RGB)

**Note:** RGB (Regional Geographic-based Blocking) is only supported for HTTP *Delivery Services*.

1. Prepare an RGB configuration file. RGB uses a configuration file in JSON format to define regional geographic blocking rules for *Delivery Services*. The file needs to be put on an HTTP server accessible to Traffic Router.

### 85: Example Configuration File

```
{
  "deliveryServices":
  [
    {
      "deliveryServiceId": "hls-live",
      "urlRegex": ".*live4\\.m3u8",
      "geoLocation": {"includePostalCode": ["NØH",
↪ "L9V", "L9W"],
      "coordinateRange": [{"minLat": ↪
↪ -12, "maxLat": 13, "minLon" : 55, "maxLon": 56}, {
↪ "minLat" : -13, "maxLat": 14, "minLon" : 55, "maxLon": ↪
↪ 56}]},
      "redirectUrl": "http://third-party.com/blacked_
↪ out.html"
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "deliveryServiceId": "hls-live",
      "urlRegex": ".*live5\\.m3u8",
      "ipWhiteList": ["185.68.71.9/22", "142.232.0.79/
↪24"],
      "geoLocation": {"excludePostalCode": ["NOH",
↪"L9V"]},
      "redirectUrl": "/live5_low_bitrate.m3u8",
      "isSteeringDS": "false"
    },
    {
      "deliveryServiceId": "linear-steering",
      "urlRegex": ".*live3\\.m3u8",
      "ipWhiteList": ["185.68.71.9/22", "142.232.0.79/
↪24"],
      "geoLocation": {"excludePostalCode": ["NOH",
↪"L9V"]},
      "redirectUrl": "http://ip-slate.cdn.example.
↪com/slate.m3u8",
      "isSteeringDS": "true"
    }
  ]
}

```

**deliveryServiceId** Should be equal to the ID or `xml_id` field of the intended *Delivery Service* as configured in Traffic Portal

**urlRegex** A regular expression to be used to determine to what URLs the rule shall apply; a URL that matches it is subject to the rule

**geoLocation** An object that currently supports only the keys `includePostalCode`, `excludePostalCode` (mutually exclusive) and `coordinateRange`. When the `includePostalCode` key is used, only the clients whose FSA (Forward Sortation Areas)s - the first three postal characters of Canadian postal codes - are in the `includePostalCode` list are able to view the content at URLs matched by the `urlRegex`. When `excludePostalCode` is used, any client whose FSA is not in the `excludePostalCode` list will be allowed to view the content. The `coordinateRange` key is used to specify a list of latitude and longitude ranges. This is used in regional geo blocking, in case the client does not have a postal code associated with it.

**redirectUrl** The URL that will be returned to the blocked clients. Without a domain name in the URL, the URL will still be served in the same *Delivery Service*. Thus Traffic Router will redirect the client to a chosen *cache server* assigned to the *Delivery Service*. If the URL includes a domain name, Traffic Router simply redirects the client to the defined

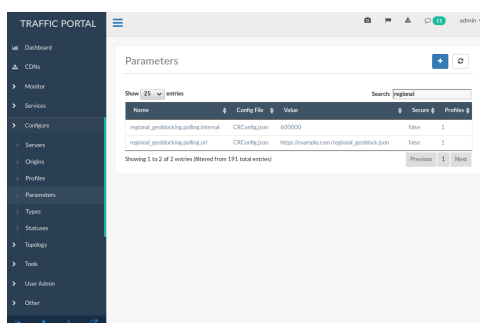
URL. In the latter case, the redirect URL must not match the `urlRegex` value, or an infinite loop of HTTP 302 Found responses will occur at the Traffic Router. *Steering-Type Delivery Services* must contain an FQDN as the re-redirect or Traffic Router will return a DENIED to the client. This is because steering services do not have caches associated to them, so a relative `redirectURL` can not be turned into a FQDN.

**ipWhiteList** An optional element that is an array of CIDR blocks indicating the IPv4 and/or IPv6 subnets that are allowed by the rule. If this list exists and the value is not empty, client IP will be matched against the CIDR list, bypassing the value of `geoLocation`. If there is no match in the white list, Traffic Router defers to the value of `geoLocation` to determine if content ought to be blocked.

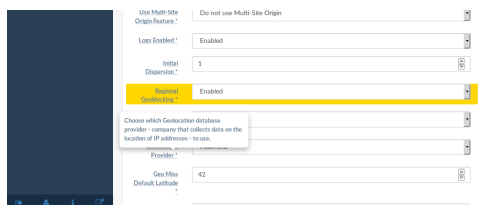
2. Add RGB *Parameters* in Traffic Portal to the *Delivery Service*'s Traffic Router(s)'s *Profile(s)*. The *Config File* value should be set to `CRConfig.json`, and the following two *Parameter Name/Value* pairs need to be specified:

**regional\_geoblock.polling.url** The URL of the RGB configuration file. Traffic Router will fetch the file from this URL using an HTTP GET request.

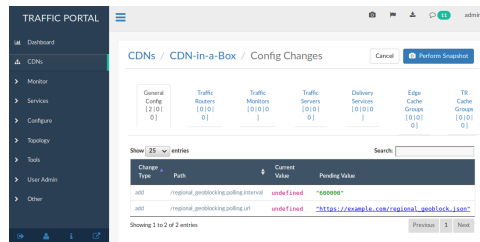
**regional\_geoblock.polling.interval** The interval on which Traffic Router polls the RGB configuration file.



3. Enable RGB for a *Delivery Service* using the *Delivery Services* view in *Traffic Portal* (don't forget to save changes!)



4. Go to *the Traffic Portal CDNs view*, click on *Diff CDN Config Snapshot*, and click *Perform Snapshot*.



## Traffic Router Access Log

See also:

*Troubleshooting and log files*

RGB extends the `rtype` field and adds a new field `rgb` in Traffic Router `access.log` to help to monitor this feature. A value of `RGALT` in the `rtype` field indicates that a request is redirected to an alternate URL by RGB; a value of `RGDENY` indicates that a request is denied by RGB because there is no matching rule in the RGB configuration file for this request. When RGB is enabled, the `rgb` field will be non-empty with following format:

```
{FSA}:{allowed/disallowed}:{include/exclude postal}:{fallback
config}:{allowed by whitelist}
```

**FSA** FSA part of the client's postal code, which is retrieved from a geographic location database. If this field is empty, a dash ("-") is filled in.

**allowed/disallowed** This flag shows if a request was allowed or disallowed by RGB (1 for yes, and 0 for no).

**include/exclude postal** This shows that when a rule in JSON is matched for a request, it's value is "I" if the rule matched because of an `includePostalCode` rule, "X" if the rule matched because of an `excludePostalCode` rule, or "-" if no rule matched.

**fallback config** When Traffic Router fails to parse an RGB configuration file as JSON, Traffic Router will handle requests with latest valid configuration that it had, but will set the `fallback config` flag to 1. If no fall-back occurred, then the flag is set to 0.

**allowed by whitelist** If a request is allowed by a `whitelist` field in the configuration, this flag is set to 1; for all other cases, it is 0.

### 86: Example

```
1446442214.685 qtype=HTTP chi=129.100.254.79 url="http://foo.geo2.cdn.
→com/live5.m3u8" cqhm=GET cqhv=HTTP/1.1 rtype=GEO rloc="-" rdtl=-
→rerr="-" rgb="N6G:1:X:0:0" pssc=302 ttms=3 rurl=http://cent6-44.geo2.
→cdn.com/live5.m3u8 rh="-"

1446442219.181 qtype=HTTP chi=184.68.71.9 url="http://foo.geo2.cdn.com/
→live5.m3u8" cqhm=GET cqhv=HTTP/1.1 rtype=RGALT rloc="-" rdtl=- rerr=
→"-" rgb="-:0:X:0:0" pssc=302 ttms=3 rurl=http://cent6-44.geo2.cdn.
→com/low_bitrate.m3u8 rh="-"
```

(continues on next page)

(continued from previous page)

```
1446445521.677 qtype=HTTP chi=24.114.29.79 url="http://foo.geo2.cdn.
→com/live51.m3u8" cqhm=GET cqhv=HTTP/1.1 rtype=RGDENY rloc="-" rdtl=-
→rerr="-" rgb="L4S:0:-:0:0" pssc=520 ttms=3 rurl="-" rh="-"
```

## Manage Server Capabilities

Server capabilities are designed to enable users with the Operations or Admin *Role* to control the flow of *Delivery Service* traffic through only the *cache servers* (*Edge* or *Mid*) that have the required capabilities. For example, *Delivery Services* designed to serve large binary files should only have requests routed to *cache servers* with sufficient disk cache. Currently, this can be controlled at the *Edge-tier* where system operators can explicitly assign only *Edge-tier caches* with sufficient disk cache to the *Delivery Service*. However, operators do not have control of *Mid-tier cache* assignments and cannot dictate which *Mid-tier caches* are qualified to serve these large binary files. This will cause a problem if a *Mid-tier cache* with insufficient disk cache is asked to serve the *Delivery Services*' large binary files.

A list of the server capabilities can be found under *Configure* → *Server Capabilities*. Users with a higher-level *Role* ("operations" or "admin") can create or delete server capabilities. Server capabilities can only be deleted if they are not currently being used by a *cache server* or required by a *Delivery Service*.

Name	
CACHE_MEMORY	
CACHE_DISK	

Fig. 43: Example Server Capabilities Listing

## Manage server capabilities assigned to a server

Users with the Operations or Admin *Role* can associate one or more server capabilities with a *cache servers* (*Edge* or *Mid*) by navigating to a server via *Configure* → *Servers* and using the context menu for the server table and selecting *Manage Capabilities* or by navigating to *Configure* → *Servers* → *Server* → *More* → *Manage Capabilities*.

Name	
CACHE_MEMORY	

Fig. 44: Example Server Capabilities for a Server Listing

## Manage delivery service required server capabilities

Users with the Operations or Admin *Role* can associate one or more required server capabilities with a *Delivery Service* by navigating to a *Delivery Service* via *Services* → *Delivery Services* and using the context menu for the *Delivery Services* table and selecting *Manage Required Server Capabilities* or by navigating to *Services* → *Delivery Services* → *Delivery Service* → *More* → *Manage Required Server Capabilities*.

Adding a required server capability to a *Delivery Service* will ensure two things:

1. Only *Edge-tier caches* with the required capability can be assigned to the *Delivery Service*
2. Only *Mid-tier caches* with the required capability will handle requests of the *Delivery Service* (if applicable)

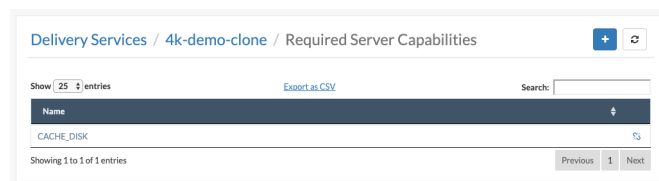


Fig. 45: Example Required Server Capabilities for a Delivery Service Listing

## Configuring Static DNS Entries

Static DNS records (historically “entries”) can be configured within the DNS subdomain of a given *Delivery Service*. In a typical scenario, the *Delivery Service* will have DNS records automatically generated based on its “xml\_id” and “routing name”, and the name and subdomain of the CDN to which it belongs. For example, in the *CDN in a Box* default environment, the “demo1” *Delivery Service* has an automatically generated DNS record for `video.demo1.mycdn.ciab.test`. Configuring a static DNS record allows for further extension of this, for example, one could create an A record that enforces lookups of the name `foo.demo1.mycdn.ciab.test` resolve to the IPv4 address `192.0.2.1`.

**Note:** It’s only possible to create static DNS records within a *Delivery Service*’s subdomain. That is, one could not create an A record for `foo.bar.mycdn.ciab.test` on the *CDN in a Box Delivery Service* “demo1”, since “demo1”’s subdomain is `demo1.mycdn.ciab.test`.

### See also:

This guide covers how to set up static DNS records using Traffic Portal. It’s also possible to do so directly using the *Traffic Ops API* endpoint *staticdnsentries*.

## Example

To set up the aforementioned rule, follow these steps.

1. In Traffic Portal, expand the *Services* sidebar menu and select *Delivery Services*.
2. From the now-displayed table of *Delivery Services*, select the desired one for static DNS record configuration.
3. From the *More* drop-down menu, select *Static DNS Entries*. The displayed table will probably be empty.

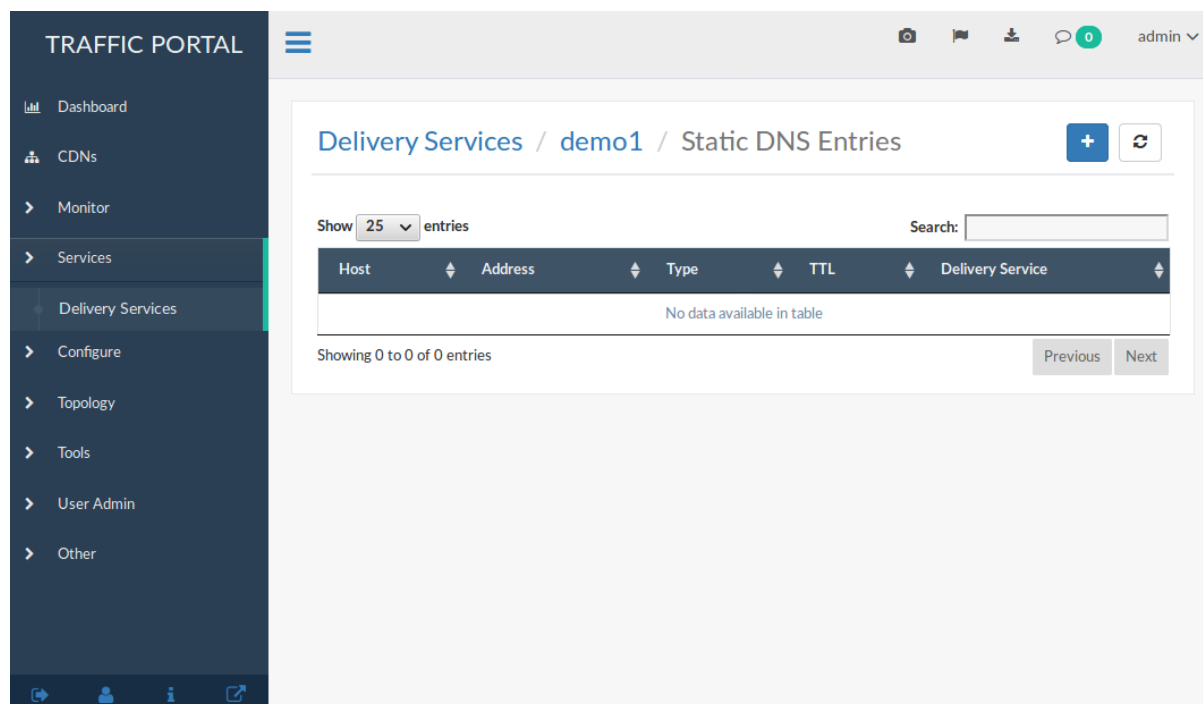


Fig. 46: The Static DNS Entries Table Page

4. Click on the blue + button to add a new static DNS Entry
5. Fill in all of the fields.

**Host** This is the lowest-level DNS label that will be used in the DNS record. In the *CDN in a Box* scenario, for example, entering `foo` here will result in a full DNS name of `foo.demo1.mycdn.ciab.test`.

**Type** Indicates the type of DNS record that will be created. The available types are

- A
- AAAA
- CNAME
- TXT



**TTL** The TTL of the DNS record, after which clients will be expected to re-request name resolution.

**Address** The meaning of this field depends on the value of the “Type” field.

- If the “Type” is **A**, this must be a valid IPv4 address
- If the “Type” is **AAAA**, this must be a valid IPv6 address
- If the “Type” is **CNAME**, this must be a valid DNS name - **not** an IP address at all
- If the “Type” is **TXT**, no restrictions are placed on the content whatsoever

Fig. 47: An Example Static DNS Entry Form

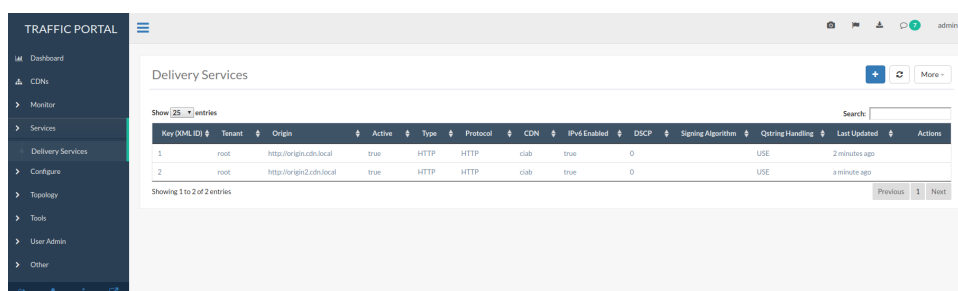
6. Click on the green *Create* button to finalize the changes.
7. At this point, although the static DNS record has been created, it will have no effect until a new CDN *Snapshot* is taken. Once that is done (and enough time has passed for Traffic Router to poll for the changes), the new DNS record should be usable through the CDN’s designated Traffic Router.

### 87: Example DNS Query to Test a New Static DNS Entry within *CDN in a Box*

```
$ docker exec cdnabox_enroller_1 dig +noall +answer foo.
demo1.mycdn.ciab.test
foo.demo1.mycdn.ciab.test. 42    IN  A    192.0.2.1
```

## Configure Delivery Service Steering

1. Create two target *Delivery Services* in Traffic Portal. They must both be HTTP *Delivery Services* that are part of the same CDN.

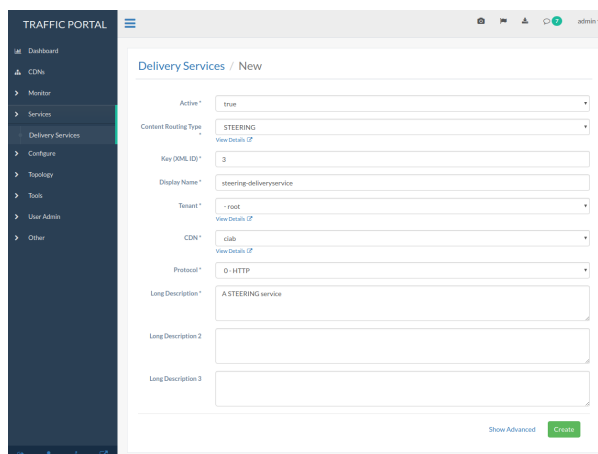


The screenshot shows the 'Delivery Services' page in the Traffic Portal. A table lists two services:

Key (DML ID)	Tenant	Origin	Active	Type	Protocol	CDN	IPv6 Enabled	DISC	Signing Algorithm	Query Handling	Last Updated	Actions
1	root	http://origin1.cdn.local	true	HTTP	HTTP	ciab	true	0		USE	2 minutes ago	
2	root	http://origin2.cdn.local	true	HTTP	HTTP	ciab	true	0		USE	a minute ago	

Fig. 48: Target *Delivery Services*

2. Create a *Delivery Service* with Type STEERING or CLIENT\_STEERING in Traffic Ops.



The screenshot shows the 'New Delivery Service' form in the Traffic Portal. The form is configured as follows:

- Active: ☒
- Content Routing Type: STEERING
- Key (DML ID): 3
- Display Name: steering-delivery-service
- Tenant: root
- CDN: ciab
- Protocol: HTTP
- Long Description: A STEERING service
- Long Description 2: (empty)
- Long Description 3: (empty)
- Buttons: Show Advanced, Create

Fig. 49: Creating a STEERING *Delivery Service*

3. Click *More* → *View Targets* and then use the blue + button to assign targets.
4. If desired, a ‘steering’ *Role* user can create filters for the target *Delivery Services* using *steering/{{ID}}/targets*

Delivery Service	Target	Type	Value
3	2	STEERING_WEIGHT	1
3	1	STEERING_WEIGHT	1

Fig. 50: STEERING Targets

---

**Note:** This is only available via the *Traffic Ops API*; no functionality for manipulating steering targets is offered by Traffic Portal. This feature has been requested and is tracked by [GitHub Issue #2811](#)

---

- Any requests to Traffic Router for the steering *Delivery Service* should now be routed to target *Delivery Services* based on configured weight or order.

---

**Note:** This example assumes that the Traffic Ops instance is running at `to.cdn.local` and the administrative username and password are `admin` and `twelve`, respectively. This is *not* recommended in production, but merely meant to replicate the default *CDN in a Box* environment!

---

### 3.1.11 CDNi Administration

CDNi specifications define the standards for interoperability within the CDN and open caching ecosystems set forth by the IETF (Internet Engineering Task Force). This integration utilizes the APIs (Application Programming Interfaces) defined by the SVA (Streaming Video Alliance).

**See also:**

For complete details on CDNi and the related API specifications see [RFC 8006](#), [RFC 8007](#), [RFC 8008](#), and the SVA documents titled *Footprint and Capabilities Interface: Open Caching API*, *Open Caching API Implementation Guidelines*, *Configuration Interface: Part 1 Specification - Overview & Architecture*, *Configuration Interface: Part 2 Specification – CDNi Metadata Model Extensions*, and *Configuration Interface: Part 3 Specification – Publishing Layer APIs*.

In short, these documents describe the CDNi metadata interface that enables interconnected CDNs (Content Delivery Networks) to exchange content distribution metadata to enable content acquisition and delivery. These define the interfaces through which a uCDN (Upstream Content Delivery Network) and a dCDN (Downstream Content Delivery Network) can communicate configuration and capacity information.

For our use case, it is assumed that ATC is the dCDN.

---

**Note:** This is currently under construction and will be for a while. This document

---

will be updated as new features are supported.

---

### /OC/FCI/advertisement

**See also:**

*OC/FCI/advertisement*

The advertisement response is unique for the uCDN and contains the complete footprint and capabilities information structure the dCDN wants to expose. This endpoint will return an array of generic FCI (Footprint and Capabilities Advertisement Interface) base objects, including type, value, and footprint for each. Currently supported base object types are *FCI.Capacity* and *FCI.Telemetry* but these will be expanded in the future.

### /OC/CI/configuration

**See also:**

*OC/CI/configuration*

An endpoint that is used to push (PUT), fetch (GET), or delete (DELETE) the entire metadata set for a given uCDN from a JWT. This puts the requested change into a queue to be reviewed later and returns an endpoint to view the asynchronous status updates.

---

**Note:** This is under construction. Currently only PUT is supported and in a very limited sense.

---

### /OC/CI/configuration/{{host}}

**See also:**

*OC/CI/configuration/{{host}}*

An endpoint that is used to push (PUT), fetch (GET), or delete (DELETE) the metadata set that is attached to host name for a given uCDN from a JWT. This puts the requested change into a queue to be reviewed later and returns an endpoint to view the asynchronous status updates.

---

**Note:** This is under construction. Currently only PUT is supported and in a very limited sense.

---

**/OC/CI/configuration/request/{id}/{approved}**

**See also:**

*OC/CI/configuration/request/{id}/{approved}*

This endpoint allows a user to approve or deny a queued update request from the previous endpoints. A denial will result in the removal from the queue and a **FAILED** status update. An approval will result in the changes being made to the configuration and a **SUCCEEDED** status update.

---

**Note:** This is under construction and only supports very limited metadata field and limited configuration updates.

---



## DEVELOPER'S GUIDE

A guide to the various internal and external APIs, and an introduction for the Traffic Control developer.

### 4.1 Developer's Guide

Use this guide to start developing applications that consume the Traffic Control APIs, to create extensions to Traffic Ops, or work on Traffic Control itself.

#### 4.1.1 API Guidelines

This section lays out guidelines to be considered by API endpoint authors. What follows are not *strictly* hard-and-fast rules, but there should be a very convincing argument accompanying endpoints that do not follow them.

Legacy API versions don't, in general, adhere to these principles. This section is not meant as a proposal for altering them to be compliant, but merely a set of guidelines for future work.

#### Response Bodies

All valid API responses will be in the form of some serialized object. The main data that represents the result of the client's request **MUST** appear in the `response` property of that object. If a warning, error message, success message, or informational message is to be issued by the server, then they **MUST** appear in the `alerts` property of the response. Some endpoints may return ancillary statistics such as the total number of objects when pagination occurs, which should be placed in the `summary` property of the response.

## Response

The `response` property of a serialized response object **MUST** only contain object representations as requested by the client. In particular, it **MUST NOT** contain admonitions, success messages, informative messages, or statistic summaries beyond the scope requested by the client.

Equally unacceptable API responses are shown in *Success Message as Response Object* and *Illegal Top-Level Property*.

### 88: Success Message as Response Object

```
{
  "response": "Thing was successfully created."
}
```

### 89: Illegal Top-Level Property

```
{
  "response": {"foo": "bar"},
  "someOtherField": {"someOtherObject"}
}
```

When requests are serviced by Traffic Ops that pass data asking that the returned object list be filtered, the `response` property **MUST** be a filtered array of those objects (assuming the request may be successfully serviced). This is true even if filtering is being done according to a uniquely identifying property - e.g. a numeric ID. The `response` field of an object returned in response to a request to create, update, or delete one or more resources may be either a single object representation or an array thereof according to the number of objects created, updated, or deleted. However, if a handler is *capable* of creating, updating, or deleting more than one object at a time, the `response` field **SHOULD** consistently be represented as an array - even if its length would only be 1.

The proper value of an empty collection is an empty collection. If a Foo can have zero or more Bars, then the representation of a Foo with no Bars **MUST** be an empty array/list/set, and in particular **MUST NOT** be either missing from the representation or represented as the “Null” value of the representation format. That is, if Fools have no other property than their Bars, then a Foo with no Bars may be represented in JSON encoding as `{"bars": []}`, but not as `{"bars": null}` or `{}`. Similarly, an empty string field is properly represented as an empty string - e.g. `{"bar": ""}` not `{"bar": null}` or `{}` - and the “zero-value” of numbers is zero itself - e.g. `{"bar": 0}` not `{"bar": null}` or `{}`. Note that “null” values *are allowed* when appropriate, but “null” values represent the *absence* of a value rather than the “zero-value” of a property. If a property is *missing* from an object representation it indicates the absence of that property, and because of that there must be a *very convincing* argument if and when that is the case.

As a special case, endpoints that report statistics including minimums, maximums and arithmetic means of data sets **MUST** use the property names `min`, `max`, and `mean`, respectively, to express those concepts. These **SHOULD** be properties of `response` directly whenever that makes sense.



## Alerts

Alerts should be presented as an array containing objects which each conform to the object definition laid out by [github.com/apache/trafficcontrol/tree/master/lib/go-tc#Alert](https://github.com/apache/trafficcontrol/tree/master/lib/go-tc#Alert). The allowable levels of an Alert are:

- **error** - This level **MUST** be used to indicate conditions that caused a request to fail. Because of this, this level **MUST NOT** appear in the `alerts` array of responses with any HTTP response code less than 400 (except when used for asynchronous tasks as discussed in [202 Accepted](#)). Details of server workings and/or failing components **MUST NOT** be exposed in this message, which should otherwise be as descriptive as possible.
- **info** - This level **SHOULD** be used to convey supplementary information to a user that is not directly the result of their request. This **SHOULD NOT** carry information indicating whether or not the request succeeded and why/why not, as that is best left to the **error** and **success** levels.
- **success** - This level **MUST** be used to convey success messages to the client. In general, it is expected that the message will be directly displayed to the user by the client, and therefore can be used to add human-friendly details about a request beyond the response payload. This level **MUST NOT** appear in the `alerts` array of responses with an HTTP response code that is not between 200 and 399 (inclusive).
- **warning** - This level is used to warn clients of potentially dangerous conditions when said conditions have not caused a request to fail. The best example of this is deprecation warnings, which should appear on all API routes that have been deprecated.

## Summary

The `summary` object is used to provide summary statistics about object collections. In general the use of `summary` is left to be defined by API endpoints (subject to some restrictions). However, its use is not appropriate in cases where the user is specifically requesting summary statistics, but should rather be used to provide supporting information - pre-calculated - about a set of objects or data that the client *has* requested.

Endpoints **MUST** use the following, reserved properties of `summary` for their described purposes (when use of `summary` is appropriate) rather than defining new `summary` or `response` properties to suit the same purpose:

- **count** - Count contains an unsigned integer that defines the total number of results that could possibly be returned given the non-pagination query parameters supplied by the client.

### HTTP Request Methods

**RFC 7231#section-4** defines the semantics of HTTP/1.1 request methods. Authors should conform to that set of standards whenever possible, but for convenience the methods recognized by Traffic Ops and their meanings in that context are herein defined.

#### GET

HTTP GET requests are issued by clients who want some data in response. In the context of Traffic Ops, this generally means a serialized representation of some object. GET requests **MUST NOT** alter the state of the server. An example of an API endpoint created in API version 1 that violates this restriction is `cdns/name/name/dnsseckey/delete`.

This is the standard method to be used by all read-only operations, and as such handlers for this method should generally be accessible to users with the “read-only” *Role*.

All endpoints dealing with the manipulation or fetching representations of “Traffic Control Objects” **MUST** support this method.

#### POST

POST requests ask the server to process some provided data. Most commonly, in Traffic Ops, this means creating an object based on the serialization of said object contained in the request body, but it can also be used virtually whenever no other method is appropriate. When an object *is* created, the response body **MUST** contain a representation of the newly created object. POST requests do not need to be *idempotent*, unlike PUT requests.

#### PUT

PUT is used to replace existing data with new data that is provided in the request body. **RFC 2616#section-9.1.2** lists PUT as an “idempotent” request method, which means that subsequent identical requests should ensure the same state is maintained on the server. What this means is that a client that PUTs an object representation to Traffic Ops expects that if they then GET a representation of that object, do the same PUT again and GET another representation, the two retrieved representations should be identical. Effectively, the `lastUpdated` field that is common to objects in the *Traffic Ops API* violates this, but the other properties of objects - which can actually be defined - generally obey this restriction. In general, fulfilling this restriction means that handlers will need to require the entirety of an object be defined in the request body.

When an object is replaced, the response body **MUST** contain a representation of the object after replacement. While **RFC 2616** states that servers **MAY** create objects for the passed representations if they do not already exist, *Traffic Ops API* endpoint authors **MUST** instead use POST handlers for object creation.

All endpoints that support the PUT request method **MUST** also support the *If-Unmodified-Since* HTTP header.

## **PATCH**

At the time of this writing, no *Traffic Ops API* endpoints handle the PATCH request method. PATCH requests that the server's stored data be mutated in some way using data provided in the request body. Unlike PUT, PATCH is not *idempotent*, which essentially means that it can be used to change only part of a stored object. When an object is modified, the response body **MUST** contain a representation of the object after modification, and that representation **SHOULD** fully describe the modified object, even the parts that were not modified.

Handlers that implement PATCH in the *Traffic Ops API* **MUST** use conditional requests to ensure that race conditions are not a problem, specifically they **MUST** support using *ETag* and *If-Match*, and **SHOULD** also support *If-Unmodified-Since*.

Clients **SHOULD** use PATCH requests rather than PUT requests for modifying existing resources whenever it is supported.

## **DELETE**

DELETE destroys an object stored on the server. Typically the request will contain identifying information for the object(s) to be destroyed either in the request URI or in the request's body. *Traffic Ops API* endpoint authors **MUST** use this request method whenever an object identified by the request URI is being destroyed. When such deletion successfully occurs, the response body **MUST** contain a representation of the destroyed object.

## **HTTP Response Codes**

Proper use of HTTP response codes can significantly improve user interfaces built on top of the API. What follows is a (non-exhaustive) set of response codes and their appropriate use in the context of Traffic Ops. For more complete information, refer to [the Mozilla Developer Network's HTTP Response Code list](#).

### 200 OK

This indicates the request succeeded, with no additional semantics. This **MUST** be the exact response status code of successful GET requests. This is also the default “success” response code for any other request.

### 201 Created

This indicates that a resource was successfully created on the server. This **MUST** be the response status code of POST requests that create a new object or objects on the server, and in that case the response **SHOULD** also include a *Location* header that provides a URI where a representation of the newly created object may be requested.

### 202 Accepted

**202 Accepted** **MUST** be used when the server is performing some task asynchronously (e.g. refreshing DNSSEC keys) but the status of that task cannot be ascertained at the current time. Ideally in this case, when the task completes - either successfully or by failing - the Traffic Ops changelog will be updated to indicate that status, along with information to uniquely identify the task (e.g. username and date/time when the task started).

Endpoints that create asynchronous jobs **SHOULD** provide a URI to which the client may send GET requests to obtain a representation of the job’s current state in the *Location* HTTP header. They **MAY** also provide an *info*-level Alert that provides the same or similar information in a more human-friendly manner.

The responses to such GET requests are subject to the same restrictions as any other API endpoint, but have the added restriction that the response objects sent **MUST** have the *status* property, which is a string limited to one of the following values and having the associated semantics:

**PENDING** This means the job has been started but is not yet completed.

**SUCCEEDED** This means that the asynchronous job has completed and encountered no errors.

**FAILED** The task encountered errors and was unable to continue, and thus has been terminated.

Note that the response code of the response carrying this information **MUST NOT** depend on the value of *status*. In particular, a response that successfully reports the status of a **FAILED** asynchronous task is still successfully servicing a client’s GET request, and therefore **MUST** have the **200 OK** response status code. However, a response encoding a **FAILED** status **MUST** be accompanied by one or more *error*-level Alerts that explain (to the greatest degree of detail allowable securely) why the job failed.

These responses **MUST** also include the *startTime* and *endTime* properties which indicate, respectively, the time at which the asynchronous job started and the time at which it concluded.

A job that has not started **MUST** have a Null-valued `startTime` and likewise a job that has yet to conclude **MUST** have a Null-value `endTime`.

## **400 Bad Request**

In general this is used when there's something syntactically wrong with the client's request. For example, Traffic Ops **MUST** respond with this code when the request body was improperly encoded. In most cases, this is also the proper response code when the client submits data that is not semantically correct. For example, dates/times represented as timestamp strings in an unsupported format should trigger this response code.

This is also the default "client failure" response code for any other request.

The response body **MUST** include an entry in the `alerts` array that describes to the client what was wrong with the request.

## **401 Unauthorized**

This **MUST** be the response code when a client without valid authorization information in the HTTP headers requests a resource which cannot be accessed without first authorizing. Which should be everything except `/ping` and endpoints that provide authorization.

## **403 Forbidden**

This **MUST** be used whenever the client is logged-in, but still does not have access to the resource they are requesting. It **MUST** also be used when they have some access to the resource, but not with the specific request method they used. This can pertain to restricted access on the basis of *Role*, User Permissions, as well as *Tenancy*.

The response body **MUST NOT** disclose any information regarding why the user was denied access.

## **404 Not Found**

This **MUST** be the returned status code when the client requests a path that does not exist on the server. Note that a *path* does not include a *query string*; in the URL `http://example.test/some/path?query#frag` the *path* consists of only `/some/path`.

### 409 Conflict

This SHOULD be used when the request cannot be completed because the current state of the server is fundamentally incompatible with the request. For example, creating a new user with an email that is already in use should result in this response.

Additionally, this MAY be used instead of *404 Not Found* when the client is requesting a link between an object identified by the request URI and some other object (e.g. when assigning a *cache server* to a *Delivery Service*) when the other object does not exist. If the request URI identifies an object that does not exist, the response MUST use *404 Not Found* instead.

This is also the proper response status code when the conditions of a request cannot be met, e.g. when a client submits a PATCH request for a resource with an *If-Match* header that does not match the stored object's *ETag*.

The response body MUST indicate what the conflict is that prevented the request from being fulfilled via one or more **error-level** alerts.

### 500 Internal Server Error

When the Traffic Ops server encounters some error - through no fault of the client or their request - that renders it incapable of servicing the client's request, it MUST return this status code if no other code is more appropriate. The response body in this case SHOULD indicate that an error occurred, but MUST NOT divulge details about what data was being processed, what (if any) other components are not functioning properly, or what process failed. Generally it is advisable that the resultant alerts array entry just say "Internal Server Error" and nothing else.

### 501 Not Implemented

This is the response code used when the client requests an API version not implemented by the server. It SHOULD NOT be used in any other case.

### 502 Bad Gateway

This code indicates that some other service on which the endpoint's processes depend has given back improper data or an error response. It MAY be used (with caution) by plugin developers, but SHOULD NOT be used by authors of proper API endpoints, as that divulges information about failing connected systems and potentially gives an attacker information about Traffic Control's weak points. API endpoint authors should instead use *500 Internal Server Error*.

## 504 Gateway Timeout

This code indicates that a connection timeout occurred when attempting to contact some other service on which the endpoint's processes depend. It MAY be used (with caution) by plugin developers, but SHOULD NOT be used by authors of proper API endpoints, as that divulges information about failing connected systems and potentially gives an attacker information about Traffic Control's weak points. API endpoint authors should instead use *500 Internal Server Error*.

## Documentation

All endpoints MUST be properly documented. For guidelines for writing API documentation, refer to *Documenting API Routes*.

## Passing Request Data

Request data may be passed in the request body or as a *application/x-www-form-urlencoded*-encoded query string in the request URI, or as a part of the request path. Request data MUST NOT be passed through a portion of the request path unless it uniquely identifies a resource with which the client may interact. For example, */foos/ID* is an acceptable path for dealing with the particular "Foo" object that has some identifier ID, but *logs/Number of Days/days* is unacceptable because reasonable default behavior can be provided if no number of days is given in the query string parameters, and that doesn't help uniquely identify a resource. Request path parameters should use double "curly-braces" (*{* and *}*) to call out variable components of the request path in documentation and references. Request path parameters MUST NOT be used for data that is optional to the request (somewhat obviously). Note that all endpoints dealing with the manipulation of "Traffic Control Objects" MUST support the GET HTTP request method.

When accepting data in the request body of requests, the endpoint MUST properly document the object representations (properties and their types) it accepts and MUST reject semantically invalid data with a *400 Bad Request* response code. For example, if an endpoint specifies it accepts a representation of a Foo object, assuming Foo objects possess only the Bar property which is an arbitrary string, then the endpoint MUST accept *{"bar": "testquest"}* as semantically valid (The data may be rejected for other reasons, e.g. if a Foo with such a Bar property already exists and Bars must be unique among all Foos) and MUST reject *{"bar": "testquest", "someOtherProperty": 10}* as semantically invalid. This is in contrast to the API's behavior at the time of this writing, which silently ignores unrecognized properties of request body objects.

The decision to pass data in the request body or query string is mainly up to the author, but some helpful tips:

- GET and DELETE requests do not typically provide request bodies.
- Query parameters should nearly always be optional. If data is required by an endpoint, consider requiring it in the request body. If the data identifies a resource, it ought to be a path parameter.



- Request body data often represents objects that are being created or updated. If an object is being created or updated, it ought to be defined in the request body, and if any additional data is (possibly optionally) required then it ought to be passed in the query string to separate it from the object definition.
- The following query parameters are reserved for special use by Traffic Ops endpoint handlers, and may not be used for any purpose other than their prescribed functions.
  - `limit`
  - `newerThan`
  - `offset`
  - `olderThan`
  - `orderBy`
  - `page`
  - `sortOrder`

### Duplicate Endpoints

No two endpoints should serve the same purpose. While it's fine to overlap a bit, an endpoint like `/foo_bars` should not exist solely to edit the `Bars` property of `Foo` objects (which can ostensibly be edited just fine on the object itself), for example. Ideally, there should be exactly one way to accomplish something through the API.

A caveat, though, is object relationships. For example, a *Delivery Service* has zero or more *cache servers* assigned to it, and in turn *cache servers* may be assigned to zero or more *Delivery Services* (a “has-and-belongs-to-many” relationship). Thus it is permissible to be able to edit the *Delivery Services* property of a *cache server* using the `/cache_servers` API endpoint as well as to be able to edit the *cache servers* property of a *Delivery Service* using the `/delivery_services` API endpoint - though they arguably provide equivalent functionality in that way (although at the time of this writing the former endpoint doesn't exist and the latter doesn't offer that functionality - this is just an example).

### Date/Time Format

Dates MUST be represented in either **RFC 3339** (with or without nanosecond precision) or as integers indicating the number of nanoseconds past the Unix epoch at which the date/time occurs. In either case, Dates included in responses from Traffic Ops MUST be in UTC. Wherever date/times are accepted as input, *Traffic Ops API* endpoints MUST accept either format and SHOULD NOT accept anything else.

Traffic Ops endpoints MUST return dates and times in **RFC 3339** format with nanosecond precision. Endpoints MAY provide ways for the client to specify alternate representations, but these SHOULD be restricted to only Unix epoch timestamps in nanoseconds.



## Age Filtering

Whenever object age is a property of that object (which is quite often in the form of `lastUpdated`), Traffic Ops endpoint handlers that respond to requests for object representations (i.e. GET requests) **SHOULD** support filtering by age. If age filtering is implemented, it **MUST** be made available using the query parameters in the table below.

Table 72: Age Filtering Query Parameters

Parameter	Meaning
<code>newerThan</code>	Timestamp to be used as the lower limit on an object's age. Objects older than this <b>MUST NOT</b> appear in the response body. That is, the response will be the set of all objects in the collection with a modification date that is greater than <i>or equal to</i> this value.
<code>olderThan</code>	Timestamp to be used as the upper limit an object's age. Objects newer than this <b>MUST NOT</b> appear in the response body. That is, the response will be the set of all objects in the collection with a modification date that is less than <i>or equal to</i> this value.

The format of these timestamps - in accordance with the [Date/Time Format](#) section - **MUST** be accepted as Unix epoch timestamps in nanoseconds, **and** in the form of [RFC 3339](#) date/time strings.

Endpoints **MAY** return errors when a client request gives these parameters improper or invalid values, but **MUST** at least provide a warning. When ambiguity or errors in age filtering controls render age filtering impossible, the handler **MUST NOT** perform age filtering.

## Tenancy

When a client requests access to a set of stored objects that are “tenantable” inevitably some of them will be inaccessible to the user on the basis of their [Tenant](#). Traffic Ops endpoint handlers that respond to requests for such object representations (i.e. GET requests) **MUST** filter their results implicitly according to the requesting Tenant's access. Any request that would modify, create, or destroy an object to which the requesting [Tenant](#) does not have access **MUST NOT** be fulfilled by the server (obviously) and in that case the response status code **MUST** be [403 Forbidden](#). Furthermore, if a request for a representation of a [Tenant](#)-inaccessible object is made *explicitly for said object* (e.g. GET `/foos/ID` rather than GET `/foos?id=ID`) the response status code **MUST** be [403 Forbidden](#).

### Naming Conventions

The names of properties of objects as they appear in said objects' serializations ought to conform to “camelCase” naming. Initialisms, abbreviations, and acronyms that appear in property names should be capitalized unless they are at the very beginning of the name. For example, `myIPAddress` and `someProperty` are both well-formed property names, while `IPAddress`, `someproperty` and `SomeProperty` are not.

Query string parameters **MUST** also follow “camelCase” naming.

API endpoints themselves should have a name that conveys their purpose. For example, `/cdns` is an endpoint that deals with manipulating, creating, destroying, or retrieving representations of CDNs. Request paths **MUST** use “snake\_case” to separate words whenever necessary, and **MUST** never include the action being performed by the handler; instead that is decided by the request *method*. For example, `/myObject/delete` is a poor request path name for both of those reasons. Furthermore, when an endpoint deals with an object type of which there are typically multiple, the request path should be plural, e.g. `/cdns` is better than `/cdn`.

API endpoints **MAY** support trailing slashes (/) in the request path, but **MUST NOT** include suffixes that indicate a particular encoding (“file extensions”); that's what the *Content-Type* header is for. For example, in API version 1.x, `/foos` and `/foos.json` are both equally valid ways to access the `/foos` endpoint handlers - this is no longer allowed!

### Relationships as Objects

Relationships **SHOULD NOT** be represented through the API as objects in their own right. For example, instead of an endpoint like `/delivery_service_servers` used to manipulate assignments of *cache servers* to *Delivery Services*, a *Delivery Service* itself should have *Servers* as a property. Thus assignments are manipulated by manipulating that property. So the only endpoints necessary for fully defining and dealing with such relationships are `/delivery_services` and `/servers`.

### Change Logging

All manipulations of objects (i.e. any operation that is not merely “reading” data) **MUST** add a Change Log entry indicating what was changed.

## 4.1.2 Environment Variables

Various ATC components and tools use specific environment variables to modify their behavior - in some cases they're even mandatory. What follows is a (hopefully) exhaustive list of the environment variables used in multiple places throughout the project - variables used by only a single tool or component can be found in the usage/development documentation for that tool or component. Developers are encouraged to expand the usage of variables mentioned here rather than create new ones to serve the same purpose in a different tool.

---

**Tip:** As environment variables can be set by any parent process, possibly unknown to the user or merely forgotten, they should generally have the lowest precedence of setting any given configuration (when multiple exist, e.g. configuration file parameters or command-line arguments) and should not be required if possible. It can be confusing to a user if `some-tool --to-url="http://to.infra.ciab.test"` actually uses some Traffic Ops URL other than the one they can see and specified explicitly. The suggested precedence order is:

1. Command-line arguments
  2. Configuration file entries/parameters/lines
  3. Environment variables
- 

## MOJO\_MODE

This is a legacy environment variable only used for compatibility's sake - new tools should not need to use it for anything, in general. It refers to the environment or "mode" of the Traffic Ops server from back in the old Perl days. Effectively, this chooses the set of configuration files it will consult. The default value is "development", and the possible values are:

- development
- integration
- production
- test

**admin** sets this to the value of the environment as specified by `admin --env` (Default: development)

## TO\_PASSWORD

This is used - typically in concert with `TO_PASSWORD` and `TO_USER` - to provide a password for a user as whom to authenticate with some Traffic Ops instance. This generally should not be validated, to avoid having to update validation to match the *Traffic Ops API*'s own validation - because in general this will/should end up in the payload of an authentication request to `user/login`.

**Caution:** For security reasons, the contents of this environment variable should not be stored anywhere for any length of time that isn't strictly necessary.

The following list details which components/tools use this variable, how, and what, if any, restrictions they place upon its content.

- `infrastructure/cdn-in-a-box/traffic_ops/to-access.sh` expects this password to authenticate the administrative-level user given by `TO_USER`.
- `atstccfg` uses this variable to authenticate the user used for fetching configuration information.

- `toaccess` uses this variable to authenticate with the Traffic Ops instance before sending requests.

### TO\_URL

This is used - typically in concert with `TO_USER` and `TO_PASSWORD` - to identify a Traffic Ops instance for some purpose. In general, this should be able to handle the various ways to identify a network location, and need not be restricted to strictly a URL - for example, an FQDN without a scheme should be acceptable, with or without port. When ports are not given, but scheme is, the port to use should be trivially deducible from a given supported scheme - 80 for `http://` and 443 for `https://`. When the scheme is not given but the port is, the reverse assumptions should be made. When neither are given, or the port is non-standard, then `https://` and port 443 (if applicable) should be assumed.

There are cases in the repository today where that is not true. The following list details the components/tools that use this variable and their restrictions and expected formats for it, if they differ from the description above.

- `infrastructure/cdn-in-a-box/traffic_ops/to-access.sh` uses this variable to connect to the *CDN in a Box* Traffic Ops instance. It is passed directly to `curl(1)` after path portions are appended, and so is subject to the restrictions and assumptions thereof.
- The Traffic Ops *Go Tests* for integration with the Go client use this to define the URL at which the Traffic Ops instance is running. It will override configuration file settings that specify the instance location.
- `atstccfg` uses this variable to specify the Traffic Ops instance used to fetch configuration information.
- `toaccess` uses this variable to identify the Traffic Ops instance to which requests will be sent.

### TO\_USER

This is used - typically in concert with `TO_PASSWORD` and `TO_USER` - to provide a user name as whom to authenticate with some Traffic Ops instance. This generally should not be validated, to avoid having to update validation to match the *Traffic Ops API*'s own validation - because in general this will/should end up in the payload of an authentication request to `user/login`.

The following list details which components/tools use this variable, how, and what, if any, restrictions they place upon its content.

- `infrastructure/cdn-in-a-box/traffic_ops/to-access.sh` expects this to be the name of an administrative-level user.
- `atstccfg` uses this variable to name the user as whom to authenticate for fetching configuration information.
- `toaccess` uses this variable to authenticate with the Traffic Ops instance before sending requests.

### 4.1.3 Building Traffic Control

The build steps for Traffic Control components are all pretty much the same, despite that they are written in a variety of different languages and frameworks. This is accomplished by using Docker.

---

**Note:** Currently, both listed methods of building Traffic Control components will produce \*.rpm files, meaning that the support of these components is limited to RedHat-based distributions - and none of them are currently tested (or guaranteed to work) outside of Rocky Linux 8 and CentOS 7, specifically.

---

#### Downloading Traffic Control

If any local work on Traffic Monitor, Traffic Router Golang, Grove or Traffic Ops is to be done, it is **highly** recommended that [the Traffic Control repository](#) be downloaded inside the \$GOPATH directory. Specifically, the best location is \$GOPATH/src/github.com/apache/trafficcontrol. Cloning the repository outside of this location will require either linking the actual directory to that point, or moving/copying it there.

**See also:**

The Golang project's GOPATH [wiki page](#)

#### Build Using pkg

This is the easiest way to build all the components of Traffic Control; all requirements are automatically loaded into the image used to build each component. The pkg command can be found at the root of the Traffic Control [repository](#).

#### Requirements

- [Docker](#)
- [Docker Compose](#)<sup>1</sup>

---

<sup>1</sup> This is optional, but recommended. If a docker-compose executable is not available the pkg script will automatically download and run it using a container. This is noticeably slower than running it natively.

### Usage

`./pkg [options] [projects]`

### Options

**-7**

Build RPMs targeting CentOS 7.

Changed in version ATCv6.0.0: Previously, **-7** was implicit if not given. As of ATC version 6.0.0, this is no longer the case, and **-8** is implicit instead.

**-8**

Build RPMs targeting Rocky Linux 8 (default).

Changed in version ATCv6.0.0: Previously, **-7** was implicit if not given. As of ATC version 6.0.0, this is no longer the case, and **-8** is implicit instead.

**-a**

Build all projects, including optional ones.

**-b**

Build builder Docker images before building projects.

**-d**

Disable compiler optimizations for debugging.

**-f** FILE

Use FILE instead of the default Docker-Compose file (`./infrastructure/docker/build/docker-compose.yml`).

**-h**

Print help message and exit.

New in version ATCv6.1.0.

**-l**

List available projects.

**Caution:** This lists only the projects that are built by default if none are specified, not *all* projects that can be built. See [Issue #6272](#).

**-L**

Don't write logs to files - respects output levels on STDERR/STDOUT as set by **-q/-v**.

**-o**

Build from the optional list. Same as passing **-f** with the option-argument `./infrastructure/docker/build/docker-compose-opt.yml`.

**-p**

Pull builder Docker images, do not build them (default).

- q**  
Quiet mode. Suppresses output (default).
- s**  
Simple output filenames - e.g. `traffic_ops.rpm` instead of `traffic_ops-6.1.0-11637.ec9ff6a6.el8.x86_64.rpm`.  
New in version ATCv6.1.0.
- S**  
Skip building “source RPMs”.  
New in version ATCv6.1.0.
- v**  
Verbose mode. Lists all build output.  
New in version ATCv6.1.0.

If present, `projects` should be one or more project names. When no specific project or project list is given the default projects will be built. Valid projects:

- `ats`<sup>2</sup>
- `docs`<sup>3</sup>
- `fakeorigin_build`<sup>?</sup>
- `grove_build`<sup>?</sup>
- `grovetccfg_build`<sup>?</sup>
- `source`<sup>?</sup>
- `traffic_monitor_build`<sup>?</sup>
- `traffic_ops_build`<sup>?</sup>
- `cache-config_build`<sup>?</sup>
- `traffic_portal_build`<sup>?</sup>
- `traffic_router_build`<sup>?</sup>
- `traffic_stats_build`<sup>?</sup>
- `weasel`<sup>?</sup>

Output `component-version.rpm` files, build logs and source tarballs will be output to the `dist/` directory at the root of the Traffic Control repository directory.

---

<sup>2</sup> This project is “optional”, which means that it cannot be built unless `-o` is given.

<sup>3</sup> This is a default project, which will be built if `pkg` is run with no `projects` argument

### Build Using docker-compose

If the `pkg` script fails, `docker-compose` can still be used to build the projects directly. The compose file can be found at `infrastructure/docker/build/docker-compose.yml` under the repository's root directory. It can be passed directly to `docker-compose`, either from the `infrastructure/docker/build/` directory or by explicitly passing a path to the `infrastructure/docker/build/docker-compose.yml` file via `-f`. It is recommended that between builds `docker-compose down -v` is run to prevent caching of old build steps. The service names are the same as the project names described above in *Usage*, and similar to the `pkg` script, the build results, logs and source tarballs may all be found in the `dist` directory after completion.

---

**Note:** Calling `docker-compose` in the way described above will build `_all_` projects, not just the default projects.

---

#### See also:

[The Docker Compose command line reference](#)

### Build the RPMs Natively

A developer may end up building the RPMs several times to test or *debug* code changes, so it can be desirable to build the RPMs quickly for this purpose. Natively building the RPMs has the lowest build time of any building method.



## Install the Dependencies

Table 73: Build dependencies for Traffic Control

OS/Pack- Man- ager	Common de- pen- den- cies	Traffic Moni- tor	Traffic Ops	Traffic Portal	Traffic Router	Traffic Stats	Grove	Grove TC Con- fig (grovetc- cfg)	Docs
ma- cOS <sup>4</sup> (home- brew)	• rpm	• go	• go	• npm • grunt- cli	• maven	• go	• go	• go	• python3
Rock- yLinux <sup>5</sup> , Red Hat, Fe- dora, Cen- tOS (yum)	• git • rpm- build • rsync			• epel- release • npm • nodejs- grunt- cli	• java- 11- openjdk • maven				• python3- devel • gcc • make
Arch Linux, Man- jaro (pac- man)	• git • rpm- tools • diff	• go	• go	• npm • grunt- cli	• jdk11- openjdk • maven	• go	• go	• go	• python- pip • python- sphinx • make
Win- dows (cyg- win) <sup>6</sup>	• git • rpm- build • rsync				• curl				
Win- dows (choco- latey)?		• golang	• golang	• nodejs	• open- jdk11 •	• golang	• golang	• golang	• python • pip
4.1. Developer’s Guide					maven				289 make

After installing the packages using your platform's package manager,

- Install the *global NPM packages* to build Traffic Portal.
- Install the Python 3 modules used to *build the documentation*.

### Run `build/clean_build.sh` directly

In a terminal, navigate to the root directory of the repository. You can run `build/clean_build.sh` with no arguments to build all components.

90: `build/clean_build.sh` with no arguments

```
build/clean_build.sh
```

This is the equivalent of running

91: `build/clean_build.sh` with all components

```
build/clean_build.sh tarball traffic_monitor traffic_ops traffic_
→portal traffic_router traffic_stats grove grove/grovetccfg docs
```

If any component fails to build, no further component builds will be attempted.

By default, the RPMs will be built targeting Rocky Linux 8. CentOS 7 is also a supported build target. You can choose which RHEL version to build for (8, 7, etc.) by setting the `RHEL_VERSION` environment variable:

---

<sup>4</sup> If you are on macOS, you additionally need to *Get OpenJDK 11 on macOS*.

<sup>5</sup> If you are on Rocky Linux, you need to *download Go directly* instead of using a package manager in order to get the latest Go version. For most users, the desired architecture is AMD64/x86\_64.

<sup>6</sup> If you are on Windows, you need to install **both** the Cygwin packages and the Chocolatey packages in order to build the Apache Traffic Control RPMs natively.

92: Building RPMs that target CentOS 7 without the build host needing to be CentOS 7

```
export RHEL_VERSION=7
```

**Warning:** Although there are no known issues with natively-built RPMs, the official, supported method of building the RPMs is by using *pkg* or *docker-compose*. Use natively-built RPMs at your own risk.

## Building Individual Components

Each Traffic Control component can be individually built, and the instructions for doing so may be found in their respective component's development documentation.

## Building This Documentation

See instructions for *building the documentation*.

### 4.1.4 Debugging inside CDN-in-a-Box

---

**Tip:** For the purposes of development, it may be easier to use *The Development Environment*.

---

Some CDN-in-a-Box components can be used with a debugger to step through lines of code, set breakpoints, see the state of all variables in each scope, etc. at runtime. Components that support debugging:

- *Enroller*
- *t3c on Caches*
  - *t3c on Edge Cache*
  - *t3c on Mid 01 Cache*
  - *t3c on Mid 02 Cache*
- *Traffic Monitor*
- *Traffic Ops*

- *Traffic Router*
- *Traffic Stats*

### Enroller

- In `infrastructure/cdn-in-a-box`, open `variables.env` and set `ENROLLER_DEBUG_ENABLE` to `true`.
- Stop CDN-in-a-Box if it is running and remove any existing volumes. Build/rebuild the `enroller-debug` image each time you have changed `infrastructure/cdn-in-a-box/enroller/enroller.go`. Then, start CDN-in-a-Box.

93: docker-compose command for debugging the CDN in a Box Enroller

```
alias mydc='docker-compose -f docker-compose.yml -f docker-  
compose.expose-ports.yml -f optional/docker-compose.  
debugging.yml'  
mydc down -v  
mydc build enroller  
mydc up
```

- Install an IDE that supports `delve` and create a debugging configuration over port 2343. If you are using VS Code, the configuration should look like this:

94: VS Code launch.json for debugging the CDN in a Box Enroller

```
{  
  "version": "0.2.0",  
  "configurations": [  
    {  
      "name": "Enroller",  
      "type": "go",  
      "request": "attach",  
      "mode": "remote",  
      "port": 2343,  
      "cwd": "${workspaceRoot}/",  
      "remotePath": "/go/src/github.com/apache/  
trafficcontrol/",  
    }  
  ]  
}
```

- Use the debugging configuration you created to start debugging the Enroller. It should connect without first breaking at any line.

For an example of usage, set a breakpoint at the `toSession.CreateDeliveryServiceV30()` call in

`enrollDeliveryService()`, then wait for the Enroller to process a file from `/shared/enroller/deliveryservices/` (only exists within the Docker container).

## t3c on Caches

### t3c on Edge Cache

- Navigate to the `infrastructure/cdn-in-a-box` directory. Remove `infrastructure/cdn-in-a-box/cache/trafficcontrol-cache-config.rpm` because it contains release Go binaries that do not include useful debugging information. Rebuild the RPM with no optimization, for debugging:

95: Remove release RPMs, then build debug RPMs

```
make very-clean
make debug cache/trafficcontrol-cache-config.rpm
```

---

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make debug native cache/trafficcontrol-cache-config.rpm` instead of `make debug cache/trafficcontrol-cache-config.rpm` to build the RPM quickly.

---

- Still in `infrastructure/cdn-in-a-box`, open `variables.env` and set `T3C_DEBUG_COMPONENT_E`
  - `t3c-apply`
  - `t3c-check`
  - `t3c-check-refs`
  - `t3c-check-reload`
  - `t3c-diff`
  - `t3c-generate`
  - `t3c-request`
  - `t3c-update`
- Stop CDN-in-a-Box if it is running and remove any existing volumes. Build the edge-debug image to make sure it uses our fresh `trafficcontrol-cache-config.rpm`. Then, start CDN-in-a-Box:

96: docker-compose command for debugging t3c running on the Edge Cache

```
alias mydc='docker-compose -f docker-compose.yml -f docker-
compose.expose-ports.yml -f optional/docker-compose.
debugging.yml'
mydc down -v
mydc build edge
mydc up -d
mydc logs -f trafficmonitor
```

- Install an IDE that supports delve and create a debugging configuration over port 2347. If you are using VS Code, the configuration should look like this:

97: VS Code launch.json for debugging t3c on the Edge Cache

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "t3c on Edge",
      "type": "go",
      "request": "attach",
      "mode": "remote",
      "port": 2347,
      "cwd": "${workspaceRoot}",
      "remotePath": "/tmp/go/src/github.com/apache/
trafficcontrol",
    }
  ]
}
```

Wait for Traffic Monitor to start, which will indicate that the SSL keys have been generated. Because T3C\_DEBUG\_COMPONENT\_EDGE is set to the name of one of the t3c binaries, t3c will *not* run automatically every minute. Start it manually:

98: Run t3c-apply with debugging enabled

```
[user@computer cdn-in-a-box]$ mydc exec edge t3c apply --run-
mode=badass --traffic-ops-url=https://trafficops.infra.ciab.test --
traffic-ops-user=admin --traffic-ops-password=twelve12 --git=yes --
dispersion=0 --log-location-error=stdout --log-location-
warning=stdout --log-location-info=stdout all
API server listening at: [::]:2347
```

The *API server listening* message is from dlvs, indicating it is ready to accept a connection from your IDE. Note that, unlike the other components, execution of t3c does not begin until your IDE connects to dlvs.

For this example, set a breakpoint at the assignment of “`##OVERRIDDEN##`” + `str` to `newstr` in `torequest.processRemapOverrides()`.

Use the debugging configuration you created to connect to `dlv` and start debugging `t3c`.

### **t3c on Mid 01 Cache**

- Navigate to the `infrastructure/cdn-in-a-box` directory. Remove `infrastructure/cdn-in-a-box/cache/trafficcontrol-cache-config.rpm` because it contains release Go binaries that do not include useful debugging information. Rebuild the RPM with no optimization, for debugging:

99: Remove release RPMs, then build debug RPMs

```
make very-clean
make debug cache/trafficcontrol-cache-config.rpm
```

---

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make debug native cache/trafficcontrol-cache-config.rpm` instead of `make debug cache/trafficcontrol-cache-config.rpm` to build the RPM quickly.

---

- **Still in `infrastructure/cdn-in-a-box`, open `variables.env` and set `T3C_DEBUG_COMPONENT_M`**
  - `t3c-apply`
  - `t3c-check`
  - `t3c-check-refs`
  - `t3c-check-reload`
  - `t3c-diff`
  - `t3c-generate`
  - `t3c-request`
  - `t3c-update`
- Stop CDN-in-a-Box if it is running and remove any existing volumes. Build the mid-debug image to make sure it uses our fresh `trafficcontrol-cache-config.rpm`. Then, start CDN-in-a-Box:

100: docker-compose command for debugging t3c running on the Mid 01 Cache

```
alias mydc='docker-compose -f docker-compose.yml -f docker-  
compose.expose-ports.yml -f optional/docker-compose.  
debugging.yml'  
mydc down -v  
mydc build mid-01  
mydc up -d  
mydc logs -f trafficmonitor
```

- Install an IDE that supports delve and create a debugging configuration over port 2348. If you are using VS Code, the configuration should look like this:

101: VS Code launch.json for debugging t3c on the Mid 01 Cache

```
{  
  "version": "0.2.0",  
  "configurations": [  
    {  
      "name": "t3c on Mid 01",  
      "type": "go",  
      "request": "attach",  
      "mode": "remote",  
      "port": 2348,  
      "cwd": "${workspaceRoot}",  
      "remotePath": "/tmp/go/src/github.com/apache/  
trafficcontrol",  
    }  
  ]  
}
```

Wait for Traffic Monitor to start, which will indicate that the SSL keys have been generated. Because T3C\_DEBUG\_COMPONENT\_MID\_01 is set to the name of one of the t3c binaries, t3c will *not* run automatically every minute. Start it manually:

102: Run t3c-apply with debugging enabled

```
[user@computer cdn-in-a-box]$ mydc exec mid-01 t3c apply --run-  
mode=badass --traffic-ops-url=https://trafficops.infra.ciab.test --  
traffic-ops-user=admin --traffic-ops-password=twelve12 --git=yes --  
dispersion=0 --log-location-error=stdout --log-location-  
warning=stdout --log-location-info=stdout all  
API server listening at: [::]:2348
```

The *API server listening* message is from dlvs, indicating it is ready to accept a connection from your IDE. Note that, unlike the other components, execution of t3c does not begin until your IDE connects to dlvs.



For this example, set a breakpoint at the assignment of “`##OVERRIDDEN##`” + `str` to `newstr` in `torequest.processRemapOverrides()`.

Use the debugging configuration you created to connect to `dlv` and start debugging `t3c`.

### **t3c on Mid 02 Cache**

- Navigate to the `infrastructure/cdn-in-a-box` directory. Remove `infrastructure/cdn-in-a-box/cache/trafficcontrol-cache-config.rpm` because it contains release Go binaries that do not include useful debugging information. Rebuild the RPM with no optimization, for debugging:

103: Remove release RPMs, then build debug RPMs

```
make very-clean
make debug cache/trafficcontrol-cache-config.rpm
```

---

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make debug native cache/trafficcontrol-cache-config.rpm` instead of `make debug cache/trafficcontrol-cache-config.rpm` to build the RPM quickly.

---

- **Still in `infrastructure/cdn-in-a-box`, open `variables.env` and set `T3C_DEBUG_COMPONENT_M`**
  - `t3c-apply`
  - `t3c-check`
  - `t3c-check-refs`
  - `t3c-check-reload`
  - `t3c-diff`
  - `t3c-generate`
  - `t3c-request`
  - `t3c-update`
- Stop CDN-in-a-Box if it is running and remove any existing volumes. Build the mid-debug image to make sure it uses our fresh `trafficcontrol-cache-config.rpm`. Then, start CDN-in-a-Box:

104: docker-compose command for debugging t3c running on the Mid 02 Cache

```
alias mydc='docker-compose -f docker-compose.yml -f docker-  
compose.expose-ports.yml -f optional/docker-compose.  
debugging.yml'  
mydc down -v  
mydc build mid-02  
mydc up -d  
mydc logs -f trafficmonitor
```

- Install an IDE that supports delve and create a debugging configuration over port 2349. If you are using VS Code, the configuration should look like this:

105: VS Code launch.json for debugging t3c on the Mid 02 Cache

```
{  
  "version": "0.2.0",  
  "configurations": [  
    {  
      "name": "t3c on Mid 02",  
      "type": "go",  
      "request": "attach",  
      "mode": "remote",  
      "port": 2349,  
      "cwd": "${workspaceRoot}",  
      "remotePath": "/tmp/go/src/github.com/apache/  
trafficcontrol",  
    }  
  ]  
}
```

Wait for Traffic Monitor to start, which will indicate that the SSL keys have been generated. Because T3C\_DEBUG\_COMPONENT\_MID\_02 is set to the name of one of the t3c binaries, t3c will *not* run automatically every minute. Start it manually:

106: Run t3c-apply with debugging enabled

```
[user@computer cdn-in-a-box]$ mydc exec mid-02 t3c apply --run-  
mode=badass --traffic-ops-url=https://trafficops.infra.ciab.test --  
traffic-ops-user=admin --traffic-ops-password=twelve12 --git=yes --  
dispersion=0 --log-location-error=stdout --log-location-  
warning=stdout --log-location-info=stdout all  
API server listening at: [::]:2349
```

The *API server listening* message is from dlvs, indicating it is ready to accept a connection from your IDE. Note that, unlike the other components, execution of t3c does not begin until your IDE connects to dlvs.

For this example, set a breakpoint at the assignment of “`##OVERRIDDEN##`” + `str` to `newstr` in `torequest.processRemapOverrides()`.

Use the debugging configuration you created to connect to `dlv` and start debugging `t3c`.

## Traffic Monitor

- Navigate to the `infrastructure/cdn-in-a-box` directory. Remove the existing RPMs because they contain release Go binaries do not include useful debugging information. Rebuild the RPMs with no optimization, for debugging:

107: Remove release RPMs, then build debug RPMs

```
make very-clean
make debug traffic_monitor/traffic_monitor.rpm
```

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make debug native traffic_monitor/traffic_monitor.rpm` instead of `make debug traffic_monitor/traffic_monitor.rpm` to build the RPM quickly.

- Still in `infrastructure/cdn-in-a-box`, open `variables.env` and set `TM_DEBUG_ENABLE` to `true`.
- Stop `CDN-in-a-Box` if it is running and remove any existing volumes. Build the `trafficmonitor-debug` image to make sure it uses our fresh `traffic_monitor.rpm`. Then, start `CDN-in-a-Box`:

108: docker-compose command for debugging Traffic Monitor

```
alias mydc='docker-compose -f docker-compose.yml -f docker-
compose.expose-ports.yml -f optional/docker-compose.
debugging.yml'
mydc down -v
mydc build trafficmonitor
mydc up
```

- Install an IDE that supports `delve` and create a debugging configuration over port 2344. If you are using VS Code, the configuration should look like this:

109: VS Code launch.json for debugging Traffic Monitor

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Traffic Monitor",
```

(continues on next page)

(continued from previous page)

```

        "type": "go",
        "request": "attach",
        "mode": "remote",
        "port": 2344,
        "cwd": "${workspaceRoot}",
        "remotePath": "/tmp/go/src/github.com/apache/
↪trafficcontrol",
    }
]
}

```

- Use the debugging configuration you created to start debugging Traffic Monitor. It should connect without first breaking at any line.

For an example of usage, set a breakpoint at the `o.m.RLock()` call in `ThreadsafeEvents.Get()`, then visit <http://trafficmonitor.infra.ciab.test/publish/EventLog> (see *Traffic Monitor APIs: /publish/EventLog*).

## Traffic Ops

- Navigate to the `infrastructure/cdn-in-a-box` directory. Remove the existing RPMs because they contain release Go binaries do not include useful debugging information. Rebuild the RPMs with no optimization, for debugging:

110: Remove release RPMs, then build debug RPMs

```

make very-clean
make debug traffic_ops/traffic_ops.rpm

```

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make debug native traffic_ops/traffic_ops.rpm` instead of `make debug traffic_ops/traffic_ops.rpm` to build the RPM quickly.

- Still in `infrastructure/cdn-in-a-box`, open `variables.env` and set `TO_DEBUG_ENABLE` to `true`.
- Stop CDN-in-a-Box if it is running and remove any existing volumes. Build the `trafficops-debug` image to make sure it uses our fresh `traffic_ops.rpm`. Then, start CDN-in-a-Box:

111: docker-compose command for debugging Traffic Ops

```

alias mydc='docker-compose -f docker-compose.yml -f docker-
↪compose.expose-ports.yml -f optional/docker-compose.
↪debugging.yml'
mydc down -v

```

(continues on next page)

(continued from previous page)

```
mydc build trafficops
mydc up
```

- Install an IDE that supports delve and create a debugging configuration over port 2345. If you are using VS Code, the configuration should look like this:

112: VS Code launch.json for debugging Traffic Ops

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Traffic Ops",
      "type": "go",
      "request": "attach",
      "mode": "remote",
      "port": 2345,
      "cwd": "${workspaceRoot}",
      "remotePath": "/tmp/go/src/github.com/apache/
↪trafficcontrol",
    }
  ]
}
```

- Use the debugging configuration you created to start debugging Traffic Ops. It should connect without first breaking at any line.

For an example of usage, set a breakpoint at the `log.Debugln()` call in `TOPProfile.Read()`, then visit <https://trafficportal.infra.ciab.test/api/4.0/profiles> (after logging into *Traffic Portal*).

## Traffic Router

- Navigate to the `infrastructure/cdn-in-a-box` directory.
- In `variables.env`, set `TR_DEBUG_ENABLE` to `true`.
- Install a debugging-capable Java IDE or text editor of your choice. If unsure, install IntelliJ IDEA Community Edition.
- At the base of the repository (not in the `cdn-in-a-box` directory), open the `traffic_router` directory in your IDE.
- Add a new “Remote” (Java) debug configuration. Use port 5005.
- Start CDN-in-a-Box, including the “expose ports” “debugging” compose files:

113: docker-compose command for debugging Traffic Router

```
alias mydc='docker-compose -f docker-compose.yml -f docker-
compose.expose-ports.yml -f optional/docker-compose.
debugging.yml'
mydc down -v
mydc build trafficrouter
mydc up -d
mydc logs --follow trafficrouter
```

- Watch the trafficrouter container's log. After DNS and certificate operations, the enroller, and Traffic Monitor, Traffic Router will start. Look for Listening for transport dt\_socket at address: 5005 in the example log below:

114: Log of the Docker container for Traffic Router

```
Warning:
The JKS keystore uses a proprietary format. It is
recommended to migrate to PKCS12 which is an industry
standard format using "keytool -importkeystore -
srckeystore /opt/traffic_router/conf/keyStore.jks -
destkeystore /opt/traffic_router/conf/keyStore.jks -
deststoretype pkcs12".
Certificate stored in file <trafficrouter.infra.
ciab.test.crt>

Warning:
The JKS keystore uses a proprietary format. It is
recommended to migrate to PKCS12 which is an industry
standard format using "keytool -importkeystore -
srckeystore /opt/traffic_router/conf/keyStore.jks -
destkeystore /opt/traffic_router/conf/keyStore.jks -
deststoretype pkcs12".
Waiting for enroller initial data load to complete.
...
Waiting for enroller initial data load to complete.
...
Waiting for enroller initial data load to complete.
...
Waiting for enroller initial data load to complete.
...
Waiting for enroller initial data load to complete.
...
Waiting for enroller initial data load to complete.
...
Waiting for enroller initial data load to complete.
...
```

(continues on next page)

```
Waiting for enroller initial data load to complete.  
→...  
Waiting for enroller initial data load to complete.  
→...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
Waiting for Traffic Monitor to start...  
tail: cannot open '/opt/tomcat/logs/catalina.log'┐  
→for reading: No such file or directory  
tail: cannot open '/opt/tomcat/logs/catalina.2020-  
→02-21.log' for reading: No such file or directory  
==> /opt/traffic_router/var/log/traffic_router.log  
→<==  
  
==> /opt/traffic_router/var/log/access.log <==  
Tomcat started.  
tail: '/opt/tomcat/logs/catalina.log' has appeared;  
→ following end of new file  
tail: '/opt/tomcat/logs/catalina.2020-02-21.log'┐  
→has appeared; following end of new file  
  
==> /opt/traffic_router/var/log/traffic_router.log  
→<==
```

## 4.1. Developer's Guide

(continued from previous page)

```

INFO 2020-02-21T05:16:07.557 [Thread-3] org.
↳ apache.traffic_control.traffic_router.protocol.
↳ LanguidPoller - Waiting for state from mbean path_
↳ traffic-router:name=languidState
INFO 2020-02-21T05:16:07.557 [Thread-4] org.
↳ apache.traffic_control.traffic_router.protocol.
↳ LanguidPoller - Waiting for state from mbean path_
↳ traffic-router:name=languidState
INFO 2020-02-21T05:16:07.558 [Thread-5] org.
↳ apache.traffic_control.traffic_router.protocol.
↳ LanguidPoller - Waiting for state from mbean path_
↳ traffic-router:name=languidState
INFO 2020-02-21T05:16:07.559 [Thread-6] org.
↳ apache.traffic_control.traffic_router.protocol.
↳ LanguidPoller - Waiting for state from mbean path_
↳ traffic-router:name=languidState

==> /opt/tomcat/logs/catalina.log <==
Listening for transport dt_socket at address: 5005

Watch for the line that mentions port 5005 -----^^^^

==> /opt/tomcat/logs/catalina.2020-02-21.log <==
21-Feb-2020 05:16:07.359 WARNING [main] org.apache.
↳ traffic_control.traffic_router.protocol.
↳ LanguidNioProtocol.<clinit> Adding BouncyCastle provider
21-Feb-2020 05:16:07.452 WARNING [main] org.apache.
↳ traffic_control.traffic_router.protocol.
↳ LanguidNioProtocol.<init> Serving wildcard certs for_
↳ multiple domains
21-Feb-2020 05:16:07.459 WARNING [main] org.apache.
↳ traffic_control.traffic_router.protocol.
↳ LanguidNioProtocol.<init> Serving wildcard certs for_
↳ multiple domains
21-Feb-2020 05:16:07.459 WARNING [main] org.apache.
↳ traffic_control.traffic_router.protocol.
↳ LanguidNioProtocol.<init> Serving wildcard certs for_
↳ multiple domains
21-Feb-2020 05:16:07.461 INFO [main] org.apache.
↳ traffic_control.traffic_router.protocol.
↳ LanguidNioProtocol.setSslImplementationName_
↳ setSslImplementation: org.apache.traffic_control.traffic_
↳ router.protocol.RouterSslImplementation

```

- When you see that Tomcat is listening for debugger connections on port 5005, start debugging using the debug configuration that you created.



## Traffic Stats

- Navigate to the `infrastructure/cdn-in-a-box` directory. Remove the existing RPMs because they contain release Go binaries do not include useful debugging information. Rebuild the RPMs with no optimization, for debugging:

115: Remove release RPMs, then build debug RPMs

```
make very-clean
make debug traffic_stats/traffic_stats.rpm
```

**Tip:** If you have gone through the steps to *Build the RPMs Natively*, you can run `make debug native traffic_stats/traffic_stats.rpm` instead of `make debug traffic_stats/traffic_stats.rpm` to build the RPMs quickly.

- Still in `infrastructure/cdn-in-a-box`, open `variables.env` and set `TS_DEBUG_ENABLE` to `true`.
- Stop CDN-in-a-Box if it is running and remove any existing volumes. Build the `trafficstats-debug` image to make sure it uses our fresh `traffic_stats.rpm`. Then, start CDN-in-a-Box:

116: docker-compose command for debugging Traffic Stats

```
alias mydc='docker-compose -f docker-compose.yml -f docker-
compose.expose-ports.yml -f optional/docker-compose.
debugging.yml '
mydc down -v
mydc build trafficstats
mydc up
```

- Install an IDE that supports `delve` and create a debugging configuration over port 2346. If you are using VS Code, the configuration should look like this:

117: VS Code launch.json for debugging Traffic Stats

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Traffic Stats",
      "type": "go",
      "request": "attach",
      "mode": "remote",
      "port": 2346,
      "cwd": "${workspaceRoot}",
      "remotePath": "/tmp/go/src/github.com/apache/
trafficcontrol",
```

(continues on next page)

(continued from previous page)

```

    }
  ]
}
```

- Use the debugging configuration you created to start debugging Traffic Stats. It should connect without first breaking at any line.

For an example of usage, set a breakpoint at the `http.Get()` call in `main.getURL()`, then wait 10 seconds for the breakpoint to be hit.

## Troubleshooting

- If you are debugging a Golang project and you don't see the values of all variables, or stepping to the next line puts you several lines ahead, rebuild the Docker image with an RPM built using `make debug`.

### 4.1.5 Documentation Guidelines

The Apache Traffic Control documentation is written in RST (reStructuredText) and uses the [Sphinx documentation build system](#) to convert these into the desired output document format. This collection of guidelines does **not** aim to be a primer in RST, but merely a style guide regarding how the components of the document ought to be formatted and structured. It may also point out some features of the markup language of which a writer may not be aware.

**See also:**

[The docutils RST reference](#).

## Building

This documentation uses the [Sphinx documentation build system](#), and as such requires a Python3 version that is at least 3.4.1. It also has dependency on Sphinx, and Sphinx extensions and themes. All of these can be easily installed using *pip* by referencing the requirements file like so:

118: Run from the Repository's Root Directory

```
python3 -m pip install --user -r docs/source/requirements.txt
```

Once all dependencies have been satisfied, build using the Makefile at `docs/Makefile`.

Alternatively, it is also possible to *Build Using pkg* or to *Build Using docker-compose*, both of which will output a documentation “tarball” to `dist/`.

## Writing

When writing documentation, the most important things to remember are:

- **Spell Check.** Most text editors have this built-in (e.g. F6 in Sublime Text) or have plugins that will do this for you.
- **Proof-Read.** Spell-checkers won't catch grammatical errors or poor wording, so it's very important to actually proof-read all documentation *before* submitting it in a Pull Request.
- **Make Sure the Documentation Actually Builds.** Please actually verify the documentation not only builds, but builds *correctly*. That means there probably shouldn't be any warnings, no malformed tables etc. and it also means that new documentation is actually accessible once built. It's not enough to create a new *something.rst* file, that file must actually be linked to from some other, already included document. Some warnings may be considered acceptable, but do be prepared to defend them.
- **Traffic Ops UI is Dead.** Do not ever create documentation that references or includes images of the Traffic Ops UI. That is officially dead now, and if the documentation being created is best made with references to a user-friendly UI, such references, examples and/or images should all be to/of Traffic Portal.

## Formatting

Whenever possible, avoid specifying manual line breaks, except as required by RST syntax. Extremely long lines will be wrapped by the user-agent, compiler, or output format as necessary. A single blank line may be used to separate paragraphs. This means that the 'flow break' character should never need to be used, i.e. no line in the documentation should ever match the regular expression `^\|`.

## Abbreviations

When using an abbreviation, acronym or initialism for the first time on a page, it **must** be named fully and followed by the abbreviation in parentheses e.g. "Fully Qualified Domain Name (FQDN)". Strictly speaking, the *best* way to create an abbreviation is to always fully name it in parentheses immediately following the abbreviation using the `:abbr:` RST text role e.g. `:abbr:`FQDN (Fully Qualified Domain Name)``, but it's not reasonable to expect that of everyone. Some abbreviations can be assumed to be understood by the documentation's target audience, and do not need full naming; they are general, basic networking and computing terms including (though not strictly limited to):

- API
- CSS
- DNS
- HTML
- HTTP

- HTTPS
- IP/IPv4/IPv6
- ISO
- JPG
- JSON
- PDF
- PNG
- RPM
- SQL
- SSL
- SVG
- TCP
- TLS
- UDP
- URL
- URI
- XML
- YAML

Please do **not** abbreviate Traffic Control terms e.g. *Cache Group*, *Delivery Service*. See *Terms* for the proper way to use these terms.

## Floating Objects

“Floating objects” are images, tables, source code listings, and equations. These may not be placed relative to other content exactly as shown in the source RST document, as it may be necessary to move them for e.g. page breaks in PDF documents so they are not split across pages.

## Figures

Images should *always* be included inside of a `.. figure` directive. **Always** caption figures to make their purpose clear, as well as to make them directly link-able inside of the document and include them in figure listings. Though not syntactically required, figures should, in general be sized explicitly. The size should not be *absolute*, however; i.e. use **70%** not **540px**. Figures should, in general, be centered on the page. When drawings, graphs, or diagrams are included they should ideally be provided in both SVG and PNG formats, and included using globbing as `filename.*`. This will use the appropriate format for the output type.

## Source Code Listings

Do not ever use the double-colon (`::`) directive to mark a section of text as a source code listing. This not only doesn't support direct linking or provide a caption, but also uses the default "domain" - which is Python - for syntax highlighting. Instead, use `.. code-block` to include source code in the documentation. Source code must always be left-aligned, so do not provide any configuration options that would alter the default.

119: Example Usage of the code-block Directive

```
.. code-block:: syntax
   :caption: A short, meaningful caption
   :linenos:

   ``:linenos:`` is an optional field which will include line numbers,
   ↪in the listing. 'syntax'
      should be the name of a valid Pygments syntax.
```

## Tables

Tables should be included in `.. table` directive bodies, **never** as a floating, block-quoted tabular environment. This ensures that all tables will be captioned, which makes their purpose clear and makes them directly link-able in the output as well as includes them in table listings. Tables should avoid wrapping lines until they reach 215 characters in width in the source `RESTRUCTUREDTEXT` document (including indentation which should be counted as 4 characters per `TAB`). No table may ever exceed 215 characters in width. Tables should, in general be left-aligned (which is the default configuration). For the usage or command-line flags or options of a utility, use an "option list" or the `.. program` and `.. option` directives instead of a table.

## Indentation

Firstly, indentation of a text paragraph is rarely required. Doing so does not "link" the text with a heading in any way, text falls beneath a section or sub-section purely by being literally *beneath* that heading. When placing source code into a source code listing, any indentation may be used for the source code (typically should represent the actual preferred indentation of the code's respective project were it included in the repository), but to avoid ambiguity in indentation used in the documentation versus indentation used in the source code, all documentation indentation should be done using *one (1) TAB character* and **never spaces**.

### Lists

When making a list, consider first what *kind* of list it is. A list only needs to be enumerated if the enumeration has meaning e.g. a list of hierarchically ordered data or a sequential list of steps to accomplish a task or desired state. When enumerating list items, use `#.` instead of literal numbers, as this will enumerate automatically which makes modification of the list much easier at a later date. Unordered lists may use `-` or `*` for each item. Lists do not need to be indented beyond the current paragraph level. If a list is an unordered list of terms and their definitions, use a “definition list” instead of any other kind of list. If a list is a list of fields in a document or object, use a “field list” instead of any other kind of list. If a list is a list of parameters, arguments, or flags used by a command-line utility, use an “option list” instead of any other kind of list.

### Notes and Footnotes

Instead of `**NOTE**` or similar, consider using the `.. note` directive, or one of the appropriate admonitions supported by RST:

**attention** The default admonition that calls attention to something without any specific semantics or attached context. Use when none of the others seem appropriate.

**caution** Includes cautionary information. Should be used when advising the reader that the containing section includes instructions or information that frequently confuse people/trip people up, and how to avoid these pitfalls.

**danger** Advises the reader of potential security risks or system damage that could occur as a result of following instructions in the containing section, or as a result of making assumptions about and/or improperly utilizing information in the containing section.

**error** Denotes an error. This has limited uses in the ATC documentation.

**hint** Offers a hint to nudge readers in the direction of a solution. This has limited uses in the ATC documentation.

**impl-detail** Contains information describing the details of an implementation of a feature described in the containing section. For an example, see [the DSCP section of the Delivery Services page](#).

---

**Note:** This is an *extension* provided by the ATC documentation build system, and will not appear in the Sphinx project documentation nor the reStructuredText standard.

---

**important** Contains information that is important to consider while reading the containing section. Typically content that is related to a section’s content in an important way should appear as content of that section, but if a section is in danger of readers “skimming” it for information this can be useful to catch their eye.

**note** Used to segregate content that is only tangentially related to the containing section, but is noteworthy nonetheless. Historically the most used admonition, containing caveats, exceptions etc.

**tip** Provides the reader with information that can be helpful - particularly to users/administrators/developers new to ATC - but not strictly necessary to understand the containing section.

**warning** Warns the reader about possible unintended consequences of following instructions/utilizing information in the containing section. If the behavior warned about constitutes a security risk and/or serious damage to one or more systems - including clients and origins - please use `.. danger` instead.

In a similar vein, instead of e.g. “(See also: some-link-or-reference)” please use the special `.. seealso` admonition. If the same admonition is required more than twice on the same page, it most likely ought to be a footnote instead. Footnotes should ideally use human-readable labels or, failing that, be labeled sequentially in the order of appearance. Footnotes should appear at the end of the major section in which they first or last appear. In practice, however, placement of the footnote is left to the writer’s discretion.

## Section Headings

When deciding on the name of a section heading, it is important to select a name that both accurately reflects the content it contains and is suitable for reference later e.g. “Health Protocol” is good, but “Configuring Multi-Site Origin” as the title of a page which not only explains MSO configuration but also the concept is not good. Section headings follow a hierarchy, and for consistency’s sake should follow this particular hierarchy:

1. Document title. There should only be one of these per page, and it should be the first heading on the page. This will also make the contained text the “Page Title” in HTML output (i.e. `<title>Page Title</title>` in the `<head>`).

120: Document Title

```
*****
Document Title
*****
```

2. Section header. This should represent a main topic of the page

121: Section Header

```
Section Header
=====
```

3. Subsection header. This should represent a key piece of a main topic on the page

### 122: Subsection Header

**Subsection Header**

-----

4. Sub-Subsection header. This represents a group of content logically separate from the rest of the subsection but still related to the content thereof. It is also acceptable to use this as an “aside” containing information only tangentially related to the subsection content.

### 123: Sub-Subsection Header

**Sub-Subsection Header**

.....

5. Aside or Sub-Sub-Subsection header. This is the lowest denomination of header, and should almost always be used exclusively for “asides” which contain information only tangentially related to the sub-subsection content.

### 124: Aside

**Aside**

....

Section headings should *always* follow this order exactly, and **never** skip levels (which will generally cause a failure to compile properly). These can be thought of as the equivalents of the HTML tags <h1> through <h5>, respectively. Sectioning should never require more specificity than can be provided by an “Aside”. Please do not use **bold text** in lieu of a proper section heading. When referencing information in another section on the same page, please do not refer to the current placement of the referenced content relative to the referencing content. For example, instead of “as discussed below”, use “as discussed in *Terms*”.

## Terms

Please always spell out the entire name of any Traffic Control terms used in the definition. For example, a collection of *cache servers* associated with a certain physical location is called a “Cache Group”, not a “CG”, “cachegroup”, “cache location” etc. A subdomain and collection of *cache servers* responsible collectively for routing traffic to a specific origin is called a *Delivery Service*”, not a “DS”, “deliveryservice” etc. Similarly, always use *full* permissions role names e.g. “operations” not “oper”. This will ensure the *Glossary* is actually helpful. To link a term to the glossary, use the `:term:` role. This should be done for virtually every use of a Traffic Control term, e.g. `:term:`Cache Group`` will render as: *Cache Group*. Generally speaking, be wary of using the word “cache”. To most people that means the *actual* cache on a hard disk somewhere. This word is frequently confused with “*cache server*”, which - when accurate - is always preferred over “cache”.



## Documenting API Routes

Follow all of the formatting conventions in *Formatting*. Maintain the structural format of the API documentation as outlined in the *Traffic Ops API* section. API routes that have variable paths e.g. `profiles/{{ID}}` should use *mustache templates* **not** the Mojolicious-specific `:param` syntax. This keeps the templates generic, familiar, and reflects the inability of a request path to contain procedural instructions or program logic. Please do not include the `/api/2.x/` part of the request path for Traffic Ops API endpoints. If an endpoint is unavailable prior to a specific version, use the `.. versionadded` directive to indicate that version. Likewise, do not make a new page for an endpoint when it changes across versions, instead call out the changes using the `.. versionchanged` directive. If an endpoint should not be used because newer endpoints provide the same functionality in a better way, use the `.. deprecated` directive to link to them and explain why they are better.

When documenting an API route, be sure to include *all* methods, request/response JSON payload fields, path parameters, and query parameters, whether they are optional or not. When describing a field in a JSON payload, remember that JSON does not have “hashes” it has “objects” or even “maps”. When documenting path parameters such as *Profile ID* in `profiles/{{ID}}`, consider that the endpoint path cannot be formed without defining **all** path parameters, and so to label them as “required” is superfluous.

The “Response Example” must **always** exist. “TODO” is **not** an acceptable Response Example for new endpoints. The “Request Example” must only exist if the request requires data in the body (most commonly this will be for PATCH, POST and PUT methods). It is, however, strongly advised that a request example be given if the endpoint takes Query Parameters or Path Parameters, and it is required if the Response Example is a response to a request that used a query or path parameter. If the Request Example *is* present, then the Response Example **must** be the appropriate response **to that request**. When generating Request/Response Examples, attempt to use the *CDN in a Box* environment whenever possible to provide a common basis and familiarity to new users who likely set up “CDN in a Box” as a primer for understanding CDNs/Traffic Control. Responses are sometimes hundreds of lines long, and in those cases only as much as is required for an understanding of the structure needs to be included in the example - along with a note mentioning that the output was trimmed. Also always attempt to place structure explanations before any example so that the content of the example can be understood by the reader (though in general the placement of a floating environment like a code listing is not known at compile-time). Whenever possible, the Request and Response examples should include the *complete HTTP stack*, which captures behavior like Query Parameters, Path Parameters and HTTP cookie operations like those used by e.g. *logs*. A few caveats to the “include all headers” rule:

- The Host header ought to reflect the actual hostname of the Traffic Ops server - which should be “trafficops.infra.ciab.test” for the CDN in a Box environment. This can be polluted when requests are made to a remotely running CDN in a Box on a different server.
- The “mojolicious” cookie is extremely long and potentially insecure to publicly show. As such, a placeholder should be used for its value, preferably “...”.
- The Content-Type header sent by `curl(1)` (and possibly others) is always `application/x-www-form-urlencoded` regardless of the actual content (unless over-

ridden). Virtually all payloads accepted by the API must be JSON, so this should be modified to reflect that when appropriate e.g. `application/json`.

- API output is often beautified by inserting line breaks and indentation, which will make the `Content-Length` header (if any) incorrect. Don't worry about fixing that - just try to leave the output as close as possible to what will actually be returned by leaving it the way it is.

File names should reflect the request path of the endpoint, e.g. a file for an endpoint of the Traffic Ops API `/api/4.0/foo/{{fooID}}/bar/{{barID}}` should be named `foo_fooID_bar_barID.rst`. Similarly, reference labels linking to the document title for API route pages should follow the convention: `<component>-api-<path>` in all lowercase where `<component>` is an abbreviated Traffic Control component name e.g. `to` and `<path>` is the request path e.g. `foo_bar`. So a label for an endpoint of the Traffic Ops API at `/api/4.0/foo_bar/{{ID}}` should be `to-api-foo_bar-id`.

## Extension

The ATC documentation provides an extension to the standard roles and directives offered by Sphinx, located at `docs/source/_ext/atc.py`. It provides the following roles and directives:

**impl-detail** An admonition directive used to contain implementation-specific notes on a subject.

125: Example impl-detail usage

```
.. impl-detail:: Implementation-specific information here.
```

This example usage renders like so:

---

### Implementation Detail

Implementation-specific information here.

---

**atc-file** Creates a link to the specified file on the master branch of the ATC repository. For example, `“:atc-file: `docs/source/development/documentation_guidelines`”` renders as a link to the source of this documenting section like so: [docs/source/development/documentation\\_guidelines](#). You can also link to directories as well as files.

**issue** A text role that can be used to easily link to GitHub Issues for the ATC repository. For example, `“:issue: `1`”` renders as: [Issue #1](#).

**pr** A text role that can be used to easily link to GitHub Pull Requests for the ATC repository. For example, `“:pr: `1`”` renders as [Pull Request 1](#).

**pull-request** A synonym for `pr`

**godoc** A text role that can be used to easily link to the documentation for any Go package, type, or function/method (grouped constants/variables not supported). For example, `“:godoc: `net/http.HandlerFunc`”` renders as [net/http.HandlerFunc](#).

**atc-godoc** This is provided for convenience, and is identical to `:godoc:` except that it is assumed to be relative to the Apache Traffic Control project. For example, `:atc-godoc: `lib/go-rfc.MimeType.Quality`` renders as [github.com/apache/trafficcontrol/lib/go-rfc.MimeType.Quality](https://github.com/apache/trafficcontrol/lib/go-rfc.MimeType.Quality).

**to-godoc** This is provided for convenience, and is identical to `:godoc:` except that it is assumed to be relative to the [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang) package. For example, `:to-godoc: `api.APIInfo`` renders as [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.APIInfo](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.APIInfo).

## 4.1.6 Godocs

### Linking to Godocs

As mentioned in *Documentation Guidelines*, you can use the `:godoc:` role, the `:atc-godoc:` role, and the `:to-godoc:` role to link to Godocs from the ATC documentation.

### Keeping pkg.go.dev from hiding the Apache Traffic Control Godocs

If less than 75% of ATC's `LICENSE` file contains OSS licenses according to [pkg.go.dev](https://pkg.go.dev) license detection system `pkg.go.dev` will hide all of ATC's Godocs for that particular ATC version, saying "Documentation not displayed due to license restrictions.". Example: <https://pkg.go.dev/github.com/apache/trafficcontrol@v5.1.0+incompatible/lib/go-tc>

When listing a dependency in the `LICENSE` file as part of [bundling a new dependency](#), make sure that the license "pointer" adheres to this format:

126: The `atc-dependency.lre` license exception from [golang.org/x/pkg/site](https://golang.org/x/pkg/site)

```
This product bundles __4__, which
(( is || are ))
available under
(( a || an ))
(( Apache-2.0 || BSD-2-Clause || BSD-3-Clause || MIT ))
license.
__15__
(( /* || .css || .js || .scss ))
(( ./licenses/__4__ || ./vendor/__16__/LICENSE
(( .libyaml || .md || .txt ))??
))
Refer to the above license for the full text.
```

Example:

127: An example of a bundled dependency license pointer

```
This product bundles go-acme/lego, which is available under an MIT_
→license.
@vendor/github.com/go-acme/lego/*
./vendor/github.com/go-acme/lego/LICENSE
Refer to the above license for the full text.
```

The ATC repository includes a GitHub Actions workflow ([.github/workflows/license-file-coverage.yml](#)) to verify that changes to the LICENSE file will not result in pkg.go.dev hiding the Apache Traffic Control Godocs.

### 4.1.7 Traffic Monitor

#### Introduction

Traffic Monitor is an HTTP service application that monitors *cache servers*, provides health state information to Traffic Router, and collects statistics for use in tools such as Traffic Portal and Traffic Stats. The health state provided by Traffic Monitor is used by Traffic Router to control which *cache servers* are available on the CDN.

#### Software Requirements

To work on Traffic Monitor you need a Unix-like (MacOS and Linux are most commonly used) environment that has a working install of Go version 1.7+.

#### Project Tree Overview

traffic\_monitor/ - base directory for Traffic Monitor.

- cache/ - Handler for processing cache results.
- config/ - Application configuration; in-memory objects from `traffic_monitor.cfg`.
- crconfig/ - data structure for deserializing the CDN *Snapshot* (historically named “CR-Config”) from JSON.
- deliveryservice/ - aggregates *Delivery Service* data from cache results.
- deliveryservicedata/ - *Delivery Service* data structures. This exists separate from `deliveryservice` to avoid circular dependencies.
- enum/ - enumerations and name alias types.
- health/ - functions for calculating *cache server* health, and creating health event objects.
- manager/ - manager goroutines (microthreads).

- `health.go` - Health request manager. Processes health results, from the health poller -> fetcher -> manager. The health poll is the “heartbeat” containing a small amount of stats, primarily to determine whether a *cache server* is reachable as quickly as possible. Data is aggregated and inserted into shared thread-safe objects.
- `manager.go` - Contains `Start` function to start all pollers, handlers, and managers.
- `monitorconfig.go` - Monitor configuration manager. Gets data from the monitor configuration poller, which polls Traffic Ops for changes to which caches are monitored and how.
- `opsconfig.go` - Ops configuration manager. Gets data from the ops configuration poller, which polls Traffic Ops for changes to monitoring settings.
- `peer.go` - Peer manager. Gets data from the peer poller -> fetcher -> handler and aggregates it into the shared thread-safe objects.
- `stat.go` - Stat request manager. Processes stat results, from the stat poller -> fetcher -> manager. The stat poll is the large statistics poll, containing all stats (such as HTTP response status codes, transactions, *Delivery Service* statistics, and more). Data is aggregated and inserted into shared thread-safe objects.
- `statecombiner.go` - Manager for combining local and peer states, into a single combined states thread-safe object, for serving the `CrStates` endpoint.
- `datareq/` - HTTP routing, which has thread-safe health and stat objects populated by stat and health managers.
- `peer/` - Manager for getting and populating peer data from other Traffic Monitors
- `srvhttp/` - HTTP(S) service. Given a map of endpoint functions, which are lambda closures containing aggregated data objects. If HTTPS, the HTTP service will redirect to HTTPS.
- `static/` - Web interface files (markup, styling and scripting)
- `threadsafe/` - Thread-safe objects for storing aggregated data needed by multiple goroutines (typically the aggregator and HTTP server)
- `todata/` - Data structure for fetching and storing Traffic Ops data needed from the CDN *Snapshot*. This is primarily mappings, such as *Delivery Service* servers, and server types.
- `trafficopswrapper/` - Thread-safe wrapper around the Traffic Ops client. The client used to not be thread-safe, however, it mostly (possibly entirely) is now. But, the wrapper also serves to overwrite the Traffic Ops `monitoring.json` values, which are live, with values from the CDN *Snapshot*.

## Architecture

At the highest level, Traffic Monitor polls *cache server*s, aggregates their data and availability, and serves it at HTTP endpoints in JSON format. In the code, the data flows through microthread (goroutine) pipelines. All stages of the pipeline are independently running microthreads<sup>1</sup>. The pipelines are:

**stat poll** Polls caches for all statistics data. This should be a slower poll, which gets a lot of data.

**health poll** Polls caches for a tiny amount of data, typically system information. This poll is designed to be a heartbeat, determining quickly whether the *cache server* is reachable. Since it's a small amount of data, it should poll more frequently.

**peer poll** Polls Traffic Monitor peers for their availability data, and aggregates it with its own availability results and that of all other peers.

**monitor config** Polls Traffic Ops for the list of Traffic Monitors and their info.

**ops config** Polls for changes to the Traffic Ops configuration file `traffic_ops.cfg`, and sends updates to other pollers when the configuration file has changed.

- The ops config manager also updates the shared Traffic Ops client, since it's the actor which becomes notified of configuration changes requiring a new client.
- The ops config manager also manages, creates, and recreates the HTTP server, since Traffic Ops configuration changes necessitate restarting the HTTP server.

All microthreads in the pipeline are started by `manager/manager.go:Start()`.

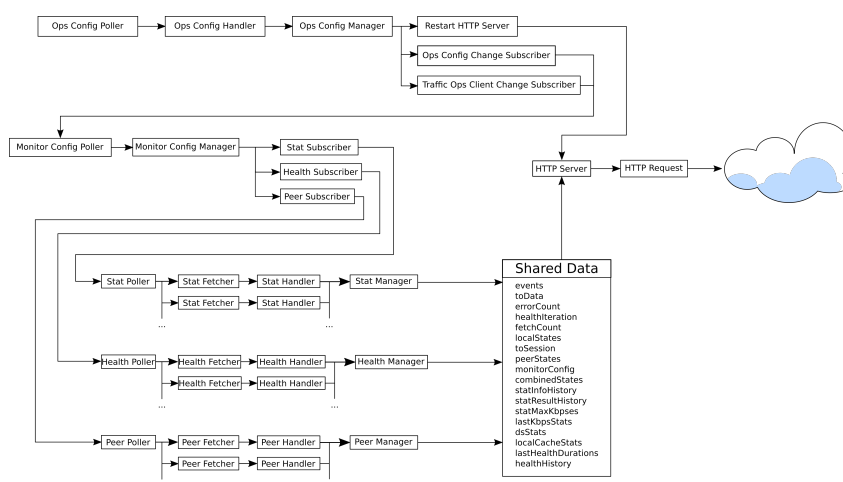


Fig. 51: Pipeline Overview

<sup>1</sup> Technically, some stages which are one-to-one simply call the next stage as a function. For example, the Fetcher calls the Handler as a function in the same microthread. But this isn't architecturally significant.

## Stat Pipeline

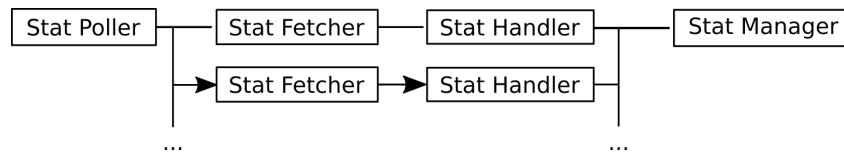


Fig. 52: The Stats Pipeline

**poller** `common/poller/poller.go:HttpPoller.Poll()`. Listens for configuration changes (from the Ops Configuration Manager), and starts its own, internal microthreads - one for each cache to poll. These internal microthreads call the Fetcher at each cache's poll interval.

**fetcher** `common/fetcher/fetcher.go:HttpFetcher.Fetch()`. Fetches the given URL, and passes the returned data to the Handler, along with any errors.

**handler** `traffic_monitor/cache/cache.go:Handler.Handle()`. Takes the given result and does all data computation possible with the single result. Currently, this computation primarily involves processing the de-normalized ATS data into Go structs, and processing System data into 'OutBytes', KBPS, etc. Precomputed data is then passed to its result channel, which is picked up by the Manager.

**manager** `traffic_monitor/manager/stat.go:StartStatHistoryManager()`. Takes preprocessed results, and aggregates them. Aggregated results are then placed in shared data structures. The major data aggregated are *Delivery Service* statistics, and *cache server* availability data. See *Aggregated Stat Data* and *Aggregated Availability Data*.

## Health Pipeline

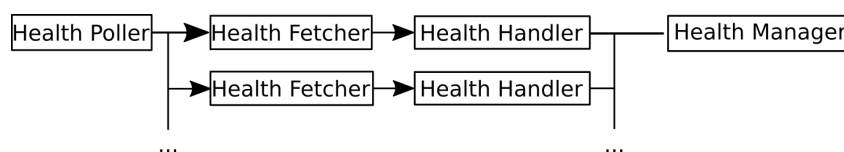


Fig. 53: The Health Pipeline

**poller** `common/poller/poller.go:HttpPoller.Poll()`. Same poller type as the Stat Poller pipeline, with a different handler object.

**fetcher** `common/fetcher/fetcher.go:HttpFetcher.Fetch()`. Same fetcher type as the Stat Poller pipeline, with a different handler object.

**handler** `traffic_monitor/cache/cache.go:Handler.Handle()`. Same handler type as the Stat Poller pipeline, but constructed with a flag to not pre-compute anything. The health endpoint is of the same form as the stat endpoint, but doesn't return all stat data. So, it doesn't pre-compute like the Stat Handler, but only processes the system data, and passes the processed result to its result channel, which is picked up by the Manager.



**manager** `traffic_monitor/manager/health.go:StartHealthResultManager()`.

Takes preprocessed results, and aggregates them. For the Health pipeline, only health availability data is aggregated. Aggregated results are then placed in shared data structures (`lastHealthDurationsThreadsafe`, `lastHealthEndTimes`, etc). See [Aggregated Availability Data](#).

## Peer Pipeline

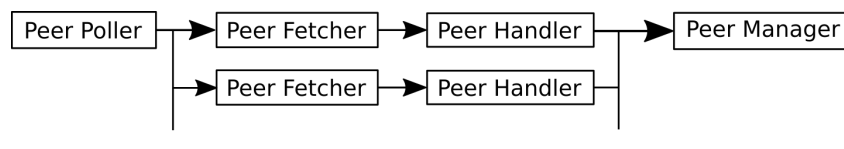


Fig. 54: The Peers Pipeline

**poller** `common/poller/poller.go:HttpPoller.Poll()`. Same poller type as the Stat and Health Poller pipelines, with a different handler object. Its configuration changes come from the Monitor Configuration Manager, and it starts an internal microthread for each peer to poll.

**fetcher** `common/fetcher/fetcher.go:HttpFetcher.Fetch()`. Same fetcher type as the Stat and Health Poller pipeline, with a different handler object.

**handler** `traffic_monitor/cache/peer.go:Handler.Handle()`. Decodes the JSON result into an object, and without further processing passes to its result channel, which is picked up by the Manager.

**manager** `traffic_monitor/manager/peer.go:StartPeerManager()`. Takes JSON peer Traffic Monitor results, and aggregates them. The availability of the Peer Traffic Monitor itself, as well as all [cache server](#) availability from the given peer result, is stored in the shared `peerStates` object. Results are then aggregated via a call to the `combineState()` lambda, which signals the State Combiner microthread (which stores the combined availability in the shared object `combinedStates`; See [State Combiner](#)).

## Monitor Config Pipeline

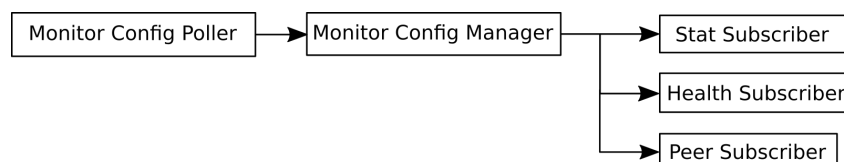


Fig. 55: The Monitor Configuration Pipeline

**poller** `common/poller/poller.go:MonitorConfigPoller.Poll()`. The Monitor Configuration poller, on its interval, polls Traffic Ops for the Monitor configuration, and writes the polled value to its result channel, which is read by the Manager.



**manager** `traffic_monitor/manager/monitorconfig.go:StartMonitorConfigManager()`. Listens for results from the poller, and processes them. Cache changes are written to channels read by the Health, Stat, and Peer pollers. In the Shared Data objects, this also sets the list of new *Delivery Services* and removes ones which no longer exist, and sets the list of peer Traffic Monitors.

## Ops Config Pipeline

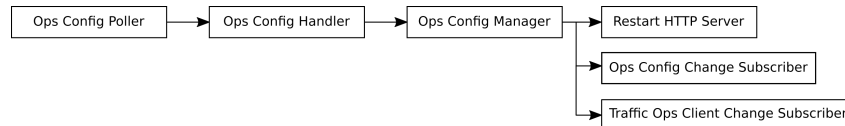


Fig. 56: The Ops Configuration Pipeline

**poller** `common/poller/poller.go:FilePoller.Poll()`. Polls for changes to the Traffic Ops configuration file `traffic_ops.cfg`, and writes the changed configuration to its result channel, which is read by the Handler.

**handler** `common/handler/handler.go:OpsConfigFileHandler.Listen()`. Takes the given raw configuration, un-marshals the JSON into an object, and writes the object to its channel, which is read by the Manager, along with any error.

**manager** `traffic_monitor/manager/monitorconfig.go:StartMonitorConfigManager()`. Listens for new configurations, and processes them. When a new configuration is received, a new HTTP dispatch map is created via `traffic_monitor/datareq/datareq.go:MakeDispatchMap()`, and the HTTP server is restarted with the new dispatch map. The Traffic Ops client is also recreated, and stored in its shared data object. The Ops Configuration change subscribers and Traffic Ops Client change subscribers (the Monitor Configuration poller) are also passed the new Traffic Ops configuration and new Traffic Ops client.

### Events

The events shared data object is passed to each pipeline microthread which needs to signal events. Most of them do. Events are then logged, and visible in the UI as well as an HTTP JSON endpoint. Most events are *cache server* becoming available or unavailable, but include other things such as peer availability changes.

### State Combiner

The State Combiner is a microthread started in `traffic_monitor/manager/manager.go:Start()` via `traffic_monitor/manager/statecombiner.go:StartStateCombiner()`, which listens for signals to combine states. It should be signaled by any pipeline which updates the local or peer availability shared data objects, `localStates` and `peerStates`. It holds the thread-safe shared data objects for local states and peer states, so no data is passed or returned, only a signal. When a signal is received, it combines the local and peer states optimistically. That is, if a *cache server* is marked available locally or by any peer, that *cache server* is marked available in the combined states. There exists a variable to combine pessimistically, which may be set at compile time (it's unusual for a CDN to operate well with pessimistic *cache server* availability). Combined data is stored in the thread-safe shared data object `combinedStates`.

### Aggregated Stat Data

The Stat pipeline Manager is responsible for aggregating stats from all *Edge-tier cache servers*, into *Delivery Services* statistics. This is done via a call to `traffic_monitor/deliveryservice/stat.go:CreateStats()`.

### Aggregated Availability Data

Both the Stat and Health pipelines aggregate availability data received from caches. This is done via a call to `traffic_monitor/deliveryservice/health.go:CalcAvailability()` followed by a call to `combineState()`. The `CalcAvailability` function calculates the availability of each *cache server* from the result of polling it, that is, local availability. The `combineState()` function is a lambda passed to the Manager, which signals the State Combiner microthread, which will combine the local and peer Traffic Monitor availability data, and insert it into the shared data `combinedStates` object.

## HTTP Data Requests

Data is provided to HTTP requests via the thread-safe shared data objects (see [Shared Data](#)). These objects are closed in lambdas created via `traffic_monitor/datareq/datareq.go:MakeDispatchMap()`. This is called by the Ops Configuration Manager when it recreates the HTTP(S) server. Each HTTP(S) endpoint is mapped to a function which closes around the shared data objects it needs, and takes the request data it needs (such as query parameters). Each endpoint function resides in its own file in `traffic_monitor/datareq/`. Because each Go HTTP routing function must be a `http.HandlerFunc`, wrapper functions take the endpoint functions and return `http.HandlerFunc` functions which call them, and which are stored in the dispatch map, to be registered with the HTTP(S) server.

## Shared Data

Processed and aggregated data must be shared between the end of the stat and health processing pipelines, and HTTP requests. The CSP (Communicating Sequential Processes) paradigm of idiomatic Go does not work efficiently with storing and sharing state. While not idiomatic Go, shared mutexed data structures are faster and simpler than CSP manager microthreads for each data object. Traffic Monitor has many thread-safe shared data types and objects. All shared data objects can be seen in `manager/manager.go:Start()`, where they are created and passed to the various pipeline stage microthreads that need them. Their respective types all include the word `Threadsafe`, and can be found in `traffic_monitor/threadsafe/` as well as, for dependency reasons, various appropriate directories. Currently, all thread-safe shared data types use mutexes. In the future, these could be changed to lock-free or wait-free structures, if the performance needs outweighed the readability and correctness costs. They could also easily be changed to internally be manager microthreads and channels, if being idiomatic were deemed more important than readability or performance.

## Disk Backup

The *Traffic Monitor configuration* and *CDN Snapshot* (see [Snapshots](#)) are both stored as backup files (`tmconfig.backup` and `crconfig.backup` or whatever you set the values to in the configuration file). This allows the monitor to come up and continue serving even if Traffic Ops is down. These files are updated any time a valid configuration is received from Traffic Ops, so if Traffic Ops goes down and Traffic Monitor is restarted it can still serve the previous data. These files can also be manually edited and the changes will be reloaded into Traffic Monitor so that if Traffic Ops is down or unreachable for an extended period of time manual updates can be done. If on initial startup Traffic Ops is unavailable then Traffic Monitor will continue through its exponential back-off until it hits the max retry interval, at that point it will create an unauthenticated Traffic Ops session and use the data from disk. It will still poll Traffic Ops for updates though and if it successfully gets through then it will login at that point.

### Formatting Conventions

Go code should be formatted with `gofmt`. See also `CONTRIBUTING.md`.

### Installing The Developer Environment

To install the Traffic Monitor Developer environment:

1. Install [Go](#) version 1.7 or greater
2. Clone the [Traffic Control repository](#) using `git`, into `$GOPATH/src/github.com/apache/trafficcontrol`
3. Change directories into `$GOPATH/src/github.com/apache/trafficcontrol/traffic_monitor`
4. Run `./build.sh`

### Test Cases

Tests can be executed by running `go test ./...` at the root of the `traffic_monitor` project.

### API

*Traffic Monitor APIs*

#### Traffic Monitor APIs

The Traffic Monitor URLs below allow certain query parameters for use in controlling the data returned.

---

**Note:** Unlike *Traffic Ops API endpoints*, no authentication is required for any of these, and as such there can be no special role requirements for a user.

---

#### `/publish/EventLog`

Gets a log of recent changes in the availability of polled caches.

## GET

**Response Type** Array (key 'events' contains an array of all data)

## Response Structure

**event** an entry in the top-level events array

**description** A string containing short description of the event

**hostname** A string containing the server's full hostname

**index** A serial integer that is incremented for each sequential event

**isAvailable** A boolean value indicating whether the server is available following this event

**name** The server's short hostname as a string

**time** A UNIX timestamp as an integer

**type** The type of the server as a string

### 128: Example Response

```
{ "events": [
  {
    "time": 1538417713,
    "index": 67848,
    "description": "REPORTED - loadavg too high (36.37 \u003e 25.
    ↳00) (health)",
    "name": "edge",
    "hostname": "edge",
    "type": "EDGE",
    "isAvailable": false
  }
]}
```

## /publish/CacheStats

Statistics gathered for each cache.

## GET

**Response Type** Object

## Request Structure

Table 74: Request Query Parameters

Parameter	Type	Description
hc	integer	The history count, number of items to display.
stats	string	A comma separated list of stats to display.
wildcard	boolean	Controls whether specified stats should be treated as partial strings.

### 129: Example Request

```
GET /publish/CacheStats HTTP/1.1
Accept: */*
Content-Type: application/json
```

## Response Structure

**pp** Stores any provided request parameters provided as a string

**date** A ctime-like string representation of the time at which the response was served

**caches** An object with keys that are the names of monitored *cache servers*

**<server name>** Each server's object is a collection of keys that are the names of statistics

**<interface name>** The name of the network interface under the same sever

**<statistic name>** The name of the statistic which this array represents. Each value in the array is one (and usually only one) object with the following structure:

**value** The statistic's value. This is *always* a string, even if that string only contains a number.

**time** An integer UNIX timestamp indicating the start time for this value of this statistic

**span** The span of time - in milliseconds - for which this value is valid. This is deter-

mined by the polling interval for the statistic

### 130: Example Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Thu, 14 May 2020 15:48:55 GMT
Transfer-Encoding: chunked

{
  "pp": "",
  "date": "Thu May 14 15:48:55 UTC 2020",
  "caches": {
    "mid": {
      "eth0": {
        "ats.proxy.process.ssl.cipher.user_agent.PSK-AES256-
→GCM-SHA384": [
          {
            "value": "0",
            "time": 1589471325624,
            "span": 99
          }
        ]
      },
      "aggregate": {
        "ats.proxy.process.http.milestone.server_begin_write":
→[
          {
            "value": "174",
            "time": 1589471325624,
            "span": 1
          }
        ]
      },
      "lo": {
        "ats.proxy.node.http.transaction_counts_avg_10s.miss_
→changed": [
          {
            "value": "0",
            "time": 1589471325624,
            "span": 99
          }
        ]
      }
    },
    "edge": {
      "eth0": {
```

(continues on next page)

(continued from previous page)

```

    "ats.proxy.process.ssl.cipher.user_agent.PSK-AES256-
    ↪GCM-SHA384": [
        {
            "value": "0",
            "time": 1589471325624,
            "span": 99
        }
    ],
    "aggregate": {
        "ats.proxy.process.http.milestone.server_begin_write": ↪
    ↪[
        {
            "value": "174",
            "time": 1589471325624,
            "span": 1
        }
    ],
    "lo": {
        "ats.proxy.node.http.transaction_counts_avg_10s.miss_
    ↪changed": [
        {
            "value": "0",
            "time": 1589471325624,
            "span": 99
        }
    ]
    }
}

```

`publish/CacheStats/{{cache}}`

Statistics gathered for only a single cache.



## GET

**Response Type** Object

## Request Structure

Table 75: Request Path Parameters

Parameter	Type	Description
cache	string	The name of the cache to inspect

Table 76: Request Query Parameters

Parameter	Type	Description
hc	integer	The history count, number of items to display.
stats	string	A comma separated list of stats to display.
wildcard	boolean	Controls whether specified stats should be treated as partial strings.

### 131: Example Request

```
GET /api/CacheStats/mid HTTP/1.1
Accept: */*
Content-Type: application/json
```

## Response Structure

**pp** Stores any provided request parameters provided as a string

**date** A ctime-like string representation of the time at which the response was served

**caches** An object with keys that are the names of monitored *cache servers* - only the cache named by the cache request path parameter will be shown

**<server name>** The requested server's object is a collection of keys that are the names of statistics

**<interface name>** The name of the network interface under the same sever

**<statistic name>** The name of the statistic which this array represents. Each value in the array is one (and usually only one) object with the following structure:

**value** The statistic's value. This is *always* a string, even if that string only contains a number.

**time** An integer UNIX timestamp indicating the start time for this value of this statistic

**span** The span of time - in milliseconds - for which this value is valid. This is determined by the polling interval for the statistic

### 132: Example Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Thu, 14 May 2020 15:54:35 GMT
Transfer-Encoding: chunked

{
  "pp": "",
  "date": "Thu May 14 15:48:55 UTC 2020",
  "caches": {
    "mid": {
      "eth0": {
        "ats.proxy.process.ssl.cipher.user_agent.PSK-AES256-
→GCM-SHA384": [
          {
            "value": "0",
            "time": 1589471325624,
            "span": 99
          }
        ]
      },
      "aggregate": {
        "ats.proxy.process.http.milestone.server_begin_write":
→[
          {
            "value": "174",
            "time": 1589471325624,
```

(continues on next page)

(continued from previous page)

```

    "span": 1
  }
]
},
"lo": {
  "ats.proxy.node.http.transaction_counts_avg_10s.miss_
→changed": [
    {
      "value": "0",
      "time": 1589471325624,
      "span": 99
    }
  ]
}
}
}
}
}
```

**/publish/DsStats**

Statistics gathered for *Delivery Services*

**GET**

**Response Type** Object

**Request Structure**

Table 77: Request Query Parameters

Parameter	Type	Description
hc	int	The history count, number of items to display.
stats	string	A comma separated list of stats to display.
wildcard	boolean	Controls whether specified stats should be treated as partial strings.

## Response Structure

TODO

`/publish/DsStats/{{deliveryService}}`

Statistics gathered for this *Delivery Service* only.

GET

**Response Type ?**

## Request Structure

Table 78: Request Path Parameters

Parameter	Type	Description
deliveryService	string	The name of the <i>Delivery Service</i> to inspect

Table 79: Request Query Parameters

Parameter	Type	Description
hc	integer	The history count, number of items to display.
stats	string	A comma separated list of stats to display.
wildcard	boolean	Controls whether specified stats should be treated as partial strings.

## Response Structure

TODO

`/publish/CrStates`

The current state of this CDN per the *Health Protocol*.

GET

**Response Type** Object

133: Example Request

```
GET /publish/CrStates HTTP/1.1
Accept: */*
```

## Response Structure

**caches** An object with keys that are the names of monitored *cache servers*.

**isAvailable** Whether or not this *cache server* is available for routing overall

**ipv4Available** Whether or not an IPv4 interface on this *cache server* is available for routing.

**ipv6Available** Whether or not an IPv6 interface on this *cache server* is available for routing.

**status** The status of this server, along with any additional reason for it to be marked as such

**lastPoll** The last time the health data for this server was polled by a traffic monitor

**deliveryServices** An object with keys that are the *XMLIDs* of monitored *Delivery Services*.

**disabledLocations** An array of the names of disabled “locations” (i.e. *Cache Groups*) for this *Delivery Service*.

**isAvailable** Whether or not this *Delivery Service* is available for routing

134: Example Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Thu, 14 May 2020 15:54:35 GMT
Transfer-Encoding: chunked

{
  "caches": {
    "edge": {
      "isAvailable": true,
      "ipv4Available": true,
      "ipv6Available": false,
      "status": "REPORTED - available",
```

(continues on next page)

(continued from previous page)

```

        "lastPoll": "2022-03-15T17:54:03.821178179Z"
      }
    },
    "deliveryServices": {
      "dev-ds": {
        "disabledLocations": [],
        "isAvailable": true
      }
    }
  }
}

```

## /publish/CrConfig

The CDN *Snapshot* (historically named a “CRConfig”) served to and consumed by Traffic Router.

**GET**

**Response Type ?**

## Response Structure

TODO

## /publish/PeerStates

The health state information from all peer Traffic Monitors.

**GET**

**Response Type ?**

## Request Structure

Table 80: Request Query Parameters

Parameter	Type	Description
hc	integer	The history count, number of items to display.
stats	string	A comma separated list of stats to display.
wildcard	boolean	Controls whether specified stats should be treated as partial strings.

## Response Structure

TODO

### `/publish/DistributedPeerStates`

The health state information from all distributed peer Traffic Monitors.

**GET**

**Response Type ?**

## Request Structure

Table 81: Request Query Parameters

Parameter	Type	Description
<code>hc</code>	integer	The history count, number of items to display.
<code>stats</code>	string	A comma separated list of stats to display.
<code>wildcard</code>	boolean	Controls whether specified stats should be treated as partial strings.

## Response Structure

TODO

### `/publish/Stats`

The general statistics about Traffic Monitor.

**GET**

**Response Type ?**

## Response Structure

TODO

### `/publish/StatSummary`

The summary of *cache server* statistics.

**GET**

**Response Type ?**

## Request Structure

Table 82: Request Query Parameters

Parameter	Type	Description
<code>startTime</code>	number	Window start. The number of milliseconds since the epoch.
<code>endTime</code>	number	Window end. The number of milliseconds since the epoch.
<code>hc</code>	integer	The history count, number of items to display.
<code>stats</code>	string	A comma separated list of stats to display.
<code>wildcard</code>	boolean	Controls whether specified stats should be treated as partial strings.
<code>cache</code>	string	Summary statistics for just this cache.

## Response Structure

TODO

### `/publish/ConfigDoc`

The overview of configuration options.

**GET**

**Response Type ?**



## Response Structure

TODO

### 4.1.8 Traffic Ops

At one point, Traffic Ops was a collection of Perl scripts, and while the current program is written in Go, many of its tools and utilities are still written in Perl.

## Introduction

Traffic Ops at its core is mainly a PostgreSQL database used to store configuration information for ATC, and a set of RESTful API endpoints for interacting with and manipulating that information. It also serves as the single point of authentication for ATC components (that is, when one hears “user” in an ATC context it nearly always means a “user” as configured in Traffic Ops) and provides interfaces to other ATC components by proxy. Additionally, there is some miscellaneous, at times obscure functionality to Traffic Ops, such as generating arbitrary Linux system images.

## Software Requirements

Traffic Ops is only supported on CentOS 7+ systems (although many developers do use Mac OS with some success). Here are the requirements:

- [PostgreSQL 13.2](#) - the machine where Traffic Ops is running must have the client tool set (e.g. `psql(1)`), but the actual database can be run anywhere so long as it is accessible.

---

**Note:** Prior to version 13.2, Traffic Ops used version 9.6. For upgrading an existing Mac OS Homebrew-based PostgreSQL instance, you can use [Homebrew](#) to easily upgrade from 9.6 to 13.2:

```
brew services stop postgresql
brew upgrade postgresql
brew postgresql-upgrade-database
brew cleanup postgresql@9.6
brew services start postgresql
```

---

- `openssl(1SSL)` is recommended to generate server certificates, though not strictly required if certificates can be obtained by other means.
- Some kind of SMTP server is required for certain [Traffic Ops API](#) endpoints to work, but for purposes unrelated to them an SMTP server is not required. [CDN in a Box](#) comes with a relayless SMTP server for testing (you can view the emails that Traffic Ops sends, but they aren’t sent anywhere outside CDN-in-a-Box).

**Tip:** Alternatively, development and testing can be done using *CDN in a Box* - albeit somewhat more slowly.

---

### Perl Script Requirements

Not much code is still in Perl, but for the scripts the following are needed:

- Perl 5.10.1
- libpcap and libpcap development library - usually libpcap-dev or libpcap-devel in your system's native package manager.
- libpq and libpq development library - usually libpq-dev or libpq-devel in your system's native package manager.
- The [JSON Perl pod from CPAN](#)
- The [JSON::PP Perl pod from CPAN](#)
- Developers should use [Perltidy](#) to format their Perl code.

135: Example Perltidy Configuration (usually in *HOME/.perltidycrc*)

```
-l=156
-et=4
-t
-ci=4
-st
-se
-vt=0
-cti=0
-pt=1
-bt=1
-sbt=1
-bbt=1
-nsfs
-nolq
-otr
-aws
-wls="= + - / * ."
-wrs="\ "= + - / * .\"
-wbb="% + - * / x != == >= <= =~ < > | & **= += *= &= <
→<= &&= -= /= |= + >>= ||= .= %= ^= x="
```

## Go Program Requirements

- Go **1.18** or later
- If the system's Go compiler doesn't provide it implicitly, also note that all Go code in the ATC repository should be formatted using `gofmt`

All Go code dependencies are managed through the `go.mod`, `go.sum`, and `vendor/modules.txt` files. With the exception of `golang.org/x` packages (see *below*), module dependencies in `vendor/` are tracked in Git and should thus be available without any extra work - and any new dependencies should be properly “vendored” into that same, top-level directory. No other `vendor/` directories exist, as Go modules only supports a single vendor directory.

Per the Go language standard's authoritative source's recommendation, all sub-packages of `golang.org/x` are treated as a part of the compiler, and so need not ever be “vendored” as though they were an external dependency. These dependencies are not listed explicitly here, so it is strongly advised that they be fetched using `go-get(1)` rather than downloaded by hand.

---

**Tip:** All new dependencies need to be subject to community review to ensure necessity (because it will be added in its entirety to the repository, after all) and license compliance via [the developer mailing list](#).

---

## Traffic Ops Project Tree Overview

- `traffic_ops/` - The root of the Traffic Ops project
  - `app/` - Holds most of the Perl code base, though many of the files contained herein are also used by the Go implementation

---

**Note:** This directory is home to many things that no longer work as intended or have been superseded by other things.

---

- \* `bin/` - Directory for scripts and tools, `cron(8)` jobs, etc.
  - `checks/` - Contains the *Check Extensions* scripts that are provided by default
  - `db/` - Contains scripts that manipulate the database beyond the scope of setup, migration, and seeding
  - `tests/` - Integration and unit test scripts for automation purposes - in general this has been superseded by `traffic_ops/testing/api/`
- \* `conf/` - Aggregated configuration for Traffic Ops. For convenience, different environments for the `app/db/admin` tool are already set up

- development/ - Configuration files for the “development” environment
- integration/ - Configuration files for the “integration” environment
- misc/ - Miscellaneous configuration files.
- production/ - Configuration files for the “production” environment
- test/ - Configuration files for the “test” environment
- \* db/ - Database setup, seeding, and upgrade/downgrade helpers
  - migrations/ - Database migration files
  - tools/ - Contains helper scripts for easing upgrade transitions when selective data manipulation must be done to achieve a desirable state
- \* script/ - Mojolicious bootstrap/startup scripts.
- \* templates/ - Mojolicious Embedded Perl (*template name.ep*) files for the now-removed Traffic Ops UI
- build/ - Contains files that are responsible for packaging Traffic Ops into an RPM file - and also for doing the same with *ORT*
- etc/ - Configuration files for various systems associated with running production instances of Traffic Ops, which are installed under */etc* by the Traffic Ops RPM
  - \* cron.d/ - Holds specifications for *cron(8)* jobs that need to be run periodically on Traffic Ops servers

---

**Note:** At least one of these jobs expects itself to be run on a server that has the Perl implementation of Traffic Ops installed under */opt/traffic\_ops/*. Nothing terrible will happen if that’s not true, just that it/they won’t work. Installation using the RPM will set up all of these kinds of things up automatically.

---

- \* init.d/ - Contains the old, initscripts-based job control for Traffic Ops
- \* logrotate.d/ - Specifications for the Linux *logrotate(8)* utility for Traffic Ops log files
- \* profile.d/traffic\_ops.sh - Sets up common environment variables for working with Traffic Ops
- install/ - Contains all of the resources necessary for a full install of Traffic Ops
  - \* bin/ - Binaries related to installing Traffic Ops, as well as installing its prerequisites, certificates, and database

- \* `data/` - Contains things that need to be accessible by the running server for certain functionality - typically installed to `/var/www/data` by the RPM (hence the name).
- \* `etc/` - This directory left empty; it's used to contain post-installation extensions and resources
- \* `lib/` - Contains libraries used by the various installation binaries
- `ort/` - Contains *ORT* and ATS configuration file-generation logic and tooling
- `testing/` - Holds utilities for testing the *Traffic Ops API*
  - \* `api/` - Integration testing for the *Traffic Ops Go client* and Traffic Ops
- `traffic_ops_golang/` - The root of the Go implementation's code-base

---

**Note:** The vast majority of subdirectories of `traffic_ops/traffic_ops_golang/` contain handlers for the *Traffic Ops API*, and are named according to the endpoint they handle. What follows is a list of subdirectories of interest that have a special role (i.e. don't handle a *Traffic Ops API* endpoint).

---

**See also:**

The GoDoc documentation for this package

- \* `api/` - A library for use by *Traffic Ops API* handlers that provides helpful utilities for common tasks like obtaining a database transaction handle or accessing Traffic Ops configuration
- \* `auth/` - Contains definitions of privilege levels and access control code used in routing and provides a library for dealing with password and token-based authentication
- \* `config/` - Defines configuration structures and methods for reading them in from files
- \* `dbhelpers/` - Assorted utilities that provide functionality for common database tasks, e.g. "Get a user by email"
- \* `plugin/` - The Traffic Ops plugin system, with examples
- \* `trafficvault/` - This package provides the Traffic Vault interface and associated backend implementations for other handlers to interact with Traffic Vault.
- \* `routing/` - Contains logic for mapping all of the *Traffic Ops API* endpoints to their handlers, as well as proxying requests back to the Perl implementation and managing plugins, and also provides some wrappers around registered handlers that set common HTTP headers and connection options

- \* swaggerdocs/ A currently abandoned attempt at defining the *Traffic Ops API* using *Swagger* - it may be picked up again at some point in the (distant) future
  - \* tenant/ - Contains utilities for dealing with *Tenantable* resources, particularly for checking for permissions
  - \* tocookie/ - Defines the method of generating the *mojolicious* cookie used by Traffic Ops for authentication
  - \* vendor/ - contains “vendored” Go packages from third party sources
- v3-client - The official Traffic Ops Go client package for working with the version 3 *Traffic Ops API*.
  - v4-client - The official Traffic Ops Go client package for working with the version 4 *Traffic Ops API*.
  - vendor/ - contains “vendored” Go packages from third party sources

### app/db/admin

The **app/db/admin** binary is for use in managing the Traffic Ops (and Traffic Vault PostgreSQL backend) database tables. This essentially serves as a front-end for *Migrate*, though *dbconf.yml* comes from *Goose*, which Traffic Ops used to use before switching to *Migrate*.

---

**Note:** For proper resolution of configuration and SQL statement files, it’s recommended that this binary be run from the *app* directory

---

### Usage

db/admin [options] command

### Options and Arguments

#### --env ENVIRONMENT

An optional environment specification that causes the database configuration to be read out of the corresponding section of the *app/db/dbconf.yml* configuration file. One of:

- development
- integration
- production
- test

**admin** sets *MOJO\_MODE* to the value of the environment as specified by this option. (Default: development)

## **--trafficvault**

When used, commands will be run against the Traffic Vault PostgreSQL backend database as specified in the `app/db/trafficvault/dbconf.yml` configuration file.

## **command**

The *command* specifies the operation to be performed on the database. It must be one of:

**createdb** Creates the database for the current environment

**create\_migration** Creates a pair of timestamped up/down migrations titled NAME.

**create\_user** Creates the user defined for the current environment

**dbversion** Displays the database version that results from the current sequence of migrations

**down** Rolls back a single migration from the current version

**drop** Drops the database for the current environment

**drop\_user** Drops the user defined for the current environment

**load\_schema** Sets up the database for the current environment according to the SQL statements in `app/db/create_tables.sql` or `app/db/trafficvault/create_tables.sql` if the `--trafficvault` option is used

**migrate** Runs a migration on the database for the current environment

**patch** Patches the database for the current environment using the SQL statements from the `app/db/patches.sql`. This command is not supported when using the `--trafficvault` option

**redo** Rolls back the most recently applied migration, then run it again

**reset** Creates the user defined for the current environment, drops the database for the current environment, creates a new one, loads the schema into it, and runs a single migration on it

**seed** Executes the SQL statements from the `app/db/seeds.sql` file for loading static data. This command is not supported when using the `--trafficvault` option

**show\_users** Displays a list of all users registered with the PostgreSQL server

**status** Deprecated, `status` is now an alias for `dbversion` and will be removed in a future Traffic Control release.

**upgrade** Performs a migration on the database for the current environment, then seeds it and patches it using the SQL statements from the `app/db/patches.sql` file

### 136: Example Usage

```
db/admin --env=test reset
```

The environments are defined in the `traffic_ops/app/db/dbconf.yml` file, and the name of the database generated will be the name of the environment for which it was created. If the `--trafficvault` option is used, the `app/db/trafficvault/dbconf.yml` file defines this information.

## Resolving Migration Failures

If you encounter an error running a migration, you will see a message like

### 137: db/admin error example

```
[root@trafficops app]# db/admin -env production migrate
Error running migrate up: migration failed: syntax error at or near
↳ "This_is_a_syntax_error" (column 1) in line 18: /*
```

That means that the migration timestamp in the version column of the `schema_migrations` table has been updated to the version of the migration that failed, but the `dirty` column is also set, and if you try to run another migration (either up or down), you will see

### 138: db/admin error migrating when the database version is dirty

```
[root@trafficops app]# db/admin -env production migrate
Error running migrate up: Dirty database version 20210708000000000. Fix
↳ and force version.
```

You will need to manually fix the database. When you are sure that it is fixed, you can unset the `dirty` column manually using an SQL client.

## Installing The Developer Environment

To install the Traffic Ops Developer environment:

1. Clone the [Traffic Control repository](#) from GitHub. In most cases it is best to clone this directly into `GOPATH/src/github.com/apache/trafficcontrol`, as otherwise the Go implementation will not function properly.
2. Install any required Go dependencies - the suggested method is using `go-get(1)`.

### 139: Install Go Development Dependencies

```
# assuming current working directory is the repository
↳ root
go mod vendor -v
```



3. Set up a role (user) in PostgreSQL

**See also:**

PostgreSQL instructions on setting up a database.

4. Use the reset and upgrade *commands* of **admin** (see [app/db/admin](#) for usage) to set up the `traffic_ops` database(s) and optionally with the `--trafficvault` option to set up the `traffic_vault` database(s).
5. Run the `traffic_ops/install/bin/postinstall` script, it will prompt for information like the default user login credentials.
6. To run Traffic Ops, follow the instructions in [Running](#).

## app/db/traffic\_vault\_migrate

The `traffic_vault_migrate` tool - located at `traffic_ops/app/db/traffic_vault_migrate/traffic_vault_migrate.go` in the [Apache Traffic Control repository](#) - is used to transfer TV keys between database servers. It interfaces directly with each backend so Traffic Ops/Vault being available is not a requirement. The tool assumes that the schema for each backend is already setup as according to the [admin setup](#).

## Usage

```
traffic_vault_migrate [-cdhmr] [-e value] [-f value] [-g value] [-i value] [-l value] [-o value] [-t value]
```

### -c, --compare

Compare ‘to’ and ‘from’ backend keys. Will fetch keys from the dbs of both ‘to’ and ‘from’, sorts them by cdn/ds/version and does a deep comparison.

---

**Note:** Mutually exclusive with `-r/--dry`

---

### -d, --dump

Write keys (from ‘from’ server) to disk in the folder ‘dump’ with the unix permissions 0640.

**Warning:** This can write potentially sensitive information to disk, use with care.

---

**Note:** Mutually exclusive with `-l/--fill`

---

### -e LEVEL, --logLevel=LEVEL

Print everything at above specified log level (error|warning|info|debug|event) [info]

---

**Note:** Mutually exclusive with *-l/--logCfg*

---

- f** CFG, **--fromCfgPath**=CFG  
From server config file ["riak.json"]
- g** CFG, **--toCfgPath**=CFG  
To server config file ["pg.json"]
- h, --help**  
Displays usage information
- i** DIR, **--fill** DIR  
Insert data into *to* server with data in this directory

---

**Note:** Mutually exclusive with *-d/--dump*

---

- l** CFG, **--logCfg** CFG  
Log configuration file

---

**Note:** Mutually exclusive with *-e/--logLevel*

---

- o** TYPE, **--toType**=TYPE  
From server types (Riak|PG) [PG]
- m, --noConfirm**  
Don't require confirmation before inserting records
- r, --dry**  
Do not perform writes. Will do a basic output of the keys on the 'from' backend.

---

**Note:** Mutually exclusive with *-c/--compare*

---

- t** TYPE, **--fromType**=TYPE  
From server types (Riak|PG) [Riak]

## Riak

### riak.json

**user** The username used to log into the Riak server.

**password** The password used to log into the Riak server.

**host** The hostname for the Riak server.

**port** The port for which the Riak server is listening for protobuf connections.

**timeout** The number of seconds to wait for each operation.

**insecure** (Optional) Determines whether to verify insecure certificates.

**tlsVersion** (Optional) Max TLS version supported. Valid values are “10”, “11”, “12”, “13”.

## Postgres

**traffic\_vault\_migrate** will properly handle both encryption and decryption of postgres data as that is done on the client side.

## pg.json

**user** The username used to log into the PG server.

**password** The password for the user to log into the PG server.

**database** The database to connect to.

**port** The port on which the PG server is listening.

**host** The hostname of the PG server.

**sslmode** The ssl settings for the client connection, [explanation here](#). Options are ‘disable’, ‘allow’, ‘prefer’, ‘require’, ‘verify-ca’ and ‘verify-full’

**aesKey** The base64 encoding of a 16, 24, or 32 bit AES key.

## Logging

The log configuration file has the structure:

**error\_log** Where to output error messages (stderr|stdout|null)

**warning\_log** Where to output warning messages (stderr|stdout|null)

**info\_log** Where to output info messages (stderr|stdout|null)

**debug\_log** Where to output error messages (stderr|stdout|null)

**event\_log** Where to output error messages (stderr|stdout|null)

### Adding a Migration Backend

To add a plugin, implement the `traffic_vault_migrate.go:TVBackend` interface and add the backend to the returned values in [github.com/apache/trafficcontrol/traffic\\_ops/app/db/traffic\\_vault\\_migrate.supportBackends](https://github.com/apache/trafficcontrol/traffic_ops/app/db/traffic_vault_migrate.supportBackends).

### Test Cases

#### Go Tests

Many (but not all) endpoint handlers and utility packages in the Go code-base define Go unit tests that can be run with `go-test(1)`. There are integration tests for the Traffic Ops Go client in `traffic_ops/testing/api/`.

#### 140: Sample Run of Go Unit Tests

```
cd traffic_ops/traffic_ops_golang

# run just one test
go test ./about

# run all of the tests
go test ./...
```

There are a few prerequisites to running the Go client integration tests:

- A PostgreSQL server must be accessible and have a Traffic Ops database schema set up (though not necessarily populated with anything).
- A running Traffic Ops Go implementation instance must be accessible - it shouldn't be necessary to also be running the Perl implementation.

---

**Note:** For testing purposes, SSL certificates are not verified, so self-signed certificates will work fine.

---

---

**Note:** It is *highly* recommended that the Traffic Ops instance be run on the same machine as the integration tests, as otherwise network latency can cause the tests to exceed their threshold time limit of ten minutes.

---

The integration tests are run using `go-test(1)`, with two configuration options available.

---

**Note:** It should be noted that the integration tests will output thousands of lines of highly repetitive text not directly related to the tests its running at the time - even if the `-v` flag is not passed to `go-test(1)`. This problem is tracked by [Issue #4017](#).

---

**Warning:** Running the tests will wipe the connected database clean, so do not **ever** run it on an instance of Traffic Ops that holds meaningful data.

### **--cfg** CONFIG

Specify the path to the *Test Configuration File*. If not specified, it will attempt to read a file named `traffic-ops-test.conf` in the working directory.

#### **See also:**

*Configuring the Integration Tests* for a detailed explanation of the format of this configuration file.

### **--fixtures** FIXTURES

Specify the path to a file containing static data for the tests to use. This should almost never be used, because many of the tests depend on the data having a certain content and structure. If not specified, it will attempt to read a file named `tc-fixtures.json` in the working directory.

### **--includeSystemTests** {no|yes}

Specify whether to run tests that depend on additional components like an SMTP server or a Traffic Vault server. Default: no

## Configuring the Integration Tests

Configuration is mainly done through the configuration file passed as **--cfg**, but is also available through the following environment variables.

In addition to the variables described here, the integration tests support identifying the network location of the Traffic Ops instance via *TO\_URL*.

### **SESSION\_TIMEOUT\_IN\_SECS**

Sets the timeout of requests made to the Traffic Ops instance, in seconds.

### **TODB\_DESCRIPTION**

An utterly cosmetic variable which, if set, gives a description of the PostgreSQL database to which the tests will connect. This has no effect except possibly changing one line of debug output.

### **TODB\_HOSTNAME**

If set, will define the FQDN at which the PostgreSQL server to be used by the tests resides<sup>1</sup>.

### **TODB\_NAME**

If set, will define the name of the database to which the tests will connect<sup>Page 318, 1</sup>.

### **TODB\_PASSWORD**

If set, defines the password to use when authenticating with the PostgreSQL server.

### **TODB\_PORT**

If set, defines the port on which the PostgreSQL server listens<sup>Page 318, 1</sup>.

<sup>1</sup> The Traffic Ops instance *must* be using the same PostgreSQL database that the tests will use.

### TODB\_SSL

If set, must be one of the following values:

**true** The PostgreSQL server to which the tests will connect uses SSL on the port on which it will be contacted.

**false** The PostgreSQL server to which the tests will connect does not use SSL on the port on which it will be contacted.

### TODB\_TYPE

If set, tells the database driver used by the tests the kind of SQL database to which they are connecting<sup>Page 318, 1</sup>. This author has no idea what will happen if this is set to something other than Pg, but it's possible the tests will fail to run. Certainly never do it.

### TODB\_USER

If set, defines the user as whom to authenticate with the PostgreSQL server.

### TO\_USER\_ADMIN

If set, will define the name of a user with the “admin” *Role* that will be created by the tests<sup>2</sup>.

### TO\_USER\_DISALLOWED

If set, will define the name of a user with the “disallowed” *Role* that will be created by the tests<sup>2</sup>.

### TO\_USER\_EXTENSION

If set, will define the name of a user with the “extension” *Role* that will be created by the tests<sup>2</sup>.

**Caution:** Due to legacy constraints, the only truly safe value for this is `extension` - anything else could cause the tests to fail.

### TO\_USER\_FEDERATION

If set, will define the name of a user with the “federation” *Role* that will be created by the tests<sup>2</sup>.

### TO\_USER\_OPERATIONS

If set, will define the name of a user with the “operations” *Role* that will be created by the tests<sup>2</sup>.

### TO\_USER\_PASSWORD

If set, will define the password used by all users created by the tests. This does not need to be the password of any pre-existing user.

### TO\_USER\_PORTAL

If set, will define the name of a user with the “portal” *Role* that will be created by the tests<sup>2</sup>.

### TO\_USER\_READ\_ONLY

If set, will define the name of a user with the “read-only” *Role* that will be created by the tests<sup>2</sup>.

---

<sup>2</sup> This does not need to match the name of any pre-existing user.

## Test Configuration File

The configuration file for the tests (defined by `--cfg`) is a JSON-encoded object with the following properties.

**Warning:** Many of these configuration options are overridden by variables in the execution environment. Where this is a problem, there is an associated warning. In general, this issue is tracked by [Issue #3975](#).

**default** An object containing sub-objects relating to default configuration settings for connecting to external resources during testing

**logLocations** An object containing key/value pairs where the keys are log levels and each associated value is the file location to which logs of that level will be written. The allowed values respect the [reserved special names used by the github.com/apache/trafficcontrol/lib/go-log package](#). Omitted keys are treated as though their values were null, in which case that level is written to `/dev/null`. The allowed keys are:

- debug
- error
- event
- info
- warning

**session** An object containing key/value pairs that define the default settings used by Traffic Ops “session” connections

**timeoutInSecs** At the time of this writing this is the only meaningful configuration option that may be present under `session`. It specifies the timeouts used by client connections during testing as an integer number of seconds. The default if not specified (or overridden) is 0, meaning no limit.

**Warning:** This configuration is overridden by `SESSION_TIMEOUT_IN_SECS`.

**trafficOps** An object containing information that defines the running Traffic Ops instance to use in testing.

**password** This password will be used for all created users used by the test suite - it does not need to be the password of any pre-existing user. The default if not specified (or overridden) is an empty string, which may or may not cause problems.

**Warning:** This is overridden by `TO_USER_PASSWORD`.

**URL** The network location of the running Traffic Ops server, including schema, hostname and optionally port number e.g. `https://localhost:6443`.

**Warning:** This is overridden by `TO_URL`.

**users** An object containing key-value pairs where the keys are the names of *Roles* and the values are the usernames of users that will be created with the associated *Role* for testing purposes. *There are very few good reasons why the values should not just be the same as the keys.* The default for any missing (and not overridden) key is the empty string which is *won't* work so please don't leave any undefined. The allowed keys are:

- admin

**Warning:** The value of this key is overridden by `TO_USER_ADMIN`.

- disallowed

**Warning:** The value of this key is overridden by `TO_USER_DISALLOWED`.

- extension

**Warning:** The value of this key is overridden by `TO_USER_EXTENSION`.

- federation



**Warning:** The value of this key is overridden by *TO\_USER\_FEDERATION*.

- operations

**Warning:** The value of this key is overridden by *TO\_USER\_OPERATIONS*.

- portal

**Warning:** The value of this key is overridden by *TO\_USER\_PORTAL*.

- readOnly

**Warning:** The value of this key is overridden by *TO\_USER\_READ\_ONLY*.

**trafficOpsDB** An object containing information that defines the database to use in testing<sup>Page 318, 1</sup>.

**dbname** The name of the database to which the tests will connect<sup>Page 318, 1</sup>.

**Warning:** This is overridden by *TODB\_NAME*.

**description** An utterly cosmetic option that need not exist at all which, if set, gives a description of the database to which the tests will connect. This has no effect except possibly changing one line of debug output.

**Warning:** This is overridden by *TODB\_DESCRIPTION*

**hostname** The FQDN of the server on which the database is running<sup>Page 318, 1</sup>

**Warning:** This is overridden by `TODB_HOSTNAME`.

**password** The password to use when authenticating with the database

**Warning:** This is overridden by `TODB_PASSWORD`.

**port** The port on which the database listens for connections<sup>Page 318, 1</sup> - as a **string**

**Warning:** This is overridden by `TODB_PORT`.

**type** The “type” of database being used<sup>Page 318, 1</sup>. This should **never** be set to anything besides “Pg”, anything else results in undefined behavior (although it’s equally possible that it simply won’t have any effect).

**Warning:** This is overridden by `TODB_TYPE`.

**ssl** An optional boolean value that defines whether or not the database uses SSL encryption for its connections - default if not specified (or overridden) is `false`

**Warning:** This is overridden by `TODB_SSL`.

**user** The name of the user as whom to authenticate with the database

**Warning:** This is overridden by `TODB_USER`.

## Writing New Endpoints

---

**Note:** Most new endpoints are accompanied by database schema changes which necessitate a new migration under `traffic_ops/app/db/migrations` and database best-practices are not discussed in this section.

---

**See also:**

This section contains a quick overview of API endpoint development; for the full guidelines for API endpoints, consult [API Guidelines](#).

The first thing to consider when writing a new endpoint is what the requests it will serve will look like. It's recommended that new endpoints avoid using "path parameters" when possible, and instead try to utilize request bodies and/or query string parameters. For example, instead of `/foos/{ID}` consider simply `/foos` with a supported `id` query parameter. The request *methods* should be restricted to the following, and respect each method's associated meaning.

**DELETE** Removes a resource or one or more of its representations from the server. This should **always** be the method used when deleting objects.

**GET** Retrieves a representation of some resource. This should *always* be used for read-only operations and note that the requesting client **never** expects the state of the server to change as a result of a request using the GET method.

**POST** Requests that the server process some passed data. This is used most commonly to create new objects on the server, but can also be used more generally e.g. with a request for regenerating encryption keys. Although this isn't strictly creating new API resources, it does change the state of the server and so this is more appropriate than GET.

**PUT** Places a new representation of some resource on the server. This is typically used for updating existing objects. For creating *new* representations/objects, use POST instead. When using PUT note that clients expect it to be *idempotent*, meaning that subsequent identical PUT requests should result in the same server state. What this means is that it's standard to require that *all* of the information defining a resource be provided for each request even if the vast majority of it isn't changing.

The HEAD and OPTIONS request methods have default implementations for any properly defined [Traffic Ops API](#) route, and so should almost never be defined explicitly. Other request methods (e.g. CONNECT) are currently unused and ought to stay that way for the time being.

---

**Note:** Utilizing the PATCH method is unfeasible at the time of this writing but progress toward supporting it is being made, albeit slowly in the face of other priorities.

---

**See also:**

The MDN (Mozilla Developer Network)'s [documentation on the various HTTP request methods](#).

The final step of creating any [Traffic Ops API](#) endpoint is to write documentation for it. When doing so, be sure to follow *all* of the guidelines laid out in [Documentation Guidelines](#). *If documentation doesn't exist for new functionality then it has accomplished nothing because no one using Traffic Control will know it exists.* Omitted documentation is how a project winds up with a dozen different API endpoints that all do essentially the same thing.

### Framework Options

The Traffic Ops code base offers two basic frameworks for defining a new endpoint. Either one may be used at the author's discretion (or even neither if desired and appropriate - though that seems unlikely).

#### Generic “CRUDer”

The “Generic ‘CRUDer’”, as it's known, is a pattern of API endpoint development that principally involves defining a type that implements the [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.CRUDEr](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.CRUDEr) interface. A description of what that entails is best left to the actual GoDoc documentation.

#### See also:

The [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.GenericCreate](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.GenericCreate), [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.GenericDelete](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.GenericDelete), [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.GenericRead](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.GenericRead), and [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.GenericUpdate](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.GenericUpdate) helpers are often used to provide the default operations of creating, deleting, reading, and updating objects, respectively. When the API endpoint being written is only meant to perform these basic operations on an object or objects stored in the database, these should be totally sufficient.

This method offers a lot of functionality “out-of-the-box” as compared to the [APIInfo](#) method, but because of that is also restrictive. For example, it is not possible to write an endpoint that returns data not encoded as JSON using this method. That's an uncommon use-case, but not unheard-of.

This method is best used for basic creation, reading, update, and deletion operations performed on simple objects with no structural differences across API versions.

#### APIInfo

Endpoint handlers can also be defined by simply implementing the [net/http.HandlerFunc](https://golang.org/pkg/net/http/#HandlerFunc) interface. The [net/http.Request](https://golang.org/pkg/net/http/#Request) reference passed into such handlers provides identifying information for the authenticated user (where applicable) in its context.

To easily obtain the information needed to identify a user and their associated permissions, as well as server configuration information and a database transaction handle, authors should use the [github.com/apache/trafficcontrol/traffic\\_ops/traffic\\_ops\\_golang/api.NewInfo](https://github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.NewInfo) function which will return all of that information in a single structure as well as any errors encountered during the process and an appropriate HTTP response code in case of such errors.

This method offers fine control over the endpoint's logic, but tends to be much more verbose than the endpoints written using the [Generic “CRUDer”](#) method. For example, a handler for retrieving an object from the database and returning it to the requesting client encoded as JSON can be twenty or more lines of code, whereas a sin-

gle call to `github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/api.GenericCreate` provides equivalent functionality.

This method is best used when requests are meant to have extensive side-effects, are performed on unusually structured objects, need fine control of the HTTP headers/options, or operate on objects that have different structures or meanings across API versions.

## Extensions

What’s typically meant by “extension” in the context of Traffic Ops is a *Check Extensions* which provides data for server “checks” which can be viewed in Traffic Portal under *Monitor* → *Cache Checks*. This type of extension need not know nor even care which implementation it is being used with, as it interacts with Traffic Ops through the *Traffic Ops API*.

Traffic Ops supports overrides or new definitions for non-standard *Traffic Ops API* routes. This type of “extension” is typically referred to as a “plugin,” and they are described in *Go Plugins*.

## Go Plugins

A plugin is defined by a Go source file in the `traffic_ops/traffic_ops_golang/plugin` directory, which is expected to be named `plugin name.go`. A plugin is registered to Traffic Ops by a call to `github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/plugin.AddPlugin` in the source file’s special `init` function.

A plugin is only enabled at runtime if its name is present in the `cdn.conf` file’s `traffic_ops_golang.plugins` array.

Each plugin may also define any, all, or none of the lifecycle hooks provided: `load`, `startup`, and `onRequest`

**load** The `load` function of a plugin, if defined, needs to implement the `github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/plugin.LoadFunc` interface, and will be run when the server starts and after configuration has been loaded. It will be passed the plugin’s own configuration as it was defined in the `cdn.conf` file’s `traffic_ops_golang.plugin_config` map.

**onRequest** The `onRequest` function of a plugin, if defined, needs to implement the `github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/plugin.OnRequestFunc` interface, and will be called on **every** request made to the *Traffic Ops API*. Because of this, it’s imperative that the function exit as soon as possible. Note that once one plugin reports that it has served the request, no others will be tried. The order in which plugins are tried is defined by their order in the `traffic_ops_golang.plugins` array of the `cdn.conf` configuration file.

### See also:

It’s very common for this function to behave like a *Traffic Ops API* endpoint, so when writing a plugin it may be useful to review *Writing New Endpoints*.

**startup** Like `load`, the `startup` function of a plugin, if defined, will be called when the server starts and after configuration has been loaded. *Unlike* `load`, however, this function should implement the `github.com/apache/trafficcontrol/traffic_ops/traffic_ops_golang/plugin.StartupFunc` interface and will be passed in the entirety of the server's configuration, including its own configuration and any shared plugin configuration data as defined in the `cdn.conf` file's `traffic_ops_golang.plugin_shared_config` map.

## Example

An example “Hello World” plugin that serves the `/_hello` request path by just writing “Hello World” in the body of a 200 OK response back to the client is provided in `traffic_ops/traffic_ops_golang/plugin/hello_world.go`:

```

1 package plugin
2
3 /*
4  Licensed under the Apache License, Version 2.0 (the "License");
5  you may not use this file except in compliance with the License.
6  You may obtain a copy of the License at
7
8  http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the License is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
13 → implied.
14 See the License for the specific language governing permissions and
15 limitations under the License.
16 */
17 import (
18     "strings"
19 )
20
21 func init() {
22     AddPlugin(10000, Funcs{onRequest: hello}, "example HTTP plugin",
23     → "1.0.0")
24 }
25
26 const HelloPath = "/_hello"
27
28 func hello(d OnRequestData) IsRequestHandled {
29     if !strings.HasPrefix(d.R.URL.Path, HelloPath) {
30         return RequestUnhandled
31     }
32     d.W.Header().Set("Content-Type", "text/plain")

```

(continues on next page)

(continued from previous page)

```

32     d.W.Write([]byte("Hello, World!"))
33     return RequestHandled
34 }

```

## Check Extensions

*Check Extensions* allow you to add custom checks to the *Monitor* → *Cache Checks* view.

Extensions are managed using the `$TO_HOME/bin/extensions` command line script

### See also:

For more information see *Managing Traffic Ops Extensions*.

## 4.1.9 Traffic Portal

### Introduction

Traffic Portal is an [AngularJS](#) client served from a [Node.js](#) web server designed to consume the *Traffic Ops API*. Traffic Portal is the official replacement for the legacy Traffic Ops UI.

### Software Requirements

To work on Traffic Portal you need a \*nix (MacOS and Linux are most commonly used) environment that has the following installed:

- [Node.js 12.0.x](#) or above
- [Grunt CLI 1.2.0](#) or above
- Access to a working instance of Traffic Ops

### Install Global NPM Packages

Grunt CLI can be installed using NPM.

## 141: Install Grunt CLI

```
npm -g install grunt-cli
```

## Traffic Portal Project Tree Overview

- **traffic\_control/traffic\_portal/app/src** - contains HTML, JavaScript and SCSS (Sassy CSS) source files.

## Installing The Traffic Portal Developer Environment

1. Clone the [Traffic Control Repository](#)
2. Navigate to the `traffic_portal` subdirectory of your cloned repository.
3. Run `npm install` to install application dependencies into `traffic_portal/node_modules`. Only needs to be done the first time unless `traffic_portal/package.json` changes.
4. Modify `traffic_portal/conf/configDev.js`:
  1. Valid SSL certificates and keys are needed for Traffic Portal to run. Generate these (e.g. using [this SuperUser answer](#)) and update `ssl`.
  2. Modify `api.base_url` to point to your Traffic Ops API endpoint.
5. Run `grunt` to package the application into `traffic_portal/app/dist`, start a local HTTPS server (Express), and start a file watcher. To use a custom configuration file (not just `traffic_portal/conf/config.js` or `traffic_portal/conf/configDev.js`), set the `TP_SERVER_CONFIG_FILE` environment variable to the location of the desired file.
6. Navigate to `http(s)://localhost:[port|sslPort]` defined in the configuration file used (default: `traffic_portal/conf/configDev.js`)

## 4.1.10 Traffic Router

### Introduction

Traffic Router is a Java Tomcat application that routes clients to the closest available cache on the CDN using both HTTP and DNS. Cache server availability is determined by Traffic Monitor; consequently Traffic Router polls Traffic Monitor for its configuration and *cache server* health state information, and uses this data to make routing decisions. HTTP routing is performed by localizing the client based on the request's source IP address (IPv4 or IPv6), and issues an HTTP 302 response to redirect to the nearest *cache server*. HTTP routing utilizes consistent hashing on request URLs to optimize cache performance and request distribution. DNS routing is performed by localizing clients, resolves in most cases, requesting A and AAAA records for



a configurable name such as `foo.deliveryservice.somecdn.net`. Traffic Router is comprised of seven separate Maven modules:

- `shared` - A reusable utility JAR for defining *Delivery Service* Certificates
- `configuration` - A reusable JAR defining the `ConfigurationListener` interface
- `connector` - A JAR that overrides Tomcat's standard `Http11Protocol` Connector class and allows Traffic Router to delay opening listen sockets until it is in a state suitable for routing traffic
- `geolocation` - Submodule for defining geolocation services
- `neustar` - A JAR that provides a bean "neustarGeolocationService" that implements the `GeolocationService` interface defined in the `geolocation` maven submodule, which can optionally be added to the build of Traffic Router
- `core` - Services DNS and HTTP requests, performs localization on routing requests, and is deployed as a WAR to a Service (read: connector/listen port) within Tomcat which is separate from the API
- `build` - A simple Maven project which gathers the artifacts from the modules and builds an RPM

## Software Requirements

To work on Traffic Router you need a \*nix (MacOS and Linux are most commonly used) environment that has the following installed:

- Eclipse >= Kepler SR2 (or another Java IDE)
- Maven >= 3.3.1
- JDK >= 11 (OpenJDK suggested, but not required)
- OpenSSL >= 1.0.2
- APR (Apache Portable Runtime) >= 1.4.8-3
- Tomcat Native >= 1.2.23
- Not Tomcat - You do not need a Tomcat installation for development. An embedded version is launched for development testing instead.

## Get OpenJDK 11 on macOS

Using Homebrew:

142: Install OpenJDK 11 on macOS

```
brew install openjdk@11
```

Next, set the `JAVA_HOME` environment variable. Add this line to your `~/.bash_profile`:

143: Set JAVA\_HOME environment variable

```
export JAVA_HOME=$(/usr/libexec/java_home -v11)
```

## Traffic Router Project Tree Overview

- `traffic_control/traffic_traffic_router/` - base directory for Traffic Router
  - `connector/` - Source code for Traffic Router Connector;
    - \* `src/main/java` - Java source directory for Traffic Router Connector
  - `core/` - Source code for Traffic Router Core, which is built as its own deployable WAR file and communicates with *Traffic Router API* using JMX
    - \* `src/main` - Main source directory for Traffic Router Core
      - `lib/systemd/system/traffic_router.service` - Unit script for launching the Traffic Router with Tomcat
      - `conf/` - All of the required configuration files for running the `traffic_router` web application, including those needed for Tomcat
      - `java/` - Java source code for Traffic Router Core
      - `resources/` - Resources pulled in during an RPM build
      - `scripts/` - Scripts used by the RPM build process
      - `webapp/` - Java “webapp” resources
      - `var/log/` - location of all the Traffic Router runtime logs
    - \* `src/test` - Test source directory for Traffic Router Core
      - `conf/` - Minimal Configuration files that make it possible to run JUnit tests
      - `db/` - Files downloaded by unit tests
      - `java/` - JUnit-based unit tests for Traffic Router Core
      - `resources/` - Example data files used by junit tests
      - `var/auto-zones` - BIND formatted zone files generated by Traffic Router Core during unit testing

## Java Formatting Conventions

None at this time. The codebase will eventually be formatted per Java standards.

## Installing The Developer Environment

To install the Traffic Router Developer environment:

1. Clone the `traffic_control` repository using Git.
2. Change directories into `traffic_control/traffic_router`.
3. Set the environment variable `TRAFFIC_MONITOR_HOSTS` to be a semicolon delimited list of Traffic Monitors that can be accessed during integration tests OR install the `traffic_monitor.properties` file.
4. Additional configuration is set using the below files:
  - copy `core/src/main/conf/dns.properties` to `core/src/test/conf/`
  - copy `core/src/main/conf/http.properties` to `core/src/test/conf/`
  - copy `core/src/main/conf/log4j2.xml` to `core/src/test/conf/`
  - copy `core/src/main/conf/traffic_monitor.properties` to `core/src/test/conf/` and then edit the `traffic_monitor.bootstrap.hosts` property
  - copy `core/src/main/conf/traffic_ops.properties` to `core/src/test/conf/` and then edit the credentials as appropriate for the Traffic Ops instance you will be using.
  - Default configuration values now reside in `core/src/main/webapp/WEB-INF/applicationContext.xml`

---

**Note:** These values may be overridden by creating and/or modifying the property files listed in `core/src/main/resources/applicationProperties.xml`

---

---

**Note:** Pre-existing properties files are still honored by Traffic Router. For example `traffic_monitor.properties` may contain the FQDN and port of the Traffic Monitor instance(s), separated by semicolons as necessary (do not include scheme e.g. `http://`)

---

5. Import the existing git repository as projects into your IDE (Eclipse):
  - a. *File* → *Import* → *Git* → *Projects from Git; Next*

- b. *Existing local repository; Next*
  - c. *Add* - browse to find `traffic_control`; *Open*
  - d. Select `traffic_control`; *Next*
  - e. Ensure *Import existing projects* is selected, expand `traffic_control`, select `traffic_router`; *Next*
  - f. Ensure `traffic_router_api`, `traffic_router_connector`, and `traffic_router_core` are checked; *Finish* (this step can take several minutes to complete)
  - g. Ensure `traffic_router_api`, `traffic_router_connector`, and `traffic_router_core` have been opened by Eclipse after importing
6. From the terminal or your IDE, run `mvn clean verify` from the `traffic_router` directory. This will run a series of integration tests and will temporarily start and embedded version of Traffic Router and a ‘fake’ simulated instance of Traffic Monitor.
7. Start the embedded Tomcat instance for Core from within your IDE by following these steps:
  - a. In the package explorer, expand `traffic_router_core`
  - b. Expand `src/test/java`
  - c. Expand the package `org.apache.traffic_control.traffic_router.core`
  - d. Open and run `TrafficRouterStart.java`

---

**Note:** If an error is displayed in the Console, run `mvn clean verify` from the `traffic_router` directory

---

Once running, the *Traffic Router API* is available over HTTP at <http://localhost:3333> and over HTTPS at <https://localhost:3443>, the HTTP routing interface is available on <http://localhost:8888> and HTTPS is available on <https://localhost:8443>. The DNS server and routing interface is available on `localhost:53` via TCP and UDP.

### Development Environment Upgrade

If a development environment is already set up for the previous version of Traffic Router, then `openssl`, `apr` and `tomcat-native` will need to be manually installed with `yum(8)` or `rpm(8)`. Also, whenever either `mvn clean verify` or `TrafficRouterStart` is/are run, the location of the `tomcat-native` libraries will need to be made known to the JVM via command line arguments.

144: Example Commands Specifying a Path to the tomcat-native Library

```
mvn clean verify -Djava.library.path=[tomcat native library path on
→your box]
java -Djava.library.path=[tomcat native library path on your box]
→TrafficRouterStart
```

## Manual Testing

Look up the URL for a test HTTP *Delivery Service* in Traffic Ops and then make a request. When Traffic Router is running and used as a resolver for the host in the *Delivery Service* URL, the requested origin content should be found through an Edge-tier *cache server*.

145: Example Test for an HTTP *Delivery Service*

```
root@enroller:/shared/enroller# curl -skvL http://video.demo1.mycdn.
→ciab.test/
* Trying fc01:9400:1000:8::60...
* TCP_NODELAY set
* Connected to video.demo1.mycdn.ciab.test (fc01:9400:1000:8::60) port
→80 (#0)
> GET / HTTP/1.1
> Host: video.demo1.mycdn.ciab.test
> User-Agent: curl/7.52.1
> Accept: */*
>
< HTTP/1.1 302 Found
< Location: http://edge.demo1.mycdn.ciab.test/
< Content-Length: 0
< Date: Wed, 16 Jan 2019 21:52:14 GMT
<
* Curl_http_done: called premature == 0
* Connection #0 to host video.demo1.mycdn.ciab.test left intact
* Issue another request to this URL: 'http://edge.demo1.mycdn.ciab.
→test/'
* Trying fc01:9400:1000:8::100...
* TCP_NODELAY set
* Connected to edge.demo1.mycdn.ciab.test (fc01:9400:1000:8::100) port
→80 (#1)
> GET / HTTP/1.1
> Host: edge.demo1.mycdn.ciab.test
> User-Agent: curl/7.52.1
> Accept: */*
>
< HTTP/1.1 200 OK
< Content-Type: text/html
```

(continues on next page)

(continued from previous page)

```
< Accept-Ranges: bytes
< ETag: "1473249267"
< Last-Modified: Wed, 07 Nov 2018 13:53:57 GMT
< Cache-Control: public, max-age=300
< Access-Control-Allow-Origin: *
< Access-Control-Allow-Headers: Accept, Origin, Content-Type
< Access-Control-Allow-Methods: GET, POST, PUT, OPTIONS
< Content-Length: 1881
< Date: Wed, 16 Jan 2019 21:52:15 GMT
< Server: ATS/7.1.4
< Age: 1
< Via: http/1.1 mid.infra.ciab.test (ApacheTrafficServer/7.1.4
→[uScMsSfWpSeN:t cCMi p sS]), http/1.1 edge.infra.ciab.test
→(ApacheTrafficServer/7.1.4 [uScMsSfWpSeN:t cCMi pSs ])
< Connection: keep-alive
<
<!DOCTYPE html>
<!-- Licensed to the Apache Software Foundation (ASF) under one
or more contributor license agreements. See the NOTICE file
distributed with this work for additional information
regarding copyright ownership. The ASF licenses this file
to you under the Apache License, Version 2.0 (the
"License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing,
software distributed under the License is distributed on an
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
KIND, either express or implied. See the License for the
specific language governing permissions and limitations
under the License. -->
<html lang="en">
<head>
    <title>CDN In a Box</title>
    <meta charset="utf-8"/>
    <meta charset="utf-8"/>
    <meta name="viewport" content="width=device-width; height=device-
→height; initial-scale=1"/>
    <link rel="shortcut-icon" href="/tc_logo.svg"/>
    <meta name="author" content="Apache"/>
    <meta name="creator" content="Apache"/>
    <meta name="publisher" content="Apache"/>
    <meta name="description" content="A simple test origin for Apache
→Traffic Control"/>
```

(continues on next page)

(continued from previous page)

```
<style type="text/css">
  html {
    height: 100vh;
    width: 100vw;
  }

  body {
    text-align: center;
    background-image: url(/tc_logo.svg);
    background-color: black;
    background-position: center;
    background-repeat: no-repeat;
    background-size: 25%;
    font-family: "Ubuntu Mono","Consolas",sans-serif;
    color: white;
    margin: 0;
    padding-top: 0.67em;
    max-width: 100%;
  }

  h1 {
    margin-top: 0.67em;
  }

  p {
    text-align: left;
    width: 80vw;
    min-width: 320px;
    margin: auto;
  }
</style>
</head>
<body>
  <h1>Test Origin</h1>
  <p>This is a test "origin" server for Apache Traffic Control</p>
</body>
</html>
* Curl_http_done: called premature == 0
* Connection #1 to host edge.demo1.mycdn.ciab.test left intact
```

### Test Cases

- Unit tests can be executed using Maven by running `mvn test` at the root of the `traffic_router` project.
- Unit and Integration tests can be executed using Maven by running `mvn verify` at the root of the `traffic_router` project.

### Debugging Unit Tests

In order to write tests or understand why a test is failing, a developer may want to debug the unit tests. In order to stop at breakpoints, you should run the tests without forking processes. Even once you have specified for Surefire not to fork tests, the debugging connection will disconnect several times while the tests are running, so you should - Have JVM act as the debugging client and have your IDE act as the debugging server, even if you are remote debugging - In your IDE's debugging configuration for the Traffic Router unit tests, enable “auto restart”, meaning that, once the debugging client running on JVM has disconnected because a particular test is over, your IDE's debugging server will exit (unavoidable) but will immediately restart in time for the next test.

---

**Note:** If you run the tests with debugging enabled and with JVM acting as the debugging client but your IDE is not actively listening for debugging connections, the unit tests *will* fail.

---

Command for running the tests with debugging enabled:

146: Run the Traffic Router unit tests with debugging enabled

```
mvn '-Dmaven.surefire.debug=-agentlib:jdwp=transport=dt_socket,
→server=n,suspend=n,address=localhost:8000 -DforkCount=0 -
→DreuseForks=false' test -Djava.library.path=/usr/share/java
```

### Debugging Unit Tests in Docker

In order to run the unit tests in a controlled, well-defined environment, you may prefer to run them from within Docker. A Docker environment for running the Traffic Router unit tests exists in the repository at `/traffic_router/tests`, and it supports debugging. In order to enable debugging, set `DEBUG_ENABLE` to `'true'` in `docker-compose.yml`. As mentioned in [Debugging Unit Tests](#), - Have your IDE act as the debugging server. In IntelliJ, this debug mode is called *Listen to remote JVM*. - Enable *auto-restart* in your IDE's debugging configuration for the Traffic Router unit tests so your IDE doesn't stop listening for connections after the first test ends - Set the port to 8000 (debugging port is specified in the Dockerfile)

Additionally, you will need to make sure that `host.docker.internal` resolves to the Docker host's IP address (NOT the Docker container's IP address). If you are using Docker Desktop for Mac or Docker Desktop for Windows, this is already set up for you. If you are on Linux, you will need



to either figure out how to make `host.docker.internal` resolve to the `docker0` network interface's IP address, or, in `docker-compose.yml`, change the value of the `DEBUG_HOST` environment variable to the IP address of the `docker0` interface.

---

**Note:** If you run the tests with debugging enabled and with JVM (in Docker) acting as the debugging client but your IDE is not actively listening for debugging connections, the unit tests *will* fail.

---

Once your IDE is listening for debugging connections, start the unit tests:

147: Run the Traffic Router unit tests in Docker, with or without debugging enabled

```
docker-compose up
```

## RPM Packaging

Running `mvn package` on a Linux-based distribution will trigger the build process to create the Traffic Router RPM and the Traffic Router `.war` file, but will not run the integration tests, so it is a good way to update those artifacts quickly during development. But the preferred way to build the Traffic Router RPMs is by following the instructions in [Building Traffic Control](#)

## API

### *Traffic Router API*

## Traffic Router API

By default, Traffic Router serves its API via HTTP (not HTTPS) on port 3333. This can be configured in `/opt/traffic_router/conf/server.xml` or by setting a [Parameter](#) with the [Name](#) “`api.port`”, and the [Config File](#) “`server.xml`” on the Traffic Router's [Profile](#).

The API can be configured via HTTPS on port 3443 in `/opt/traffic_router/conf/server.xml` or by setting a [Parameter](#) named `secure.api.port` with `configFile server.xml` on the Traffic Router's [Profile](#). The post install script will generate self signed certificates at `/opt/traffic_router/conf/`, create a new Java Keystore named `/opt/traffic_router/conf/keyStore.jks`, and add the new certificate to the Keystore. The password for the Java Keystore and the Keystore location are stored in `/opt/traffic_router/conf/https.properties`. To override the self signed certificates with new ones from a certificate authority, update the properties for the Keystore location and password at `/opt/traffic_router/conf/https.properties`.

The API can be configured via HTTPS on port 3443 in `/opt/traffic_router/conf/server.xml` or by setting a [Parameter](#) named `secure.api.port` with `configFile server.xml` on the Traffic Router's [Profile](#). When `systemctl start traffic_router` is run, it will

generate self signed certificates at `/opt/traffic_router/conf/`, create a new Java Keystore named `/opt/traffic_router/conf/keyStore.jks`, and add the new certificate to the Keystore. The password for the Java Keystore and the Keystore location are stored in `/opt/traffic_router/conf/https.properties`. To override the self signed certificates with new ones from a certificate authority, either replace the Java Keystore in the default location or update the properties for the new Keystore location and password at `/opt/traffic_router/conf/https.properties` and then restart the Traffic Router using `systemctl`.

Traffic Router API endpoints only respond to GET requests.

### `/crs/stats`

General stats.

## Request Structure

148: Request Example

```
GET /crs/stats HTTP/1.1
Host: trafficrouter.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

## Response Structure

149: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked
Date: Tue, 15 Jan 2019 21:02:09 GMT

{ "app": {
  "buildTimestamp": "2019-01-10",
  "name": "traffic_router",
  "deploy-dir": "/opt/traffic_router",
  "git-revision": "437e9df81",
  "version": "3.0.0"
},
"stats": {
  "dnsMap": {},
  "httpMap": {
    "video.demo1.mycdn.ciab.test": {
      "czCount": 0,
```

(continues on next page)

(continued from previous page)

```

        "geoCount": 0,
        "deepCzCount": 0,
        "missCount": 0,
        "dsrCount": 0,
        "errCount": 0,
        "staticRouteCount": 0,
        "fedCount": 0,
        "regionalDeniedCount": 0,
        "regionalAlternateCount": 0
    }
},
"totalDnsCount": 0,
"totalHttpCount": 1,
"totalDsMissCount": 0,
"appStartTime": 1547584831677,
"averageDnsTime": 0,
"averageHttpTime": 1547584863270,
"updateTracker": {
    "lastHttpsCertificatesCheck": 1547586068932,
    "lastGeolocationDatabaseUpdaterUpdate": 1547584858917,
    "lastCacheStateCheck": 1547586128932,
    "lastCacheStateChange": 1547584867102,
    "lastNetworkUpdaterUpdate": 1547584857484,
    "lastHttpsCertificatesUpdate": 1547586071079,
    "lastSteeringWatcherUpdate": 1547584923514,
    "lastConfigCheck": 1547586127344,
    "lastConfigChange": 1547584863406,
    "lastNetworkUpdaterCheck": 1547584857465,
    "lastGeolocationDatabaseUpdaterCheck": 1547584858906,
    "lastFederationsWatcherUpdate": 1547584863433,
    "lastHttpsCertificatesFetchSuccess": 1547586069070,
    "lastSteeringWatcherCheck": 1547586124630,
    "lastFederationsWatcherCheck": 1547586124584,
    "lastHttpsCertificatesFetchAttempt": 1547586068932
}
}}
```

`/crs/stats/ip/{{IP}}`

Geolocation information for an IPv4 or IPv6 address.

## Request Structure

Table 83: Request Path Parameters

Name	Description
IP	The IP address for which statics will be returned. May be IPv4 or IPv6

### 150: Request Example

```
GET /crs/stats/ip/255.255.255.255 HTTP/1.1
Host: trafficrouter.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

## Response Structure

### 151: Response Example

```
HTTP/1.1 200 OK
Content-Disposition: inline;filename=f.txt
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked
Date: Tue, 15 Jan 2019 21:06:09 GMT

{ "locationByGeo": {
  "city": "Woodridge",
  "countryCode": "US",
  "latitude": "41.7518",
  "postalCode": "60517",
  "countryName": "United States",
  "longitude": "-88.0489"
},
"locationByFederation": "not found",
"requestIp": "69.241.118.34",
"locationByCoverageZone": "not found",
"locationByDeepCoverageZone": "not found"
}
```

## /crs/locations

A list of configured *Cache Groups* to which the Traffic Router is capable of routing client traffic.

### Request Structure

152: Request Example

```
GET /crs/locations HTTP/1.1
Host: trafficrouter.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

### Response Structure

**locations** An array of strings that are the *Names of Cache Groups* to which this Traffic Router is capable of routing client traffic

153: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked
Date: Tue, 15 Jan 2019 21:12:17 GMT

{ "locations": [
  "CDN_in_a_Box_Edge"
]}
```

## /crs/locations/caches

A mapping of caches to cache groups and their current health state.

### Request Structure

154: Request Example

```
GET /crs/locations/caches HTTP/1.1
Host: trafficrouter.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

## Response Structure

155: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked
Date: Tue, 15 Jan 2019 21:15:53 GMT

{ "locations": {
  "CDN_in_a_Box_Edge": [
    {
      "cacheId": "edge",
      "fqdn": "edge.infra.ciab.test",
      "ipAddresses": [
        "172.16.239.100",
        "fc01:9400:1000:8:0:0:0:100"
      ],
      "port": 0,
      "adminStatus": null,
      "lastUpdateHealthy": false,
      "lastUpdateTime": 0,
      "connections": 0,
      "currentBW": 0,
      "availBW": 0,
      "cacheOnline": true
    }
  ]
}}
```

/crs/locations/{{cachegroup}}/caches

A list of *cache servers* for this *Cache Group* only.

## Request Structure

Table 84: Request Path Parameters

Name	Description
cachegroup	The <i>Name of a Cache Group</i> of which a list of constituent <i>cache servers</i> will be retrieved

#### 156: Request Example

```
GET /crs/locations/CDN_in_a_Box_Edge/caches HTTP/1.1
Host: trafficrouter.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

### Response Structure

#### 157: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked
Date: Tue, 15 Jan 2019 21:18:25 GMT

{ "caches": [
  {
    "cacheId": "edge",
    "fqdn": "edge.infra.ciab.test",
    "ipAddresses": [
      "172.16.239.100",
      "fc01:9400:1000:8:0:0:0:100"
    ],
    "port": 0,
    "adminStatus": null,
    "lastUpdateHealthy": false,
    "lastUpdateTime": 0,
    "connections": 0,
    "currentBW": 0,
    "availBW": 0,
    "cacheOnline": true
  }
]}
```

### `/crs/consistenthash/cache/coveragezone`

The resulting cache of the consistent hash using coverage zone file for a given client IP, *Delivery Service*, and request path.

#### Request Structure

Table 85: Request Query Parameters

Name	Re-quired	Description
ip	yes	The IP address of a potential client
delivery-ServiceId	yes	The integral, unique identifier?'/xml_id'/?/name? of a <i>Delivery Service</i> served by this Traffic Router
requestPath	yes	The... request path?

#### Response Structure

TBD

### `/crs/consistenthash/cache/deep/coveragezone`

The resulting cache of the consistent hash using deep coverage zone file (deep caching) for a given client IP, *Delivery Service*, and request path.

#### Request Structure

Table 86: Request Query Parameters

Name	Re-quired	Description
ip	yes	The IP address of a potential client
delivery-ServiceId	yes	The integral, unique identifier?'/xml_id'/?/name? of a <i>Delivery Service</i> served by this Traffic Router
requestPath	yes	The... request path?



## Response Structure

TBD

### /crs/consistenthash/cache/geolocation

The resulting cache of the consistent hash using geographic location for a given client IP, *Delivery Service*, and request path.

## Request Structure

Table 87: Request Query Parameters

Name	Re-quired	Description
ip	yes	The IP address of a potential client
delivery-ServiceId	yes	The integral, unique identifier?/?xml_id'?/name? of a <i>Delivery Service</i> served by this Traffic Router
requestPath	yes	The... request path?

## Response Structure

TBD

### /crs/consistenthash/deliveryservice/

The resulting *Delivery Service* of the consistent hash for a given *Delivery Service* and request path – used to test STEERING *Delivery Services*.

## Request Structure

Table 88: Request Query Parameters

Name	Re-quired	Description
delivery-ServiceId	yes	The integral, unique identifier?/?xml_id'?/name? of a <i>Delivery Service</i> served by this Traffic Router
requestPath	yes	The... request path?

### 158: Request Example

```
GET /crs/consistenthash/deliveryservice?deliveryServiceId=demo1&
→requestPath=/ HTTP/1.1
Host: trafficrouter.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

## Response Structure

### 159: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Date: Tue, 15 Jan 2019 21:40:51 GMT

{ "id": "demo1",
  "coverageZoneOnly": false,
  "geoRedirectUrl": null,
  "geoRedirectFile": null,
  "geoRedirectUrlType": "INVALID_URL",
  "routingName": "video",
  "missLocation": {
    "latitude": 42,
    "longitude": -88,
    "postalCode": null,
    "city": null,
    "countryCode": null,
    "countryName": null,
    "defaultLocation": false,
    "properties": {
      "city": null,
      "countryCode": null,
      "latitude": "42.0",
      "postalCode": null,
      "countryName": null,
      "longitude": "-88.0"
    }
  },
  "dispersion": {
    "limit": 1,
    "shuffled": true
  },
  "ip6RoutingEnabled": true,
```

(continues on next page)

(continued from previous page)

```
"responseHeaders": {},
"requestHeaders": [],
"regionalGeoEnabled": false,
"geolocationProvider": "maxmindGeolocationService",
"anonymousIpEnabled": false,
"sslEnabled": true,
"acceptHttp": true,
"deepCache": "NEVER",
"dns": false,
"locationLimit": 0,
"maxDnsIps": 0,
"sslReady": true,
"available": true
}
```

`/crs/coveragezone/caches`

A list of caches for a given *Delivery Service* and *Cache Group*.

## Request Structure

Table 89: Request Query Parameters

Name	Re-quired	Description
delivery-ServiceId	yes	The integral, unique identifier?'/xml_id'/?/name? of a <i>Delivery Service</i> served by this Traffic Router
cacheLocationId	yes	The <i>Name of a Cache Group</i> to which this Traffic Router is capable of routing client traffic

## Response Structure

TBD

`/crs/coveragezone/cachelocation`

The resulting *Cache Group* for a given client IP and *Delivery Service*.

## Request Structure

Table 90: Request Query Parameters

Name	Re-quired	Description
ip	yes	The IP address of a potential client
delivery-ServiceId	yes	The integral, unique identifier?/'xml_id'/?/name? of a <i>Delivery Service</i> served by this Traffic Router

## Response Structure

TBD

### /crs/deepcoveragezone/cachelocation

The resulting *Cache Group* using the *Deep Coverage Zone File* (deep caching) for a given client IP and *Delivery Service*.

## Request Structure

Table 91: Request Query Parameters

Name	Re-quired	Description
ip	yes	The IP address of a potential client
delivery-ServiceId	yes	The integral, unique identifier?/'xml_id'/?/name? of a <i>Delivery Service</i> served by this Traffic Router

## Response Structure

TBD

### /crs/consistenthash/patternbased/regex

The resulting path that will be used for consistent hashing when the given regex is applied to the given request path.

## Request Structure

Table 92: Request Query Parameters

Name	Re-quired	Description
regex	yes	The (URI encoded) regular expression to be used to test pattern based consistent hashing
request-Path	yes	The (URI encoded) request path to use to test pattern based consistent hashing

### 160: Request Example

```
GET /crs/consistenthash/patternbased/regex?regex=%2F.*%3F%28%2F.*%3F%2F
→%29.*%3F%28%5C.m3u8%29&requestPath=%2Ftext1234%2Fname%2Fasset.m3u8
→HTTP/1.1
Host: localhost:3333
User-Agent: curl/7.54.0
Accept: */*
```

## Response Structure

### 161: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Transfer-Encoding: chunked
Date: Fri, 15 Feb 2019 22:06:53 GMT

{
  "resultingPathToConsistentHash": "/name/.m3u8",
  "consistentHashRegex": "/.*?(/.*?/).*(\\.m3u8)",
  "requestPath": "/text1234/name/asset.m3u8"
}
```

### /crs/consistenthash/patternbased/deliveryservice

The resulting path that will be used for consistent hashing for the given delivery service and the given request path.

## Request Structure

Table 93: Request Query Parameters

Name	Re- quired	Description
requestPath	yes	The (URI encoded) request path to use to test pattern based consistent hashing
delivery- ServiceId	yes	The integral, unique identifier?/'xml_id'?/name? of a <i>Delivery Service</i> served by this Traffic Router

#### 162: Request Example

```
GET /crs/consistenthash/patternbased/deliveryservice?
→deliveryServiceId=asdf&requestPath=%2Fsometext1234%2Fstream_name
→%2Fasset_name.m3u8 HTTP/1.1
Host: localhost:3333
User-Agent: curl/7.54.0
Accept: */*
```

## Response Structure

#### 163: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Transfer-Encoding: chunked
Date: Fri, 15 Feb 2019 22:12:38 GMT

{
  "resultingPathToConsistentHash": "/sometext1234/stream_name/asset_name.
→m3u8",
  "deliveryServiceId": "asdf",
  "requestPath": "/sometext1234/stream_name/asset_name.m3u8"
}
```

### /crs/consistenthash/cache/coveragezone/steering

The resulting cache of the consistent hash using coverage zone for a given client IP, delivery service and, request path – used to test cache selection for steering delivery services.

## Request Structure

Table 94: Request Query Parameters

Name	Re-quired	Description
requestPath	yes	The (URI encoded) request path to use to test pattern based consistent hashing
delivery-ServiceId	yes	The integral, unique identifier?'/xml_id'/?/name? of a <i>Delivery Service</i> served by this Traffic Router
ip	yes	The IP address of a potential client

### Response Structure

TBD

### 4.1.11 Traffic Stats

#### Introduction

Traffic Stats is a utility written in [Go](#) that is used to acquire and store statistics about CDNs controlled by Traffic Control. Traffic Stats mines metrics from the [Traffic Monitor APIs](#) and stores the data in [InfluxDB](#). Data is typically stored in InfluxDB on a short-term basis (30 days or less) and is used to drive graphs created by [Grafana](#) which are linked from Traffic Ops. Traffic Stats also calculates daily statistics from InfluxDb and stores them in the Traffic Ops database.

#### Software Requirements

- A \*nix (MacOS and Linux are most commonly used) environment
- [Go 1.7.x or above](#)
- Access to a working instance of Traffic Ops
- Access to a working instance of Traffic Monitor
- [InfluxDB version 1.0.0 or greater](#)

#### Traffic Stats Project Tree Overview

- `traffic_stats/` - contains Go source files and files used to create the Traffic Stats RPM.
  - `grafana/` - contains a javascript file which is installed on the Grafana server. This allows Traffic Ops to create custom dashboards for [Delivery Services](#), `:term:cache server`s`, etc.
  - `influxdb_tools/` - contains [sync/sync\\_ts\\_databases.go](#) and [create/create\\_ts\\_databases.go](#) which are helpful if you have multiple instances and they get out of sync with data.

#### Go Formatting Conventions

In general [Go fmt](#) is the standard for formatting Go code. It is also recommended to use [Go lint](#).



## Installing The Developer Environment

1. Clone the `traffic_control` repository using Git into a location accessible by your `$GOPATH`
2. Navigate to the `traffic_ops/v4-client` directory of your cloned repository. (This is the directory containing Traffic Ops client code used by Traffic Stats)
3. From the `traffic_ops/v4-client` directory, run `go test` to test the client code. This will run all unit tests for the client and return the results. If there are missing dependencies you will need to run `go mod vendor -v` to get the dependencies
4. Once the tests pass, run `go install` to build and install the Traffic Ops client package. This makes it accessible to Traffic Stats.
5. Navigate to your cloned repository under Traffic Stats
6. Run `go build traffic_stats.go` to build `traffic_stats`. You will need to run `go mod vendor -v` for any missing dependencies.

### 4.1.12 The Development Environment

A development environment is available in `dev/`. This environment only depends on `Docker` (version 20+) and `Docker-Compose` <<https://docs.docker.com/compose/> (version 1.27+) and enables rapid changes to be made to components during active development. This is, in general far faster than *Debugging inside CDN-in-a-Box*, but covers less complex configurations for testing purposes. Continuous Integration typically makes use of *CDN-in-a-Box*, so developers in general are free to use the Development Environment.

---

**Note:** Many ports used by the development environment clash with those exposed locally by *CDN-in-a-Box* when the `infrastructure/cdn-in-a-box/docker-compose.expose-ports.yml` Compose file is included, so the two cannot be used at the same time.

---

## atc

The command `atc` is made available by sourcing `dev/atc.dev.sh` (e.g. `source dev/atc.dev.sh`). While at the repository root, this command can be used to manipulate the development environment - most notably stopping and starting it.

Sourcing this file also sets `TO_URL`, `TO_USER`, and `TO_PASSWORD` to the values appropriate for the default setup of the development environment, such that `toaccess` may be used to access the development Traffic Ops instance without any extra steps.

### 164: atc Usage

```
atc [-h] OPERATION
```

**-h, --help**

Print usage information and exit.

Each valid OPERATION is outlined in its corresponding section.

### build

Build Docker images for the environment, but do not start it.

#### 165: atc build Usage

```
atc build [SERVICE...]
```

#### SERVICE

If specified, only the given services will be built. By default, all services are built.

#### 166: atc build Example

```
# Build only Traffic Ops
atc build trafficops

# Build all services
atc build
```

### connect

Connect to a shell session inside a development container.

---

**Note:** Connecting to `trafficrouter` will result in connecting as a non-root user, so privileged access requires a more careful, manual use of `docker(1)`.

---

### exec

Run a command in a dev container.

#### 167: atc exec Usage

```
atc exec SERVICE CMD...
```

#### SERVICE

The service within which to execute commands.

#### CMD

An argv to pass to the service as a command.

#### 168: atc exec Example

```
# Check ping statistics for communications from Traffic Ops to Traffic_
↳Monitor.
atc exec trafficops ping -c 4 trafficmonitor
```

## ready

Check if the development environment is ready. If it is ready the exit code is 0, and if it isn't ready the exit code is non-zero. "Readiness" is defined by the availability of the Traffic Ops API.

#### 169: atc ready [-h] [-w] Usage

```
atc ready [SERVICE...]
```

**-h, --help**

Print usage information and exit.

**-w, --wait**

Wait for ATC to be ready, instead of just checking if it is ready.

#### 170: atc ready Example

```
# Print "ready" if ATC is ready, "not ready" if it isn't.
if atc ready; then
    echo "ready";
else
    echo "not ready";
fi

# Block until ATC is ready before getting a CDN Snapshot for the_
↳development CDN.
atc ready -w && toget -k cdns/dev/snapshot
```

## restart

Restart the development environment. This is functionally equivalent to *stop* followed by *start* where the same arguments that would be passed to *restart* are instead given to each of those.

**Warning:** Restarting Traffic Ops also stops every service that either it depends on or that depends on it - which is all of them. However, it only *starts* the services that Traffic Ops *depends on*, which is only the database service. So `atc restart trafficops` stops everything and only starts Traffic Ops back up again.

---

**Tip:** The services automatically rebuild the ATC components they run when those components change, so usually restarting is only necessary if you're making changes to the developer environment itself.

---

### 171: atc restart Usage

```
atc restart [SERVICE...]
```

#### **SERVICE**

If specified, only the given services will be restarted. By default, all services are restarted.

### 172: atc restart Example

```
# Restart only Traffic Router
atc restart trafficrouter

# Restart all services
atc restart
```

## **start**

Start up the development environment.

---

**Note:** Starting Traffic Ops also starts the Traffic Ops database and Traffic Vault (which isn't its own service).

---

### 173: atc start Usage

```
atc start [SERVICE...]
```

#### **SERVICE**

If specified, only the given services will be started. By default, all services are started.

174: atc start Example

```
# Start only Traffic Portal
atc start trafficportal

# Start all services
atc start
```

## stop

Stop the development environment.

---

**Note:** Stopping Traffic Ops also stops every service that either it depends on or that depends on it - which is all of them.

---

175: atc stop Usage

```
atc stop [SERVICE...]
```

### SERVICE

If specified, only the given services will be built. By default, all services are built.

176: atc stop Example

```
# Stop only Traffic Router
atc stop trafficrouter

# Stop all services
atc stop
```

## t3c

The `atc.dev.sh` file also provides a way to run `t3c-t3c` commands with debugging enabled. While in most production deployments `t3c-t3c` runs on a `cron(8)` schedule, it is never run in the development environment, normally. One must manually trigger a run in order to run it and debug it.

The usage of this provided function is exactly as if one were simply calling `t3c-t3c`. A [delve](#) debugging session is automatically started when `t3c-t3c` is run, which listens on port 8081.



A guide to external RESTful APIs for Traffic Ops

## 5.1 Traffic Ops API

The Traffic Ops API provides programmatic access to read and write Traffic Control data which allows for the monitoring of CDN performance and configuration of Traffic Control settings and parameters.

### 5.1.1 How to Read this Documentation

Each endpoint for each version is on its own page, titled with the request path. The request paths shown on each endpoint's page are - unless otherwise noted - only usable by being appended to the request path prefix `/api/<version>/` where `<version>` is the API version being requested. The API versions officially supported as of the time of this writing are 3.0, 3.1, and 4.0. All endpoints are documented as though they were being used in version 3.1 in the version 3 documentation and version 4.0 in the version 4 documentation. If an endpoint or request method of an endpoint is only available after a specific version, that will be noted next to the method or endpoint name. If changes were made to the structure of an endpoint's input or output, the version number and nature of the change will be noted.

Every endpoint is documented with a section for each method, containing the subsections "Request Structure" and "Response Structure" which identify all properties and structure of the Request to and Response from the endpoint. Before these subsections, three key pieces of information will be provided:

**Auth. Required** This will either be 'Yes' to indicate that a user must be authenticated (or "logged-in") via e.g. [user/login](#) to use this method of the endpoint, or 'No' to indicate that this is not required.

**Roles Required** Deprecated since version ATCv7.0: Roles for the use of authentication/permissions have been deprecated in favor of role based permissions, see [Pull Request 5848](#).

Any permissions roles that are allowed to use this method of the endpoint will be listed here. Users with roles not listed here will be unable to properly use these endpoints.

**Permissions Required** Any permissions that are needed to use this endpoint. Users with roles that don't have the permissions will be unable to properly use these endpoints.

**Response Type** Unless otherwise noted, all responses are JSON objects. See [Response Structure](#) for more information.

The methods of endpoints that require/accept data payloads - unless otherwise noted - always interpret the content of the payload as a JSON object, regardless of the request's `Content-Type` header. Because of this, all payloads are - unless otherwise noted - JSON objects. The Request Structure and Response Structure subsections will contain explanations of the fields before any examples like e.g.

**foo** A constant field that always contains "foo"

**bar** An array of objects that each represent a "bar" object

**name** The bar's name

**value** The bar's value (an integer)

All fields are mandatory in a request payload, or always present in a response payload unless otherwise noted in the field description.

In most cases, JSON objects have been "pretty-printed" by inserting line breaks and indentation. This means that the `Content-Length` HTTP header does not, in general, accurately portray the length of the content displayed in Request Examples and Response Examples. Also, the Traffic Ops endpoints will ignore any content negotiation, meaning that the `Content-Type` header of a request is totally meaningless. A utility may choose to pass the data as e.g. `application/x-www-form-urlencoded` (cURL's default `Content-Type`) when constructing a Request Example, but the example itself will most often show `application/json` in order for syntax highlighting to properly work.

## Response Structure

Unless otherwise noted, all response payloads come as JSON objects.

### 177: Response Structure

```
{
  "response": "<JSON object with main response>",
}
```

To make the documentation easier to read, only the `<JSON object with main response>` is documented, even though the response endpoints may return other top-level objects (most commonly the "alerts" object). The field definitions listed in the Response Structure subsection of an endpoint method are the elements of this object. Sometimes the response object is a string, sometimes it's an object that maps keys to values, sometimes it's an array that contains many arbitrary objects, and sometimes it isn't present at all. For ease of reading, the field lists delegate the distinction to be made by the `Response Type` field directly under the request method heading.



## Response Type Meanings

**Array** The fields in the field list refer to the keys of the objects in the response array.

**Object** The fields in the field list refer to the keys of the response object.

**undefined** No response object is present in the response payload. Unless the format is otherwise noted, this means that there should be no field list in the “Response Structure” subsection.

## Summary

The top-level `summary` object is used to provide summary statistics about object collections. In general the use of `summary` is left to be defined by API endpoints (subject to some restrictions). When an endpoint uses the `summary` object, its fields will be explained in a subsection of the “Response Structure” named “Summary Fields”.

The following, reserved properties of `summary` are guaranteed to always have their herein-described meaning.

**count** `count` contains an unsigned integer that defines the total number of results that could possibly be returned given the non-pagination query parameters supplied by the client.

## Traffic Ops’s Custom Date/Time Format

Traffic Ops will often return responses from its API that include dates. As of the time of this writing, the vast majority of those dates are written in a non-`rfc:3339` format (this is tracked by [Issue #5911](#)). This is most commonly the case in the `last_updated` properties of objects returned as JSON-encoded documents. The format used is `YYYY-MM-DD hh:mm:ss±ZZ`, where `YYYY` is the 4-digit year, `MM` is the two-digit (zero padded) month, `DD` is the two-digit (zero padded) day of the month, `hh` is the two-digit (zero padded) hour of the day, `mm` is the two-digit (zero padded) minute of the hour, `ss` is the two-digit (zero padded) second of the minute, and `ZZ` is the two-digit (zero padded) timezone offset in hours of the date/time’s local timezone from UTC (the offset can be positive or negative as indicated by a `+` or a `-` directly before it, where the sample has a `±`).

---

**Note:** In practice, all Traffic Ops API responses use the UTC timezone (offset `+00`), but do note that this custom format is not capable of representing all timezones.

---

178: Example Date/Timestamp

```
2021-06-07 08:01:02+00
```

## 5.1.2 Using API Endpoints

1. Authenticate with valid Traffic Control user account credentials (the same used by Traffic Portal).
2. Upon successful user authentication, note the Mojolicious cookie value in the response headers<sup>1</sup>.

---

**Note:** Many tools have methods for doing this without manual intervention - a web browser for instance will automatically remember and properly handle cookies. Another common tool, cURL, has command line switches that will also accomplish this. Most high-level programming language libraries will implement a cookie-handling method as well.

---

3. Pass the Mojolicious cookie value, along with any subsequent calls to an authenticated API endpoint.

---

**Note:** Although many endpoints in API version 1.x supported a `.json` suffix, API version 2.x does not support it at all. Even when using API version 1.x using the `.json` suffix should be avoided at all costs, because there's no real consistency regarding when it may be used, and the output of API endpoints, in general, are not capable of representing POSIX-compliant files (as a 'file extension' might imply).

---

### Example Session

A user makes a request to the `/api/4.0/asns` endpoint.

```
GET /api/4.0/asns HTTP/1.1
Accept: application/json
Host: trafficops.infra.ciab.test
User-Agent: example
```

The response JSON indicates an authentication error.

---

<sup>1</sup> A cookie obtained by logging in through Traffic Portal can be used to access API endpoints under the Traffic Portal domain name - since it will proxy such requests back to Traffic Ops. This is not recommended in actual deployments, however, because it will involve an extra network connection which could be avoided by simply using the Traffic Ops domain itself.

```
HTTP/1.1 401 UNAUTHORIZED
Content-Length: 68
Content-Type: application/json
Date: Tue, 02 Oct 2018 13:12:30 GMT

{ "alerts": [
  {
    "level": "error",
    "text": "Unauthorized, please log in."
  }
]}
```

To authenticate, the user sends a POST request containing their login information to the `/api/4.0/user/login` endpoint.

```
POST /api/4.0/user/login HTTP/1.1
User-Agent: example
Host: trafficops.infra.ciab.test
Accept: application/json
Content-Length: 32
Content-Type: application/x-www-form-urlencoded
```

Traffic Ops responds with a Mojolicious cookie to be used for future requests, and a message indicating the success or failure (in this case success) of the login operation.

```
HTTP/1.1 200 OK
Connection: keep-alive
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Content-Type: application/json
Date: Tue, 02 Oct 2018 12:53:32 GMT
Access-Control-Allow-Credentials: true
Content-Length: 81
X-Server-Name: traffic_ops_golang/

{ "alerts": [
  {
    "level": "success",
    "text": "Successfully logged in."
  }
]}
```

Using this cookie, the user can now access their original target - the `/api/4.0/asns` endpoint...

```
GET /api/4.0/asns HTTP/1.1
Accept: application/json
Cookie: mojolicious=...;
Host: trafficops.infra.ciab.test
User-Agent: Example
```

... and the Traffic Ops server will now happily service this request.

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Connection: keep-alive
Content-Encoding: gzip
Content-Length: 48
Content-Type: application/json
Date: Tue, 02 Oct 2018 12:55:57 GMT
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-SHA512: u+Q5X7z/DMTc/
→VzRGaFlJBA8btA8EC...dnA85HCYTm8vVwsQCvle+uVc1nA==
X-Server-Name: traffic_ops_golang/

{ "response": {
  "asns": [
    {
      "lastUpdated": "2012-09-17 21:41:22",
      "id": 27,
      "asn": 7015,
      "cachegroup": "us-ma-woburn",
      "cachegroupId": 2
    },
    {
      "lastUpdated": "2012-09-17 21:41:22",
      "id": 28,
      "asn": 7016,
      "cachegroup": "us-pa-pittsburgh",
      "cachegroupID": 3
    }
  ]
}}
```

### 5.1.3 API Errors

If an API endpoint has something to say besides the actual response (usually an error message), it will add a top-level object to the response JSON with the key "alerts". This will be an array of objects that represent messages from the server, each with the following string fields:

**level** "success", "info", "warning" or "error" as appropriate

**text** The alert's actual message

The most common errors returned by Traffic Ops are:

**401 Unauthorized** When a “mojolicious” cookie is supplied that is invalid or expired, or the login credentials are incorrect the server responds with a 401 UNAUTHORIZED response code.

179: Example of a Response to a Login Request with Bad Credentials

```
HTTP/1.1 401 Unauthorized
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-
    ↳Type, Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Whole-Content-Sha512: xRKu2Q7Yj07UA6A6SyxMNmcBpuBcW2/
    ↳bzuK05eTZ2y4V27rXfP/5bSkNPesomJbi00+xSmiybDsHlcL3P+pzpg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 02 Oct 2018 13:28:30 GMT
Content-Length: 69

{ "alerts": [
  {
    "text": "Invalid username or password.",
    "level": "error"
  }
]}
```

**404 Not Found** When the requested resource (path) doesn't exist, Traffic Ops returns a 404 NOT FOUND response code.

180: Example Response to GET /not/an/api/path  
HTTP/1.1 with Proper Cookies

```
HTTP/1.1 404 Not Found
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-
    ↳Type, Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: text/html; charset=UTF-8
Date: Tue, 02 Oct 2018 13:58:56 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019
→17:40:54 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: Ff5h08ZUNUMbwCW0mBuUlsvrSmm/
→Giijpq703uLivLZ6V0u6eGom4Jag6UqlBbbDBnP6AG7l1Szdt74TT6NidA==
Transfer-Encoding: chunked

{ "alerts": [
  {
    "level": "error",
    "text": "Resource not found."
  }
]}
}]}
```

**500 Internal Server Error** When a server-side error occurs, the API will return a 500 INTERNAL SERVER ERROR response.

181: Example Response to GET /api/4.0/servers.  
(when a server error such as a postgres failure occurred)

```
HTTP/1.1 500 Internal Server Error
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-
→Type, Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Length: 93
Content-Type: application/json
Date: Tue, 02 Oct 2018 17:29:42 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019
→17:40:54 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→gFa4NYFmofCbV7YqgwyFRzKk90+KNgoZu6p2Nx98J4Gy7/
→2j55tYknvk53WXuMdMKKrgYMop4uiY0la1k1ozQQ==

{ "alerts": [
  {
    "level": "error",
    "text": "An error occurred. Please contact your
→administrator."
```

(continues on next page)

(continued from previous page)

```
}
[]}
```

The rest of the API documentation will only document the 200 OK case, where no errors have occurred.

### 5.1.4 TrafficOps Native Client Libraries

TrafficOps client libraries are available in Java, Go and Python. You can read (very little) more about them in the client README at [traffic\\_control/clients](https://traffic-control/clients).

### 5.1.5 API V3 Routes

API routes available in version 3.

Deprecated since version ATCv7: Traffic Ops API version 3 is deprecated in favor of version 4.

#### about

#### GET

Returns info about the Traffic Ops build that is currently running, generated at startup. The output will be the same until the Traffic Ops *version changes*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

#### Request Structure

No parameters available.

182: Request Example

```
GET /api/3.0/about HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**commitHash** The [Git](#) commit hash that Traffic Ops was built at.

**commits** The number of commits in the branch of the commit that Traffic Ops was built at, including that commit. Calculated by running `git rev-list HEAD | wc -l`.

**goVersion** The version of [Go](#) that was used to build Traffic Ops.

**release** The major version of CentOS or Red Hat Enterprise Linux that the build environment was running.

**name** The human-readable name of the [RPM](#) file.

**RPMVersion** The entire name of the RPM file, excluding the file extension.

**Version** The version of ATC that this version of Traffic Control belongs to.

### 183: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 19:35:28.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→7SVQsddCUVRs+sineziRGR6OyMli7XLZbjxyMQgW6E506bh5thM0uttPFT7aJckDcgT45PlhexycwlApO
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 18:35:28 GMT
Content-Length: 145

{
  "commitHash": "1c9a2e9c",
  "commits": "10555",
  "goVersion": "go1.11.13",
  "release": "el7",
  "name": "traffic_ops",
  "RPMVersion": "traffic_ops-4.0.0-10555.1c9a2e9c.el7",
  "Version": "4.0.0"
}
```



## acme\_accounts

New in version 3.1.

### GET

Gets information for all *ACME Account* s.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Array

### Request Structure

No parameters available

### Response Structure

**email** The email connected to the *ACME Account*.

**privateKey** The private key connected to the *ACME Account*.

**uri** The URI for the *ACME Account*. Differs per provider.

**provider** The ACME provider.

#### 184: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": [
  {
    "email": "sample@example.com",
    "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----END RSA PRIVATE KEY-----\n",
    "uri": "https://acme.example.com/acct/1",
    "provider": "Lets Encrypt"
  }
]}
```

## POST

Creates a new *ACME Account*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

The request body must be a single *ACME Account* object with the following keys:

**email** The email connected to the *ACME Account*.

**privateKey** The private key connected to the *ACME Account*.

**uri** The URI for the *ACME Account*. Differs per provider.

**provider** The ACME provider.

### 185: Request Example

```
POST /api/3.1/acme_accounts HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 181
Content-Type: application/json

{
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
↪END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}
```

## Response Structure

**email** The email connected to the *ACME Account*.

**privateKey** The private key connected to the *ACME Account*.

**uri** The URI for the *ACME Account*. Differs per provider.

**provider** The ACME provider.

## 186: Response Example

```

HTTP/1.1 201 Created
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Dec 2020 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳eQrl48zWids0kDpfCYmntYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
    ↳Cfh2X9i3G+MQ36eH95gukJq0BOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:18:21 GMT
Content-Length: 253

{ "alerts": [
  {
    "text": "Acme account created",
    "level": "success"
  }
],
"response": {
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
    ↳END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}}
```

## PUT

Updates an existing *ACME Account*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

The request body must be a single *ACME Account* object with the following keys:

- email** The email connected to the *ACME Account*.
- privateKey** The private key connected to the *ACME Account*.
- uri** The URI for the *ACME Account*. Differs per provider.
- provider** The ACME provider.

### 187: Request Example

```
PUT /api/3.1/acme_accounts HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 181
Content-Type: application/json

{
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
→END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}
```

## Response Structure

- email** The email connected to the *ACME Account*.
- privateKey** The private key connected to the *ACME Account*.
- uri** The URI for the *ACME Account*. Differs per provider.
- provider** The ACME provider.

### 188: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Dec 2020 17:40:54.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512:↵
↵eQrl48zWids0kDpfCYmntYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
↵Cfh2X9i3G+MQ36eH95gukJq0BOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:18:21 GMT
Content-Length: 253

{ "alerts": [
  {
    "text": "Acme account updated",
    "level": "success"
  }
],
"response": {
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
↵END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}}
```

`acme_accounts/{provider}/{email}`

New in version 3.1.

## DELETE

Delete *ACME Account* information.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 95: Request Path Parameters

Name	Description
provider	The ACME provider for the account to be deleted
email	The email used in the <i>ACME Account</i> to be deleted

## Response Structure

### 189: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Dec 2020 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: rGD2sOMHYF0sga1zuTytyLHCUkkc3ZwQRKvZ/
    ↳HuPzObOP4WztKTOVXB4uhs3iJqBg9zRB2TucMxONHN+3/yShQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 10 Dec 2020 14:24:34 GMT
Content-Length: 60

{"alerts": [
  {
    "text": "Acme account deleted",
    "level": "success"
  }
]}
```

## api\_capabilities

Deprecated since version 3.1.

Deals with the capabilities that may be associated with API endpoints and methods. These capabilities are assigned to *Roles*, of which a user may have one or more. Capabilities support “wildcarding” or “globbing” using asterisks to group multiple routes into a single capability

### GET

Get all API-capability mappings.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 96: Request Query Parameters

Name	Required	Type	Description
capability	no	string	Capability name

### 190: Request Example

```
GET /api/3.0/api_capabilities?capability=types-write HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**capability** Capability name

**httpMethod** An HTTP request method, practically one of:

- GET
- POST
- PUT
- PATCH
- DELETE

**httpRoute** The request route for which this capability applies - relative to the Traffic Ops server's URL

**id** An integer which uniquely identifies this capability

**lastUpdated** The time at which this capability was last updated, in *Traffic Ops's Custom Date/Time Format*

### 191: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 01 Nov 2018 14:45:24 GMT
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: wptErtIop/AfTTQ+1MZdA2YpPXEOuLFfrPQvvaHq0/
→uX5fRruOVYW+7p8JTTrtH1xg1WN+x6FnjQnSHuWwcpyJg==
Content-Length: 393

{ "response": [
  {
    "httpMethod": "POST",
    "lastUpdated": "2018-11-01 14:10:22.794114+00",
    "httpRoute": "types",
    "id": 261,
    "capability": "types-write"
  },
  {
    "httpMethod": "PUT",
    "lastUpdated": "2018-11-01 14:10:22.795917+00",
    "httpRoute": "types/*",
    "id": 262,
    "capability": "types-write"
  },
  {
    "httpMethod": "DELETE",
    "lastUpdated": "2018-11-01 14:10:22.799748+00",
    "httpRoute": "types/*",
    "id": 263,
    "capability": "types-write"
  }
]}

```

asns

See also:

[The Autonomous System Wikipedia page](#) for an explanation of what an ASN actually is.



## GET

List all ASNs.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 97: Request Query Parameters

Parameter	Required	Description
cacheGroup	no	The <i>ID</i> of a <i>Cache Group</i> - only ASNs for this <i>Cache Group</i> will be returned.
id	no	The integral, unique identifier of the desired ASN-to- <i>Cache Group</i> association
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n<sup>th</sup></i> page of results, where “n” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

## 192: Request Example

```
GET /api/3.0/asns HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- asn** An ASN as specified by IANA for identifying a service provider
- cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN
- cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN
- id** An integral, unique identifier for this association between an ASN and a *Cache Group*
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

## 193: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 22:51:14
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳F2NmDbTpXqrIQDX7IBKH9+1drTL4XedSfJv6klMgLEZwbLCkddIXuSLpmgVCID6kTVqy3fTKjZS3U+HJ
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 21:51:14 GMT
Content-Length: 128

{ "response": [
  {
    "asn": 1,
    "cachegroup": "TRAFFIC_ANALYTICS",
    "cachegroupId": 1,
```

(continues on next page)

(continued from previous page)

```

    "id": 1,
    "lastUpdated": "2019-12-02 21:49:08+00"
  }
}]

```

## POST

Creates a new ASN.

---

**Note:** There cannot be two different ASN object with the same `asn`. An ASN may only belong to one cachegroup, but a cachegroup can have zero or more ASNs.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**asn** The value of the new ASN

**cachegroup** An optional field which, if present, is a string that specifies the *Name* of a *Cache Group* to which this ASN will be assigned

---

**Note:** While this endpoint accepts the `cachegroup` field, sending this in the request payload has no effect except that the response will (erroneously) name the *Cache Group* to which the ASN was assigned. Any subsequent requests will reveal that, in fact, the *Cache Group* is set entirely by the `cachegroupId` field, and so the actual *Name* may differ from what was in the request.

---

**cachegroupId** An integer that is the *ID* of a *Cache Group* to which this ASN will be assigned

### 194: Request Example

```

POST /api/3.0/asns HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 29

```

(continues on next page)

(continued from previous page)

```
{"asn": 1, "cachegroupId": 1}
```

## Response Structure

- asn** An ASN as specified by IANA for identifying a service provider
- cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN
- cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN
- id** An integral, unique identifier for this association between an ASN and a *Cache Group*
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

### 195: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 22:49:08
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→mx8b2GTyojz4QtMxXCmoQyZogCB504vs0yv6WGly4dwM81W3XiejWNuUwchRBYyi8QHawsMZ3DaiGGfQi
→8Giw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 21:49:08 GMT
Content-Length: 150

{ "alerts": [
  {
    "text": "asn was created.",
    "level": "success"
  }
],
"response": {
  "asn": 1,
  "cachegroup": null,
```

(continues on next page)

(continued from previous page)

```
"cachegroupId": 1,
"id": 1,
"lastUpdated": "2019-12-02 21:49:08+00"
}}
```

## PUT

Updates an existing ASN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**asn** The value of the new ASN.

**cachegroup** A string that specifies the *Name* of a *Cache Group* to which this ASN will be assigned. If you do not pass this field, the cachegroup will be null.

**cachegroupId** The integral, unique identifier of the status of the *Cache Group*.

### 196: Request Example

```
PUT /api/3.0/asns?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 53

{
  "asn": 1,
  "cachegroup": "TRAFFIC_OPS",
  "cachegroupId": 2
}
```

## Response Structure

- asn** An ASN as specified by IANA for identifying a service provider
- cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN
- cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN
- id** An integral, unique identifier for this association between an ASN and a *Cache Group*
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

### 197: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 07:21:10
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳MjvwJg6AFbdqGPlAhK+2pfiN+VFjzgeNnhXoMVbh6+fRQYKeej6CCj3x09hw0l4uhp9d9RySrE/
  ↳CQ3+L1b2VGQ==
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

(continued from previous page)

Date: Tue, 25 Feb 2020 06:21:10 GMT

Content-Length: 164

```
{
  "alerts": [
    {
      "text": "asn was updated.",
      "level": "success"
    }
  ],
  "response": {
    "asn": 1,
    "cachegroup": "TRAFFIC_OPS",
    "cachegroupId": 2,
    "id": 1,
    "lastUpdated": "2020-02-25 06:21:10+00"
  }
}
```

## DELETE

Deletes an existing ASN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

#### 198: Request Example

```
DELETE /api/3.0/asns?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

### Response Structure

#### 199: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 08:27:33
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Woz8NSHIYVpX4V5X4xZWZIX1hvGL2uian7nUhjZ8F23Nb9RWQRMig/cc+1vXEzkT/
    ↳ehKV9t11FKRLX+avSae0g==
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 07:27:33 GMT
Content-Length: 83

{
  "alerts": [
    {
      "text": "asn was deleted.",
      "level": "success"
    }
  ]
}
```



`asns/{{id}}`

**See also:**

The [Autonomous System Wikipedia page](#) for an explanation of what an ASN actually is.

## PUT

Allows user to edit an existing ASN-to-*Cache Group* association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**asn** The new ASN which will be associated with the identified *Cache Group* - must not conflict with existing associations

**cachegroup** An optional field which, if present, is a string that specifies the *Name* of a *Cache Group* to which this ASN will be assigned

---

**Note:** While this endpoint accepts the `cachegroup` field, sending this in the request payload has no effect except that the response will (erroneously) name the *Cache Group* to which the ASN was assigned. Any subsequent requests will reveal that, in fact, the *Cache Group* is set entirely by the `cachegroupId` field, and so the actual *Name* may differ from what was in the request.

---

**cachegroupId** An integer that is the *ID* of a *Cache Group* to which this ASN will be assigned - must not conflict with existing associations

Table 98: Request Path Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the desired ASN-to- <i>Cache Group</i> association

## 200: Request Example

```
PUT /api/3.0/asns/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

(continued from previous page)

```
Content-Length: 29
Content-Type: application/x-www-form-urlencoded

{"asn": 2, "cachegroupId": 1}
```

## Response Structure

- asn** An ASN as specified by IANA for identifying a service provider
- cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN
- cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN
- id** An integral, unique identifier for this association between an ASN and a *Cache Group*
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

### 201: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: /83P4LJVsrQx9BKHFxo5pbhQMLB4o3a9v3PpkspyOJcpNx1S/
→GJhCPpiANBki547sbY+0vTq76IriHZ4GYp8bA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 08 Nov 2018 14:37:39 GMT
Content-Length: 160

{ "alerts": [
  {
    "text": "asn was updated.",
    "level": "success"
  }
],
"response": {
  "asn": 2,
  "cachegroup": null,
```

(continues on next page)

(continued from previous page)

```
"cachegroupId": 1,
"id": 1,
"lastUpdated": "2018-11-08 14:37:39+00"
}}
```

## DELETE

Deletes an association between an ASN and a *Cache Group*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 99: Request Path Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the desired ASN-to- <i>Cache Group</i> as- sociation

## 202: Request Example

```
DELETE /api/3.0/asns/1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

## 203: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 23:06:24.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→6t3WA+DOcfPJB5UnvDpzEVx5ySfmJgEV9wgk071U5k32L1VXpxcaTdDVLNGgDD19sdNftmYnKXf5jpfWU
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 22:06:24 GMT
Content-Length: 81

{ "alerts": [
  {
    "text": "asn was deleted.",
    "level": "success"
  }
]}
```

## cache\_stats

Retrieves detailed, aggregated statistics for caches in a specific CDN.

### See also:

This gives an aggregate of statistics for *all caches* within a particular CDN and time range. For statistics basic statistics from all caches regardless of CDN and at the current time, use *caches/stats*.

## GET

Retrieves statistics about the caches within the CDN

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 100: Request Query Parameters

Name	Required	Description
cdnName	yes	The name of a CDN. Results will represent caches within this CDN
end-Date	yes	The date and time until which statistics shall be aggregated in <a href="#">RFC 3339</a> format (with or without sub-second precision), the number of nanoseconds since the Unix Epoch, or in the same, proprietary format as the <code>lastUpdated</code> fields prevalent throughout the Traffic Ops API
exclude	no	Either “series” to omit the data series from the result, or “summary” to omit the summary data from the result - directly corresponds to fields in the <a href="#">Response Structure</a>
interval	no	Specifies the interval within which data will be “bucketed”; e.g. when requesting data from 2019-07-25T00:00:00Z to 2019-07-25T23:59:59Z with an interval of “1m”, the resulting data series (assuming it is not excluded) should contain $24 \frac{\text{hours}}{\text{day}} \times 60 \frac{\text{minutes}}{\text{hour}} \times 1 \text{day} \times 1 \frac{\text{minute}}{\text{datapoint}} = 1440 \text{datapoints}$ The allowed values for this parameter are valid InfluxQL duration literal strings matching <code>^d+[mhdw]\$</code>
limit	no	A natural number indicating the maximum amount of data points should be returned in the <code>series</code> object
metric-Type	yes	The metric type being reported - one of: ‘connections’, ‘bandwidth’, ‘maxkpbs’
offset	no	A natural number of data points to drop from the beginning of the returned data set
orderby	no	Though one struggles to imagine why, this can be used to specify “time” to sort data points by their “time” (which is the default behavior)
start-Date	yes	The date and time from which statistics shall be aggregated in <a href="#">RFC 3339</a> format (with or without sub-second precision), the number of nanoseconds since the Unix Epoch, or in the same, proprietary format as the <code>lastUpdated</code> fields prevalent throughout the Traffic Ops API

### 204: Request Example

```
GET /api/3.0/cache_stats?cdnName=CDN&endDate=2019-10-28T20:49:00Z&
→metricType=bandwidth&startDate=2019-10-28T20:45:00Z HTTP/1.1
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: application/json;timestamp=unix, application/json;
→timestamp=rfc;q=0.9, application/json;q=0.8, */*;q=0.7
Connection: keep-alive
Cookie: mojolicious=...
```

### Content Format

It's important to note in *Request Example* the use of a complex “Accept” header. This endpoint accepts two special media types in the “Accept” header that instruct it on how to format the timestamps associated with the returned data. Specifically, Traffic Ops will recognize the special, optional, non-standard parameter of *application/json: timestamp*. The values of this parameter are restricted to one of

**rfc** Returned timestamps will be formatted according to **RFC 3339** (no sub-second precision).

**unix** Returned timestamps will be formatted as the number of nanoseconds since the Unix Epoch (midnight on January 1<sup>st</sup> 1970 UTC).

---

#### Implementation Detail

The endpoint passes back nanoseconds, specifically, because that is the form used both by InfluxDB, which is used to store the data being served, and Go's standard library. Clients may need to convert the value to match their own standard libraries - e.g. the `Date()` class in Javascript expects milliseconds.

---

The default behavior - when only e.g. *application/json* or *\*/\** is given - is to use **RFC 3339** formatting. It will, however, respect quality parameters. It is suggested that clients request timestamps they can handle specifically, rather than relying on this default behavior, as it is **subject to change** and is in fact **expected to invert in the next major release** as string-based time formats become deprecated.

#### See also:

For more information on the “Accept” HTTP header, consult [its dedicated page on MDN](#).

## Response Structure

**series** An object containing the actual data series and information necessary for working with it.

**columns** This is an array of names of the columns of the data contained in the “values” array - should always be `["time", "sum_count"]`

**count** The number of data points contained in the “values” array

**name** The name of the data set. Should always match `metric.ds.1min` where `metric` is the requested `metricType`

**values** The actual array of data points. Each represents a length of time specified by the `interval` query parameter

**time** The time at which the measurement was taken. This corresponds to the *beginning* of the interval. This time comes in the format of either an [RFC 3339](#)-formatted string, or a number containing the number of nanoseconds since the Unix Epoch depending on the “Accept” header sent by the client, according to the rules outlined in [Content Format](#).

**value** The value of the requested `metricType` at the time given by `time`. This will always be a floating point number, unless no data is available for the data interval, in which case it will be `null`

**summary** A summary of the data contained in the “series” object

**average** The arithmetic mean of the data’s values

**count** The number of measurements taken within the requested timespan. This is, in general, **not** the same as the `count` field of the `series` object, as it reflects the number of underlying, un-“bucketed” data points, and is therefore dependent on the implementation of Traffic Stats.

**fifthPercentile** Data points with values less than or equal to this number constitute the “bottom” 5% of the data set

**max** The maximum value that can be found in the requested data set

**min** The minimum value that can be found in the requested data set

**ninetyEighthPercentile** Data points with values greater than or equal to this number constitute the “top” 2% of the data set

**ninetyFifthPercentile** Data points with values greater than or equal to this number constitute the “top” 5% of the data set

## 205: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: p4asf1n7fXGtgpW/
    ↳dWgolJWdXjwDcCjyvJOPFqkckbgoXGUHEj5/
    ↳wlz7brlQ48t3Zn0WCq0lbsu2eSiBssBtUQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 28 Oct 2019 20:49:51 GMT

{ "response": {
  "series": {
    "columns": [
      "time",
      "sum_count"
    ],
    "count": 4,
    "name": "bandwidth.cdn.1min",
    "tags": {
      "cdn": "CDN-in-a-Box"
    },
    "values": [
      [
        1572295500000000000,
        null
      ],
      [
        1572295560000000000,
        113.66666666666666
      ],
      [
        1572295620000000000,
        108.83333333333334
      ]
    ]
  }
}
```

(continues on next page)



(continued from previous page)

```

        ],
        [
            15722956800000000000,
            113
        ]
    ],
    },
    "summary": {
        "average": 111.83333333333333,
        "count": 3,
        "fifthPercentile": 0,
        "max": 113.66666666666666,
        "min": 108.83333333333334,
        "ninetyEighthPercentile": 113.66666666666666,
        "ninetyFifthPercentile": 113.66666666666666
    }
}
}

```

### cachegroupparameters

Deprecated since version ATCv6.

#### GET

Extract information about the *Parameters* associated with *Cache Groups*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Response Structure

Table 101: Request Query Parameters

Name	Required	Description
cache-group	no	Show only the <i>Parameters</i> with the <i>Cache Group</i> identified by this integral, unique identifier
parameter	no	Show only the <i>Parameters</i> with the <i>Parameter</i> identified by this integral, unique identifier
order-by	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## Response Structure

**cacheGroupParameters** An array of identifying information for the *Parameters* of *Cache Groups*

**cacheGroup** A string containing the *Name* of the *Cache Group*

**last\_updated** Date and time of last modification in *Traffic Ops’s Custom Date/Time Format*

**parameter** An integer that is the *Parameter’s ID*

### 206: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 18:24:12 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
➔GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Vary: Accept-Encoding
Whole-Content-Sha512: PZyh09NeYYy4sXSv+Bfov0v32EuEk/1y7/
↪B+4fyvhbcPxWQ650NXBDpAe8IsmYZQYVRB03xlBtc33bo3Ixunbg==
Content-Length: 124

{ "response": {
  "cachegroupParameters": [
    {
      "parameter": 124,
      "last_updated": "2018-11-14 18:23:40.488853+00",
      "cachegroup": "test"
    }
  ]
}}
```

Deprecated since version ATCv6.

## POST

Assign *Parameter*(s) to *Cache Group*(s).

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Array

## Request Structure

This endpoint accepts two formats for the request payload:

**Single Object Format** For assigning a single *Parameter* to a single *Cache Group*

**Array Format** For making multiple assignments of *Parameters* to *Cache Groups* simultaneously

## Single Object Format

**cacheGroupId** An integer that is the *ID* of the *Cache Group* to which a *Parameter* is being assigned

**parameterId** An integer that is the *ID* of the *Parameter* being assigned

207: Request Example

```
POST /api/3.0/cacheGroupParameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 39
Content-Type: application/json

{
  "cacheGroupId": 8,
  "parameterId": 124
}
```

## Array Format

**cacheGroupId** An integer that is the *ID* of the *Cache Group* to which a *Parameter* is being assigned

**parameterId** An integer that is the *ID* of the *Parameter* being assigned

208: Request Example

```
POST /api/3.0/cacheGroupParameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 39
Content-Type: application/json

[
  {
    "cacheGroupId": 8,
    "parameterId": 124
  },
  {
    "cacheGroupId": 8,
    "parameterId": 125
  }
]
```

## Response Structure

**cacheGroupId** An integer that is the *ID* of the *Cache Group* to which a *Parameter* has been assigned

**parameterId** An integer that is the *ID* of the *Parameter* which has been assigned

### 209: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 15:47:49 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: wCv388wFaSjgFLCnI9dchlcyGxaVr8IhBAG25F+rpI2/
→azCswEYTcVBSlYOg6NxTQRzGkluMvn67jI6rV+vNsQ==
Content-Length: 136

{ "alerts": [
  {
    "level": "success",
    "text": "Cachegroup parameter associations were created."
  }
],
"response": [
  {
    "cacheGroupId": 8,
    "parameterId": 124
  }
]}
```

`cachegroupparameters/{ID}/{parameterID}`

Deprecated since version ATCv6.

## DELETE

Dissociate a *Parameter* with a *Cache Group*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 102: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Cache Group</i> which will have the <i>Parameter</i> association deleted
parameterID	The <i>ID</i> of the <i>Parameter</i> which will be removed from a <i>Cache Group</i>

### 210: Request Example

```
DELETE /api/3.0/cachegroupparameters/8/124 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 211: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 18:26:40 GMT
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→Cuj+ZPAKsDLp4FpbJDCwsWY0yVQAi1Um1CWraeTIQEMlyJSBEm17oKQWDjzTrvqqV8Prhu3gzlcHoVPzE
Content-Length: 84

{ "alerts": [
  {
    "level": "success",
    "text": "cachegroup parameter was deleted."
  }
]}
```

## cachegroups

### GET

Extract information about *Cache Groups*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 103: Request Query Parameters

Name	Re-quired	Description
id	no	Return the only <i>Cache Group</i> that has this id
name	no	Return only the <i>Cache Group</i> identified by this <i>Name</i>
type	no	Return only <i>Cache Groups</i> that are of the <i>Type</i> identified by this integral, unique identifier
topology	no	Return only <i>Cache Groups</i> that are used in the <i>Topology</i> identified by this unique identifier
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 212: Request Example

```
GET /api/3.0/cachegroups?type=23 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**fallbacks** An array of strings that are *Cache Group names* that are registered as *Fallbacks* for this *Cache Group*<sup>1</sup>

**fallbackToClosest** A boolean value that defines the *Fallback to Closest* behavior of this *Cache Group*<sup>Page 394, 1</sup>

**id** An integer that is the *ID* of the *Cache Group*

**lastUpdated** The time and date at which this entry was last updated in *Traffic Ops's Custom Date/Time Format*

<sup>1</sup> Traffic Router will first check for a fallbacks array and, when that is empty/unset/all the *Cache Groups* in it are also unavailable, will subsequently check for fallbackToClosest. If that is true, then it falls back to the geographically closest *Cache Group* capable of serving the same content or, when it is false/no such *Cache Group* exists/said *Cache Group* is also unavailable, will respond to clients with a failure response indicating the problem.



**latitude** A floating-point *Latitude* for the *Cache Group*

**localizationMethods** An array of *Localization Methods* as strings

**longitude** A floating-point *Longitude* for the *Cache Group*

**name** A string containing the *Name* of the *Cache Group*

**parentCachegroupId** An integer that is the *ID* of this *Cache Group*'s *Parent* - or null if it doesn't have a *Parent*

**parentCachegroupName** A string containing the *Name* of this *Cache Group*'s *Parent* - or null if it doesn't have a *Parent*

**secondaryParentCachegroupId** An integer that is the *ID* of this *Cache Group*'s *Secondary Parent* - or null if it doesn't have a *Secondary Parent*

**secondaryParentCachegroupName** A string containing the *Name* of this *Cache Group*'s *Secondary Parent Cache Group* - or null if it doesn't have a *Secondary Parent*

**shortName** A string containing the *Short Name* of the *Cache Group*

**typeId** An integral, unique identifier for the '*Type*' of the *Cache Group*

**typeName** A string that names the *Type* of this *Cache Group*

**Note:** The default value of `fallbackToClosest` is 'true', and if it is 'null' Traffic Control components will still interpret it as 'true'.

### 213: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳oV6ifEgoFy+v049tVjSsRdWQf4bxjrUvIYfDdgpUtlxiC7gzCv31m5bXQ8EUBW4eg2hfYM+BsGvJpnNDZ
X-Server-Name: traffic_ops_golang/
Date: Wed, 07 Nov 2018 19:46:36 GMT
Content-Length: 379

{ "response": [
  {
    "id": 7,
    "name": "CDN_in_a_Box_Edge",
```

(continues on next page)

(continued from previous page)

```

    "shortName": "ciabEdge",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "parentCachegroupName": "CDN_in_a_Box_Mid",
    "parentCachegroupId": 6,
    "secondaryParentCachegroupName": null,
    "secondaryParentCachegroupId": null,
    "fallbackToClosest": [],
    "localizationMethods": [],
    "typeName": "EDGE_LOC",
    "typeId": 23,
    "lastUpdated": "2018-11-07 14:45:43+00",
    "fallbacks": []
  }
}]

```

## POST

Creates a *Cache Group*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**fallbacks** An optional field which, when present, should contain an array of strings that are the *Names* of other *Cache Groups* which will be the *Fallbacks*<sup>Page 394, 1</sup>

**fallbackToClosest** A boolean that sets the *Fallback to Closest* behavior of the *Cache Group*<sup>Page 394, 1</sup>

---

**Note:** The default value of `fallbackToClosest` is `true`, and if it is `null` Traffic Control components will still interpret it as though it were `true`.

---

**latitude** An optional field which, if present, should be a floating-point number that will define the *Latitude* for the *Cache Group*<sup>2</sup>

**localizationMethods** Array of *Localization Methods* (as strings)

---

<sup>2</sup> While these fields are technically optional, note that if they are not specified many things may break. For this reason, Traffic Portal requires them when creating or editing *Cache Groups*.

---

**Tip:** This field has no defined meaning if the *Type* identified by `typeId` is not “EDGE\_LOC”.

---

**longitude** An optional field which, if present, should be a floating-point number that will define the *Longitude* for the *Cache Group*?

**name** The *Name* of the *Cache Group*

**parentCachegroupId** An optional field which, if present, should be an integer that is the *ID* of a *Parent* for this *Cache Group*.

**secondaryParentCachegroupId** An optional field which, if present, should be an integral, unique identifier for this *Cache Group*’s secondary parent

**shortName** An abbreviation of the name

**typeId** An integral, unique identifier for the *Cache Group*’s *Type*

---

**Note:** The actual, integral, unique identifiers for these *Types* must first be obtained, generally via *types*.

---

#### 214: Request Example

```
POST /api/3.0/cacheGroups HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 252
Content-Type: application/json

{
  "name": "test",
  "shortName": "test",
  "latitude": 0,
  "longitude": 0,
  "fallbackToClosest": true,
  "localizationMethods": [
    "DEEP_CZ",
    "CZ",
    "GEO"
  ],
  "typeId": 23,
}
```

## Response Structure

- fallbacks** An array of strings that are *Cache Group names* that are registered as *Fallbacks* for this *Cache Group*<sup>Page 394, 1</sup>
- fallbackToClosest** A boolean value that defines the *Fallback to Closest* behavior of this *Cache Group*<sup>Page 394, 1</sup>
- id** An integer that is the *ID* of the *Cache Group*
- lastUpdated** The time and date at which this entry was last updated in *Traffic Ops's Custom Date/Time Format*
- latitude** A floating-point *Latitude* for the *Cache Group*
- localizationMethods** An array of *Localization Methods* as strings
- longitude** A floating-point *Longitude* for the *Cache Group*
- name** A string containing the *Name* of the *Cache Group*
- parentCachegroupId** An integer that is the *ID* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*
- parentCachegroupName** A string containing the *Name* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*
- secondaryParentCachegroupId** An integer that is the *ID* of this *Cache Group's Secondary Parent* - or null if it doesn't have a *Secondary Parent*
- secondaryParentCachegroupName** A string containing the *Name* of this *Cache Group's Secondary Parent Cache Group* - or null if it doesn't have a *Secondary Parent*
- shortName** A string containing the *Short Name* of the *Cache Group*
- typeId** An integral, unique identifier for the '*Type*' of the *Cache Group*
- typeName** A string that names the *Type* of this *Cache Group*

### 215: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→YvZlh3rpf13nBq6SbNVhbkt3IvckbB9amqGW2JhLxWK9K3cxjBq5J2sIHBUhrLKUhE9afpxtvaYrLRxjt
→YMQ==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Wed, 07 Nov 2018 22:11:50 GMT
Content-Length: 379

{ "alerts": [
  {
    "text": "cachegroup was created.",
    "level": "success"
  }
],
"response": {
  "id": 8,
  "name": "test",
  "shortName": "test",
  "latitude": 0,
  "longitude": 0,
  "parentCachegroupName": null,
  "parentCachegroupId": null,
  "secondaryParentCachegroupName": null,
  "secondaryParentCachegroupId": null,
  "fallbackToClosest": true,
  "localizationMethods": [
    "DEEP_CZ",
    "CZ",
    "GEO"
  ],
  "typeName": "EDGE_LOC",
  "typeId": 23,
  "lastUpdated": "2019-12-02 22:21:08+00",
  "fallbacks": []
}}
```

`cachegroups/{ID}`

**PUT**

Update *Cache Group*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 104: Request Path Parameters

Parameter	Description
ID	The <i>ID</i> of a <i>Cache Group</i>

**fallbacks** An optional field which, when present, should contain an array of strings that are the *Names* of other *Cache Groups* which will be the *Fallbacks*<sup>1</sup>

**fallbackToClosest** A boolean that sets the *Fallback to Closest* behavior of the *Cache Group*<sup>Page 394, 1</sup>

---

**Note:** The default value of `fallbackToClosest` is `true`, and if it is `null` Traffic Control components will still interpret it as though it were `true`.

---

**latitude** An optional field which, if present, should be a floating-point number that will define the *Latitude* for the *Cache Group*<sup>2</sup>

**localizationMethods** Array of *Localization Methods* (as strings)

---

**Tip:** This field has no defined meaning if the *Type* identified by `typeId` is not “EDGE\_LOC”.

---

**longitude** An optional field which, if present, should be a floating-point number that will define the *Longitude* for the *Cache Group*<sup>?</sup>

**name** The *Name* of the *Cache Group*

**parentCachegroupId** An optional field which, if present, should be an integer that is the *ID* of a *Parent* for this *Cache Group*.

**secondaryParentCachegroupId** An optional field which, if present, should be an integral, unique identifier for this *Cache Group*’s secondary parent

**shortName** An abbreviation of the name

**typeId** An integral, unique identifier for the *Cache Group*’s *Type*

---

**Note:** The actual, integral, unique identifiers for these *Types* must first be obtained, generally via *types*.

---



---

<sup>1</sup> Traffic Router will first check for a `fallbacks` array and, when that is empty/unset/all the *Cache Groups* in it are also unavailable, will subsequently check for `fallbackToClosest`. If that is `true`, then it falls back to the geographically closest *Cache Group* capable of serving the same content or, when it is `false`/no such *Cache Group* exists/said *Cache Group* is also unavailable, will respond to clients with a failure response indicating the problem.

<sup>2</sup> While these fields are technically optional, note that if they are not specified many things may break. For this reason, Traffic Portal requires them when creating or editing *Cache Groups*.

## 216: Request Example

```
PUT /api/3.0/cachegroups/8 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 118
Content-Type: application/json

{
  "latitude": 0.0,
  "longitude": 0.0,
  "name": "test",
  "fallbacks": [],
  "fallbackToClosest": true,
  "shortName": "test",
  "typeId": 23,
  "localizationMethods": ["GEO"]
}
```

## Response Structure

**fallbacks** An array of strings that are *Cache Group names* that are registered as *Fallbacks* for this *Cache Group*<sup>Page 394, 1</sup>

**fallbackToClosest** A boolean value that defines the *Fallback to Closest* behavior of this *Cache Group*<sup>Page 394, 1</sup>

**id** An integer that is the *ID* of the *Cache Group*

**lastUpdated** The time and date at which this entry was last updated in *Traffic Ops's Custom Date/Time Format*

**latitude** A floating-point *Latitude* for the *Cache Group*

**localizationMethods** An array of *Localization Methods* as strings

**longitude** A floating-point *Longitude* for the *Cache Group*

**name** A string containing the *Name* of the *Cache Group*

**parentCachegroupId** An integer that is the *ID* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*

**parentCachegroupName** A string containing the *Name* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*

**secondaryParentCachegroupId** An integer that is the *ID* of this *Cache Group's Secondary Parent* - or null if it doesn't have a *Secondary Parent*

**secondaryParentCachegroupName** A string containing the *Name* of this *Cache Group's Secondary Parent Cache Group* - or `null` if it doesn't have a *Secondary Parent*

**shortName** A string containing the *Short Name* of the *Cache Group*

**typeId** An integral, unique identifier for the '*Type*' of the *Cache Group*

**typeName** A string that names the *Type* of this *Cache Group*

## 217: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: t1W65/2kj25QyHt0Ib0xpBaAR2sXu2k0sRZ49WjKZp/
    ↳AK5S1YWhX7VNWCUUGiN1VNM4QRNqODC/7ewhYDFUncA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 19:14:28 GMT
Content-Length: 385

{ "alerts": [
  {
    "text": "cachegroup was updated.",
    "level": "success"
  }
],
"response": {
  "id": 8,
  "name": "test",
  "shortName": "test",
  "latitude": 0,
  "longitude": 0,
  "parentCachegroupName": null,
  "parentCachegroupId": null,
  "secondaryParentCachegroupName": null,
  "secondaryParentCachegroupId": null,
  "fallbacks": [],
  "fallbackToClosest": true,
  "localizationMethods": [
    "GEO"
  ],
  "typeName": "EDGE_LOC",
```

(continues on next page)



(continued from previous page)

```
"typeId": 23,
"lastUpdated": "2018-11-14 19:14:28+00"
}}
```

## DELETE

Delete a *Cache Group*. A *Cache Group* which has assigned servers or is the *Parent* of one or more other *Cache Groups* cannot be deleted.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 105: Request Path Parameters

Parameter	Description
ID	The <i>ID</i> of a <i>Cache Group</i> to be deleted

### 218: Request Example

```
DELETE /api/3.0/cacheGroups/42 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 219: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 5jZBg07h1eNF70J/cmlbi3Hf9KJPx+WLMb1H/pSKF3FWb/
→10GUHIN35ZOB+1N5LZYCkmk3izGbTFkiruG8I41Q==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:31:04 GMT
Content-Length: 57

{ "alerts": [
  {
    "text": "cachegroup was deleted.",
    "level": "success"
  }
]}
```

## cachegroups/{{ID}}/deliveryservices

### POST

Assigns all of the “assignable” servers within a *Cache Group* to one or more *Delivery Services*.

---

**Note:** “Assignable” here means all of the *Cache Group’s servers* that have a *Type* that matches one of the glob patterns `EDGE*` or `ORG*`. If even one server of any *Type* exists within the *Cache Group* that is not assigned to the same CDN as the *Delivery Service* to which an attempt is being made to assign them, the request will fail.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

**deliveryServices** The integral, unique identifiers of the *Delivery Services* to which the *Cache Group’s servers* are being assigned

220: Request Example

```
POST /api/3.0/cachegroups/8/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 25
Content-Type: application/json

{"deliveryServices": [2]}
```

## Response Structure

**deliveryServices** An array of integral, unique identifiers for *Delivery Services* to which the *Cache Group's servers* have been assigned

**id** An integer that is the *Cache Group's ID*

**serverNames** An array of the (short) hostnames of all of the *Cache Group's* “assignable” *Servers*

### 221: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: j/yH0gvJoaGjiLZU/
→0MA8o5He2004aJ5wh1eF9ex6F6IB01liM9Wk9RkWCw7sdiUHoy13/
→mf7gDntisZwzP7yw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 19:54:17 GMT
Content-Length: 183

{ "alerts": [
  {
    "text": "Delivery services successfully assigned to all the
→servers of cache group 8.",
    "level": "success"
  }
],
"response": {
  "id": 8,
  "serverNames": [
    "foo"
  ],
  "deliveryServices": [
    2
  ]
}}
```

## cachegroups/{ID}/parameters

Deprecated since version ATCv6.

### GET

Gets all of a *Cache Group's parameters*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

Table 106: Request Query Parameters

Name	Re-quired	Description
parameterId	no	Show only the <i>Parameter</i> with the given <i>ID</i>
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <b>response</b> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <b>limit</b>
page	no	Return the <i>n<sup>th</sup></i> page of results, where “n” is the value of this parameter, pages are <b>limit</b> long and the first page is 1. If <b>offset</b> was defined, this query parameter has no effect. <b>limit</b> must be defined to make use of <b>page</b> .

Table 107: Request Path Parameters

Parameter	Description
ID	The <i>ID</i> of a <i>Cache Group</i>

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name of the Parameter*

**secure** A boolean value describing whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 222: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 19:56:23 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: DfqPtySzVMpnBYqVt/45sSRG/
→1pRTlQdIcYuQZ0CQt79QSHLzU5e4TbDqht6ntvNP041LimKsj5RzPlPX1n6tg==
Content-Length: 135

{ "response": [
  {
    "lastUpdated": "2018-11-14 18:22:43.754786+00",
    "value": "foobar",
    "secure": false,
    "name": "foo",
    "id": 124,
    "configFile": "bar"
  }
]}
```

## cachegroups/{ID}/queue\_update

### POST

*Queue* or “dequeue” updates for all of a *Cache Group’s servers*, limited to a specific CDN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

Table 108: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Cache Group</i> for which updates are being <i>Queued</i> /dequeued

**action** The action to perform; one of “queue” or “dequeue”

**cdn** The full name of the CDN in need of *Queue Updates*, or a “dequeue” thereof<sup>1</sup>

**cdnId** The integral, unique identifier for the CDN in need of *Queue Updates*, or a “dequeue” thereof<sup>Page 394, 1</sup>

### 223: Request Example

```
POST /api/3.0/cachegroups/8/queue_update HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 42
Content-Type: application/json

{"action": "queue", "cdn": "CDN-in-a-Box"}
```

### Response Structure

**action** The action processed, one of “queue” or “dequeue”

**cachegroupId** An integer that is the *ID* of the *Cache Group* for which *Queue Updates* was performed or cleared

**cachegroupName** The name of the *Cache Group* for which updates were queued/dequeued

**cdn** The name of the CDN to which the queue/dequeue operation was restricted

<sup>1</sup> Either ‘cdn’ or ‘cdnID’ *must* be in the request data (but not both).

**serverNames** An array of the (short) hostnames of the *Cache Group's servers* which are also assigned to the CDN specified in the "cdn" field

#### 224: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→UAcP7LrflU1RnfR4UqbQrJczlk5rkrcL0tTXJTFvIUxxK1EklZkHkE4vewjDaVIhJJ6YQg8jmPGQpr+x1
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:19:46 GMT
Content-Length: 115

{ "response": {
  "cachegroupName": "test",
  "action": "queue",
  "serverNames": [
    "foo"
  ],
  "cdn": "CDN-in-a-Box",
  "cachegroupID": 8
}}
```

### caches/stats

An API endpoint that returns cache statistics using the *Traffic Monitor APIs*.

#### See also:

This gives a set of basic statistics for *all cache servers* at the current time. For statistics from time ranges and/or aggregated over a specific CDN, use *cache\_stats*.

### GET

Retrieves cache stats from Traffic Monitor. Also includes rows for aggregates.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

No parameters available.

### Response Structure

**cachegroup** A string that is the *Name* of the *Cache Group* to which this *cache server* belongs

**connections** Current number of TCP connections maintained by the *cache server*

**healthy** true if Traffic Monitor has marked the *cache server* as “healthy”, false otherwise

**See also:**

*Health Protocol*

**hostname** The (short) hostname of the *cache server*

**ip** The IP address of the *cache server*

**kbps** The *cache server*’s upload speed (to clients) in Kilobits per second

**profile** The *Name* of the *Profile* in use by this *cache server*

**status** The status of the *cache server*

225: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 20:25:01 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
↪GMT; Max-Age=3600; HttpOnly
```

(continues on next page)



(continued from previous page)

```
Vary: Accept-Encoding
Whole-Content-Sha512: DqbLgitanS8q81/
→qKC1i+ImMiEMF+SW4G9rb79FWdeWcgwFjL810t1TRp1nNNfHV+tajgjyK+wMHobqVyaNEfA==
Content-Length: 133

{ "response": [
  {
    "profile": "ALL",
    "connections": 0,
    "ip": null,
    "status": "ALL",
    "healthy": true,
    "kbps": 0,
    "hostname": "ALL",
    "cachegroup": "ALL"
  }
]}
```

## capabilities

Deprecated since version 3.1.

### GET

Get all capabilities.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 109: Request Query Parameters

Name	Required	Description
name	no	Return only the capability that has this name
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 226: Request Example

```
GET /api/3.0/capabilities?name=test HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**name** Name of the capability

**description** Describes the permissions covered by the capability.

**lastUpdated** Date and time of the last update made to this capability, in *Traffic Ops’s Custom Date/Time Format*

### 227: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Vary: Accept-Encoding
Transfer-Encoding: chunked
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: c18+GtX2ZI8PoCSwuAzBhl+6w3vDpKQTa/
→cDJC0WHxdpguOL378KBxGWW5PCSyZfJUUb7wPyOL5qKMn6NNTufhg==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Aug 2019 17:20:20 GMT
Content-Length: 161

{ "response": [
  {
    "description": "This is only a test. If this were a real_
→capability, it might do something",
    "lastUpdated": "2019-08-15 17:18:03+00",
    "name": "test"
  }
]}
```

## cdns

Extract information about all CDNs

### GET

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 110: Request Query Parameters

Parameter	Required	Description
domainName	no	Return only the CDN that has this domain name
dnssecEnabled	no	Return only the CDNs that are either dnssec enabled or not
id	no	Return only the CDN that has this id
name	no	Return only the CDN that has this name
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

## Response Structure

**dnssecEnabled** true if DNSSEC is enabled on this CDN, otherwise false

**domainName** Top Level Domain name within which this CDN operates

**id** The integral, unique identifier for the CDN

**lastUpdated** Date and time when the CDN was last modified in *Traffic Ops’s Custom Date/Time Format*

**name** The name of the CDN

### 228: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→z9P1NkxGebPncUhaChDHtYKYI+XVZfhE6Y84TuwoASZFIMfISELwADLpvpPTN+wwnzBfREksLYn+0313Q
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:46:57 GMT
Content-Length: 237
```

```
{ "response": [
  {
    "dnssecEnabled": false,
    "domainName": "-",
    "id": 1,
    "lastUpdated": "2018-11-14 18:21:06+00",
    "name": "ALL"
  },
  {
    "dnssecEnabled": false,
    "domainName": "mycdn.ciab.test",
    "id": 2,
    "lastUpdated": "2018-11-14 18:21:14+00",
    "name": "CDN-in-a-Box"
  }
]}
```

## POST

Allows user to create a CDN

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**dnssecEnabled** If true, this CDN will use DNSSEC, if false it will not

**domainName** The top-level domain (TLD) belonging to the new CDN

**name** Name of the new CDN

229: Request Structure

```
POST /api/3.0/cdns HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)

(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 63
Content-Type: application/json

{"name": "test", "domainName": "quest", "dnssecEnabled": false}
```

## Response Structure

**dnssecEnabled** true if the CDN uses DNSSEC, false otherwise

**domainName** The top-level domain (TLD) assigned to the newly created CDN

**id** An integral, unique identifier for the newly created CDN

**name** The newly created CDN's name

### 230: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳1rZRl0fQioGRrEb4nCfjGGx7y3Ub2h7BZ4z6NbhcY4acPslKSUNM8QLjWTVwLU4WpkfJNxc0yy8NlKULF
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:49:28 GMT
Content-Length: 174

{ "alerts": [
  {
    "text": "cdn was created.",
    "level": "success"
  }
],
"response": {
  "dnssecEnabled": false,
  "domainName": "quest",
  "id": 3,
  "lastUpdated": "2018-11-14 20:49:28+00",
```

(continues on next page)

(continued from previous page)

```
"name": "test"
}}
```

## cdns/capacity

### GET

Retrieves the aggregate capacity percentages of all *Cache Groups* for a given CDN.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

### Request Structure

No parameters available.

### Response Structure

**availablePercent** The percent of available (unused) bandwidth to 64 bits of precision<sup>1</sup>

**unavailablePercent** The percent of unavailable (used) bandwidth to 64 bits of precision<sup>1</sup>

**utilizedPercent** The percent of bandwidth currently in use to 64 bits of precision<sup>1</sup>

**maintenancePercent** The percent of bandwidth being used for administrative or analytical processes internal to the CDN to 64 bits of precision<sup>1</sup>

#### 231: Response Example

```
{ "response": {
  "availablePercent": 89.0939840205533,
  "unavailablePercent": 0,
  "utilizedPercent": 10.9060020300395,
  "maintenancePercent": 0.0000139494071146245
}}
```

<sup>1</sup> Following IEEE 754

### cdns/dnsseckey/generate

#### POST

Generates ZSK and KSK keypairs for a CDN and all associated *Delivery Services*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object (string)

#### Request Structure

**effectiveDate** An optional string containing the date and time at which the newly-generated ZSK and KSK become effective, in **RFC 3339** format. Defaults to the current time if not specified.

**key** Name of the CDN

**kskExpirationDays** Expiration (in days) for the KSKs (Key-Signing Keys)

**ttl** Time, in seconds, for which the keypairs shall remain valid

**zskExpirationDays** Expiration (in days) for the ZSKs (Zone-Signing Keys)



### 232: Request Example

```
POST /api/3.0/cdns/dnsseckey/generate HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 130

{
  "key": "CDN-in-a-Box",
  "kskExpirationDays": 1095,
  "ttl": 3600,
  "zskExpirationDays": 1095
}
```

## Response Structure

### 233: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 19:42:15
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 09SPWzeMNFgg6I/PPeXittBIh3/
  ↳zUKK1NwNlYIM9SszSrK0h/Dfz7tnwgnA7h/s6M4eYBJxykDpCfVC7xpeg==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 18:42:15 GMT
Content-Length: 89

{
  "response": "Successfully created dnssec keys for CDN-in-a-Box"
}
```

### cdns/dnsseckey/refresh

#### GET

Refresh the DNSSEC keys for all CDNs. This call initiates a background process to refresh outdated keys, and immediately returns a response that the process has started.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object (string)

#### Request Structure

No parameters available

#### Response Structure

##### 234: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 21:37:30 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
  ↳Uw1+924m6Ye3NraFP+RBp1dkhcNTTDyXHZbzRaYV95p9tP56Z61gcKeKSr1oQIkNXjXcCsDN5Dmum7Zk1
Content-Length: 69

{
  "response": "Checking DNSSEC keys for refresh in the background"
}
```

## cdns/domains

### GET

Gets a list of domains and their related Traffic Router *Profiles* for all CDNs.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

No parameters available.

### Response Structure

**domainName** The TLD assigned to this CDN

**parameterId** The *ID* for the *Parameter* that sets this TLD on the Traffic Router

**profileDescription** A short, human-readable description of the Traffic Router's profile

**profileId** The *ID* of the *Profile* assigned to the Traffic Router responsible for serving domainName

**profileName** The *Name* of the *Profile* assigned to the Traffic Router responsible for serving domainName

## 235: Response Example

```
{ "response": [
  {
    "profileId": 12,
    "parameterId": -1,
    "profileName": "CCR_CIAB",
    "profileDescription": "Traffic Router for CDN-In-A-Box",
    "domainName": "mycdn.ciab.test"
  }
]}
```

### cdns/health

Extract health information from all *Cache Groups* across all CDNs

**See also:**

*Health Protocol*

**GET**

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

### Request Structure

No parameters available

### Response Structure

**cachegroups** An array of objects describing the health of each Cache Group

**name** The name of the Cache Group

**offline** The number of OFFLINE caches in the Cache Group

**online** The number of ONLINE caches in the Cache Group

**totalOffline** Total number of OFFLINE caches across all Cache Groups which are assigned to any CDN

**totalOnline** Total number of ONLINE caches across all Cache Groups which are assigned to any CDN

## 236: Response Example

```
{ "response": {
  "totalOffline": 0,
  "totalOnline": 1,
  "cachegroups": [
    {
      "offline": 0,
      "name": "CDN_in_a_Box_Edge",
      "online": 1
    }
  ]
}}
```

**cdns/{ID}**

**PUT**

Allows a user to edit a specific CDN

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 111: Request Path Parameters

Name	Description
ID	Integral, unique identifier for the CDN to update

**dnssecEnabled** If `true`, this CDN will use DNSSEC, if `false` it will not

**domainName** The top-level domain (TLD) belonging to the CDN

**name** Name of the new CDN

## 237: Request Example

```
PUT /api/3.0/cdns/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 63
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
```

```
{"name": "quest", "domainName": "test", "dnssecEnabled": false}
```

## Response Structure

**dnssecEnabled** true if the CDN uses DNSSEC, false otherwise

**domainName** The top-level domain (TLD) assigned to the newly created CDN

**id** An integral, unique identifier for the newly created CDN

**name** The newly created CDN's name

### 238: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: sI1hzBwG+/
    ↳VAzoFY20kqGFA2RgrU0ThtMeeJqk0ZxH3TRxTWuA8BetACct/
    ↳XICC3n7hPDLlRVpwckEyBdyJkXg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:54:33 GMT
Content-Length: 174

{ "alerts": [
  {
    "text": "cdn was updated.",
    "level": "success"
  }
],
"response": {
  "dnssecEnabled": false,
  "domainName": "test",
  "id": 4,
  "lastUpdated": "2018-11-14 20:54:33+00",
  "name": "quest"
}}
```

## DELETE

Allows a user to delete a specific CDN

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 112: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the CDN to delete

## Response Structure

239: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Zy4cJN6BEct4ltFLN4e296mM8XnzOs0EQ3/
    ↳jp4TA3L+g8qtkI0WrL+ThcFq4xbJPU+KHVDSi+b0JBav3xsYPqQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:51:23 GMT
Content-Length: 58

{ "alerts": [
  {
    "text": "cdn was deleted.",
    "level": "success"
  }
]}
```

`cdns/{ID}/queue_update`

POST

*Queue* or “dequeue” updates for all servers assigned to a specific CDN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 113: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the CDN on which to (de)queue updates

**action** One of “queue” or “dequeue” as appropriate

240: Request Example

```
POST /api/3.0/cdns/2/queue_update HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 19
Content-Type: application/json

{"action": "queue"}
```

## Response Structure

**action** The action processed, either “queue” or “dequeue”

**cdnId** The integral, unique identifier for the CDN on which *Queue Updates* was performed or cleared

241: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)



(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→rBpFfrrP+9IFkwsRloEM+v+I8MuBZDXqFu+WUTGtRGypnAn2gHooPoNQRYVvJGjyIQrLXLvqjEtve+lH2
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:02:07 GMT
Content-Length: 41

{ "response": {
  "action": "queue",
  "cdnId": 2
}}
```

`cdns/{{name}}/configs/monitoring`

**See also:**

*Health Protocol*

**GET**

Retrieves information concerning the monitoring configuration for a specific CDN.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 114: Request Path Parameters

Name	Description
name	The name of the CDN for which monitoring configuration will be fetched

## Response Structure

**cacheGroups** An array of objects representing each of the *Cache Groups* being monitored within this CDN

**coordinates** An object representing the geographic location of this *Cache Group*

**latitude** This *Cache Group's latitude* as a floating-point number

**longitude** This *Cache Group's longitude* as a floating-point number

**name** A string that is this *Cache Group's name*

**config** A collection of parameters used to configure the monitoring behaviour of Traffic Monitor

**health.polling.interval** An interval in milliseconds on which to poll for cache statistics

**heartbeat.polling.interval** An interval in milliseconds on which to poll for health statistics. If missing, defaults to `health.polling.interval`.

**tm.polling.interval** The interval at which to poll for configuration updates

**deliveryServices** An array of objects representing each *Delivery Service* provided by this CDN

**hostRegexes** An array of strings which are the Delivery Service's HOST\_REGEX-type regexes

**status** The *Delivery Service's* status

**topology** A string that is the name of the Delivery Service's *Topology* (if assigned one)

**totalKbpsThreshold** A threshold rate of data transfer this *Delivery Service* is configured to handle, in Kilobits per second

**totalTpsThreshold** A threshold amount of transactions per second that this *Delivery Service* is configured to handle

**type** A string that is the Delivery Service's type category ("HTTP" or "DNS")

**xmlId** A string that is the *Delivery Service's XMLID*

**profiles** An array of the *Profiles* in use by the *cache servers* and *Delivery Services* belonging to this CDN

**name** A string that is the *Profile's Name*

**parameters** An array of the *Parameters* in this *Profile* that relate to monitoring configuration. This can be null if the servers using this *Profile* cannot be monitored (e.g. Traffic Routers)

**health.connection.timeout** A timeout value, in milliseconds, to wait before giving up on a health check request

**health.polling.url** A URL to request for polling health. Substitutions can be made in a shell-like syntax using the properties of an object from the "trafficServers" array

**health.threshold.availableBandwidthInKbps**  
The total amount of bandwidth that servers using this profile are allowed - across all network interfaces - in Kilobits per second. This is a string and using comparison operators to specify ranges, e.g. ">10" means "more than 10 kbps"

**health.threshold.loadavg** The UNIX load-avg at which the server should be marked "unhealthy"

**See also:**

*uptime(1)*

**health.threshold.queryTime** The highest allowed length of time for completing health queries (after connection has been established) in milliseconds

**history.count** The number of past events to store; once this number is reached, the oldest event will be forgotten before a new one can be added

**type** A string that names the *Profile's Type*

**topologies** A map of *Topology* names to objects

**nodes** An array of strings which are the names of the EDGE\_LOC-type cache groups in the topology

**trafficMonitors** An array of objects representing each Traffic Monitor that monitors this CDN (this is used by Traffic Monitor's "peer polling" function)

**cachegroup** The name of the *Cache Group* to which this Traffic Monitor belongs

- fqdn** An FQDN that resolves to the IPv4 (and/or IPv6) address of the server running this Traffic Monitor instance
- hostname** The hostname of the server running this Traffic Monitor instance
- ip6** The IPv6 address of this Traffic Monitor - when applicable
- ip** The IPv4 address of this Traffic Monitor
- port** The port on which this Traffic Monitor listens for incoming connections
- profile** A string that is the *Name* of the *Profile* assigned to this Traffic Monitor
- status** The status of the server running this Traffic Monitor instance
- trafficServers** An array of objects that represent the *cache servers* being monitored within this CDN
- cachegroup** The *Cache Group* to which this *cache server* belongs
- deliveryServices** An array of objects which contain the XML IDs of the delivery services to which this cache server is assigned (this field is omitted entirely if no delivery services are assigned to this cache server)
- xmlId** A string which is the XML ID of the delivery service
- fqdn** An FQDN that resolves to the *cache server*'s IPv4 (or IPv6) address
- hashId** The (short) hostname for the *cache server* - named "hashId" for legacy reasons
- hostName** The (short) hostname of the *cache server*
- port** The port on which the *cache server* listens for incoming connections
- profile** A string that is the *Name* of the *Profile* assigned to this *cache server*
- status** The status of the *cache server*
- type** A string that names the *Type* of the *cache server* - should (ideally) be either EDGE or MID
- interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or null to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is true, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic

Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning “no limit”. It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface’s device name on the server, e.g. `eth0`.

242: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→uLR+tRoqR8SY038j3DV9wQ+IkJ7Kf+MCoFkcWZtsghpLJ+0S6f+IiI8laNVeDgrM/
→P23MAQ6BSepm+EJRl1AXQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:09:31 GMT
Transfer-Encoding: chunked

{ "response": {
  "topologies": {
    "example-topology": {
      "nodes": [
        "CDN_in_a_Box_Edge"
```

(continues on next page)

(continued from previous page)

```

    ]
  }
},
"trafficServers": [
  {
    "profile": "ATS_EDGE_TIER_CACHE",
    "status": "REPORTED",
    "port": 80,
    "interfaces": [
      {
        "ipAddresses": [
          {
            "address": "172.16.239.100",
            "gateway": "172.16.239.0/24",
            "serviceAddress": "true"
          },
          {
            "address": "fc01:9400:1000:8::100",
            "gateway": "fc01::",
            "serviceAddress": "true"
          }
        ],
        "name": "eth0",
        "monitor": "true",
        "mtu": 9000,
        "maxBandwidth": 150
      }
    ],
    "cachegroup": "CDN_in_a_Box_Edge",
    "hostname": "edge",
    "fqdn": "edge.infra.ciab.test",
    "type": "EDGE",
    "hashid": "edge"
  },
  {
    "profile": "ATS_MID_TIER_CACHE",
    "status": "REPORTED",
    "port": 80,
    "interfaces": [
      {
        "ipAddresses": [
          {
            "address": "172.16.239.120",
            "gateway": "172.16.239.0/24",
            "serviceAddress": "true"
          }
        ],

```

(continues on next page)

(continued from previous page)

```

        {
            "address": "fc02:9400:1000:8::100",
            "gateway": "fc02::",
            "serviceAddress": "true"
        }
    ],
    "name": "eth0",
    "monitor": "true",
    "mtu": 9000,
    "maxBandwidth": 150
}
],
"cacheGroup": "CDN_in_a_Box_Mid",
"hostname": "mid",
"fqdn": "mid.infra.ciab.test",
"type": "MID",
"hashid": "mid"
}
],
"trafficMonitors": [
{
    "profile": "RASCAL-Traffic_Monitor",
    "status": "ONLINE",
    "ip": "172.16.239.40",
    "ip6": "fc01:9400:1000:8::40",
    "port": 80,
    "cacheGroup": "CDN_in_a_Box_Edge",
    "hostname": "trafficmonitor",
    "fqdn": "trafficmonitor.infra.ciab.test"
}
],
"cacheGroups": [
{
    "name": "CDN_in_a_Box_Mid",
    "coordinates": {
        "latitude": 38.897663,
        "longitude": -77.036574
    }
},
{
    "name": "CDN_in_a_Box_Edge",
    "coordinates": {
        "latitude": 38.897663,
        "longitude": -77.036574
    }
}
]

```

(continues on next page)



(continued from previous page)

```

],
"profiles": [
  {
    "name": "CCR_CIAB",
    "type": "CCR",
    "parameters": null
  },
  {
    "name": "ATS_EDGE_TIER_CACHE",
    "type": "EDGE",
    "parameters": {
      "health.connection.timeout": 2000,
      "health.polling.url": "http://${hostname}/_astats?
→application=&inf.name=${interface_name}",
      "health.threshold.availableBandwidthInKbps": ">1750000
→",
      "health.threshold.loadavg": "25.0",
      "health.threshold.queryTime": 1000,
      "history.count": 30
    }
  },
  {
    "name": "ATS_MID_TIER_CACHE",
    "type": "MID",
    "parameters": {
      "health.connection.timeout": 2000,
      "health.polling.url": "http://${hostname}/_astats?
→application=&inf.name=${interface_name}",
      "health.threshold.availableBandwidthInKbps": ">1750000
→",
      "health.threshold.loadavg": "25.0",
      "health.threshold.queryTime": 1000,
      "history.count": 30
    }
  }
],
"deliveryServices": [
  {
    "xmlId": "example-ds",
    "totalTpsThreshold": 0,
    "status": "REPORTED",
    "totalKbpsThreshold": 0,
    "type": "DNS",
    "topology": "example-topology",
    "hostRegexes": [
      ".*\\.example-ds\\.\\.*)"

```

(continues on next page)

(continued from previous page)

```

    ]
  }
],
"config": {
  "health.polling.interval": 6000,
  "heartbeat.polling.interval": 3000,
  "peers.polling.interval": 3000,
  "tm.polling.interval": 2000
}
}}

```

`cdns/{name}/dnsseckey/ksk/generate`

**POST**

Generates a new KSK for a specific CDN.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object (string)

## Request Structure

Table 115: Request Path Parameters

Name	Required	Description
name	yes	The name of the CDN for which the KSK will be generated

**expirationDays** The integral number of days until the newly generated KSK expires

**effectiveDate** An optional string containing the date and time at which the newly generated KSK becomes effective, in **RFC 3339** format. Defaults to the current time if not specified

## Response Structure

243: Response Example

```

{ "response": "Successfully generated ksk dnssec keys for my-cdn-name" }

```

`cdns/{{name}}/federations`

**GET**

Retrieves a list of federations in use by a specific CDN.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 116: Request Path Parameters

Name	Description
name	The name of the CDN for which federations will be listed

Table 117: Request Query Parameters

Name	Description
id	Return only the federation that has this id
order-by	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## 244: Request Example

```
GET /api/3.0/cdns/CDN-in-a-Box/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cname** The Canonical Name (CNAME) used by the federation

**deliveryService** An object with keys that provide identifying information for the *Delivery Service* using this federation

**id** The integral, unique identifier for the *Delivery Service*

**xmlId** The *Delivery Service*'s uniquely identifying 'xml\_id'

**description** An optionally-present field containing a description of the field

**Note:** This key will only be present if the description was provided when the federation was created. Refer to the POST method of this endpoint to see how federations can be created.

**lastUpdated** The date and time at which this federation was last modified, in *Traffic Ops's Custom Date/Time Format*

**ttl** Time to Live (TTL) for the **cname**, in hours

## 245: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
  ↳SJA7G+7G5Kc0fCtnE3Dq5DCobWtGRUKSppiDkflZoG5+paq4E1aZGqUb6vGVsd+TpPg75MLlhyqfdfCHnl
  ↳g==
x-server-name: traffic_ops_golang/
content-length: 170
date: Wed, 05 Dec 2018 00:35:40 GMT
```

(continues on next page)

(continued from previous page)

```
{ "response": [
  {
    "id": 1,
    "cname": "test.quest.",
    "ttl": 48,
    "description": "A test federation",
    "lastUpdated": "2018-12-05 00:05:16+00",
    "deliveryService": {
      "id": 1,
      "xmlId": "demo1"
    }
  }
]}
```

## POST

Creates a new federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 118: Request Path Parameters

Name	Description
name	The name of the CDN for which a new federation will be created

**cname** The Canonical Name (CNAME) used by the federation

---

**Note:** The CNAME must end with a “.”

---

**description** An optional description of the federation

**ttl** Time to Live (TTL) for the name record used for **cname**

### 246: Request Example

```
POST /api/3.0/cdns/CDN-in-a-Box/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
```

(continues on next page)

(continued from previous page)

```
Cookie: mojolicious=...
Content-Length: 72
Content-Type: application/json

{
  "cname": "test.quest.",
  "ttl": 48,
  "description": "A test federation"
}
```

## Response Structure

**cname** The Canonical Name (CNAME) used by the federation

**description** An optionally-present field containing a description of the field

---

**Note:** This key will only be present if the description was provided when the federation was created

---

**lastUpdated** The date and time at which this federation was last modified, in *Traffic Ops's Custom Date/Time Format*

**ttl** Time to Live (TTL) for the **cname**, in hours

### 247: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
→rRsWAIhXzVlj8Hy+8aFjp4Jo1QGTK49m0N1AP5QDyyAZ1TfNIIdgtcgiuehu7FiN1IPWRFiv6D9CygFYKG
x-server-name: traffic_ops_golang/
content-length: 192
date: Wed, 05 Dec 2018 00:05:16 GMT

{ "alerts": [
  {
    "text": "cdnfederation was created.",
    "level": "success"
  }
]}
```

(continues on next page)

(continued from previous page)

```
[,
  "response": {
    "id": 1,
    "cname": "test.quest.",
    "ttl": 48,
    "description": "A test federation",
    "lastUpdated": "2018-12-05 00:05:16+00"
  }
}]
```

`cdns/{{name}}/federations/{{ID}}`

**PUT**

Updates a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 119: Request Path Parameters

Name	Description
name	The name of the CDN for which the federation identified by ID will be inspected
ID	An integral, unique identifier for the federation to be inspected

**cname** The Canonical Name (CNAME) used by the federation

---

**Note:** The CNAME must end with a “.”

---

**description** An optional description of the federation

**ttl** Time to Live (TTL) for the name record used for **cname**

### 248: Request Example

```
PUT /api/3.0/cdns/CDN-in-a-Box/federations/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 33
```

(continues on next page)

(continued from previous page)

**Content-Type:** application/json

```
{
  "cname": "foo.bar.",
  "ttl": 48
}
```

## Response Structure

**cname** The Canonical Name (CNAME) used by the federation

**description** An optionally-present field containing a description of the field

---

**Note:** This key will only be present if the description was provided when the federation was created

---

**lastUpdated** The date and time at which this federation was last modified, in *Traffic Ops's Custom Date/Time Format*

**ttl** Time to Live (TTL) for the cname, in hours

### 249: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
↳qcjfq+gDjNxYQ1aq+dlddgrkFWnkFYxsFF+SHDqqH0uVHBVksmU0aTFgltozek/
↳u6wbrGoR1LFf9Fr1C1SbigA==
x-server-name: traffic_ops_golang/
content-length: 174
date: Wed, 05 Dec 2018 01:03:40 GMT

{ "alerts": [
  {
    "text": "cdnfederation was updated.",
    "level": "success"
  }
],
"response": {
```

(continues on next page)



(continued from previous page)

```
"id": 1,
"cname": "foo.bar.",
"ttl": 48,
"description": null,
"lastUpdated": "2018-12-05 01:03:40+00"
}}
```

## DELETE

Deletes a specific federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** undefined

## Request Structure

Table 120: Request Path Parameters

Name	Description
name	The name of the CDN for which the federation identified by ID will be inspected
ID	An integral, unique identifier for the federation to be inspected

### 250: Request Example

```
DELETE /api/3.0/cdns/CDN-in-a-Box/federations/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 251: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
```

(continues on next page)

(continued from previous page)

```
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512: Cnkfj6dmzTD3if9oiDq33tqf7CnAflKK/
  ↳SPgqJyfu6HUf0jLJ0gKIZvkcs2wWY6EjLVdw5qsatsd4FPoCyjvcw==
x-server-name: traffic_ops_golang/
content-length: 68
date: Wed, 05 Dec 2018 01:17:24 GMT

{ "alerts": [
  {
    "text": "cdnfederation was deleted.",
    "level": "success"
  }
]}
```

`cdns/{{name}}/health`

**GET**

Retrieves the health of all *Cache Groups* for a given CDN.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 121: Request Path Parameters

Name	Description
name	The name of the CDN for which health will be reported

## 252: Request Example

```
GET /api/3.0/cdns/CDN-in-a-Box/health HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**cachegroups** An array of objects describing the health of each *Cache Group*

**name** A string that is the *Cache Group's Name*

**offline** The number of OFFLINE *cache servers* in the *Cache Group*

**online** The number of ONLINE *cache servers* in the *Cache Group*

**totalOffline** Total number of OFFLINE *cache servers* across all *Cache Groups* which are assigned to the CDN defined by the name request path parameter

**totalOnline** Total number of ONLINE *cache servers* across all *Cache Groups* which are assigned to the CDN defined by the name request path parameter

## 253: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Encoding: gzip
Content-Length: 108
Content-Type: application/json
Date: Tue, 03 Dec 2019 21:33:59 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; expires=Wed, 04 Dec 2019 01:33:59 GMT;
  ↳path=/; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
  ↳KpXViXeAgch58ueQqdyU8NuINBw1EUedE6Rv2ewcLUajJp6kowdbVynpwW7XiSvAyHdtClIOuT30khIm...
```

(continues on next page)

(continued from previous page)

```
{ "response": {
  "totalOffline": 0,
  "totalOnline": 1,
  "cachegroups": [
    {
      "offline": 0,
      "name": "CDN_in_a_Box_Edge",
      "online": 1
    }
  ]
}}
```

`cdns/name/{name}}`

## DELETE

Allows a user to delete a CDN by name

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 122: Request Path Parameters

Name	Description
name	The name of the CDN to be deleted

## Response Structure

### 254: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: Zy4cJN6BEct4ltFLN4e296mM8XnzOs0EQ3/
→jp4TA3L+g8qtkI0WrL+ThcFq4xbJPU+KHVDSi+b0JBav3xsYPqQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:59:22 GMT
Content-Length: 58

{ "alerts": [
  {
    "text": "cdn was deleted.",
    "level": "success"
  }
]}
```

`cdns/name/{name}}/dnsseckey`

**GET**

Gets a list of DNSSEC keys for CDN and all associated *Delivery Services*. Before returning response to user, this will make sure DNSSEC keys for all *Delivery Services* exist and are not expired. If they don't exist or are expired, they will be (re-)generated.

**Auth. Required** Yes

**Roles Required** "admin"

**Response Type** Object

## Request Structure

Table 123: Request Path Parameters

Name	Description
name	The name of the CDN for which keys will be fetched

## Response Structure

**name** The name of the CDN or *Delivery Service* to which the enclosed keys belong

**zsk** The short-term ZSK

**expirationDate** A Unix epoch timestamp (in seconds) representing the date and time whereupon the key will expire

**inceptionDate** A Unix epoch timestamp (in seconds) representing the date and time when the key was created

**name** The name of the domain for which this key will be used

**private** Encoded private key

**public** Encoded public key

**ttl** The time for which the key should be trusted by the client

**ksk** The long-term KSK

**dsRecord** An optionally present object containing information about the algorithm used to generate the key

**algorithm** The name of the algorithm used to generate the key

**digest** A hash of the DNSKEY record

**digestType** The type of hash algorithm used to create the value of digest

**expirationDate** A Unix epoch timestamp (in seconds) representing the date and time whereupon the key will expire

**inceptionDate** A Unix epoch timestamp (in seconds) representing the date and time when the key was created

**name** The name of the domain for which this key will be used

**private** Encoded private key

**public** Encoded public key

**ttl** The time for which the key should be trusted by the client

255: Response Example

```
{ "response": {  
  "cdn1": {  
    "zsk": {
```

(continues on next page)

(continued from previous page)

```

        "ttl": "60",
        "inceptionDate": "1426196750",
        "private": "zsk private key",
        "public": "zsk public key",
        "expirationDate": "1428788750",
        "name": "foo.kabletown.com."
    },
    "ksk": {
        "name": "foo.kabletown.com.",
        "expirationDate": "1457732750",
        "public": "ksk public key",
        "private": "ksk private key",
        "inceptionDate": "1426196750",
        "ttl": "60",
        "dsRecord": {
            "algorithm": "5",
            "digestType": "2",
            "digest": "abc123def456"
        }
    }
},
"ds-01": {
    "zsk": {
        "ttl": "60",
        "inceptionDate": "1426196750",
        "private": "zsk private key",
        "public": "zsk public key",
        "expirationDate": "1428788750",
        "name": "ds-01.foo.kabletown.com."
    },
    "ksk": {
        "name": "ds-01.foo.kabletown.com.",
        "expirationDate": "1457732750",
        "public": "ksk public key",
        "private": "ksk private key",
        "inceptionDate": "1426196750"
    }
}
}
}

```

### DELETE

Delete DNSSEC keys for a CDN and all associated *Delivery Services*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object (string)

### Request Structure

Table 124: Request Path Parameters

Name	Description
name	The name of the CDN for which DNSSEC keys will be deleted

### Response Structure

256: Response Example

```
{
  "response": "Successfully deleted dnssec keys for test"
}
```

`cdns/name/{{name}}/sslkeys`

### GET

Returns SSL certificates for all *Delivery Services* that are a part of the CDN.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Array

### Request Structure

Table 125: Request Path Parameters

Name	Description
name	The name of the CDN for which keys will be fetched



## Response Structure

**certificate** An object representing The SSL keys used for the *Delivery Service* identified by `deliveryservice`

**key** Base 64-encoded private key for SSL certificate

**crt** Base 64-encoded SSL certificate

**deliveryservice** A string that is the *xml\_id* of the *Delivery Service* using the SSL key within `certificate`

### 257: Response Example

```
{ "response": [
  {
    "deliveryservice": "ds1",
    "certificate": {
      "crt": "base64encodedcrt1",
      "key": "base64encodedkey1"
    }
  },
  {
    "deliveryservice": "ds2",
    "certificate": {
      "crt": "base64encodedcrt2",
      "key": "base64encodedkey2"
    }
  }
]}
```

`cdns/{{name}}/snapshot`

**Caution:** This page is a stub! Much of it may be missing or just downright wrong - it needs a lot of love from people with the domain knowledge required to update it.

## GET

Retrieves the *current Snapshot* for a CDN, which represents the current *operating state* of the CDN, **not** the current *configuration* of the CDN. The contents of this *Snapshot* are currently used by Traffic Monitor and Traffic Router.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 126: Request Path Parameters

Name	Description
name	The name of the CDN for which a <i>Snapshot</i> shall be returned

### 258: Request Example

```
GET /api/3.0/cdns/CDN-in-a-Box/snapshot HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**config** An object containing basic configurations on the actual CDN object

**api.cache-control.max-age** A string containing an integer which specifies the value of `max-age` in the `Cache-Control` header of some HTTP responses, likely the *Traffic Router API* responses

**certificates.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for updated SSL certificates

**consistent.dns.routing** A string containing a boolean which indicates whether DNS routing will use a consistent hashing method or “round-robin”

“false” The “round-robin” method will be used to define DNS routing

“true” A consistent hashing method will be used to define DNS routing

**coveragezone.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should check for a new Coverage Zone file

**coveragezone.polling.url** The URL where a *Coverage Zone File* may be requested by Traffic Routers

**dnssec.dynamic.response.expiration** A string containing a number and unit suffix that specifies the length of time

for which dynamic responses to DNSSEC lookup queries should remain valid

**dnssec.dynamic.concurrencylevel** An integer that defines the size of the concurrency level (threads) of the Guava cache used by ZoneManager to store zone material

**dnssec.dynamic.initialcapacity** An integer that defines the initial size of the Guava cache, default is 10000. Too low of a value can lead to expensive resizing

**dnssec.init.timeout** An integer that defines the number of minutes to allow for zone generation, this bounds the zone priming activity

**dnssec.enabled** A string that tells whether or not the CDN uses DNSSEC; one of:

“false” DNSSEC is not used within this CDN

“true” DNSSEC is used within this CDN

**domain\_name** A string that is the TLD served by the CDN

**federationmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new federation mappings

**federationmapping.polling.url** The URL where Traffic Control components can request federation mappings

**geolocation.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new IP-to-geographic-location mapping databases

**geolocation.polling.url** The URL where Traffic Control components can request IP-to-geographic-location mapping database files

**keystore.maintenance.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should refresh their zone caches

**neustar.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new “Neustar” databases

**neustar.polling.url** The URL where Traffic Control components can request “Neustar” databases

**soa** An object defining the SOA for the CDN’s TLD (defined in domain\_name)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

**See also:**

[The Wikipedia page on Start of Authority records.](#)

**steeringmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Control components should check for new steering mappings

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

**zonemanager.cache.maintenance.interval** A configuration option for the ZoneManager Java class of Traffic Router

**zonemanager.threadpool.scale** A configuration option for the ZoneManager Java class of Traffic Router

**contentRouters** An object containing keys which are the (short) hostnames of the Traffic Routers that serve requests for *Delivery Services* in this CDN

**api.port** A string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTP

**secure.api.port** An optionally present string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTPS

**fqdn** This Traffic Router's FQDN

**httpsPort** The port number on which this Traffic Router listens for incoming HTTPS requests

**ip** This Traffic Router's IPv4 address

**ip6** This Traffic Router's IPv6 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Router belongs

**port** The port number on which this Traffic Router listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by this Traffic Router

**status** The health status of this Traffic Router

**See also:**

*Health Protocol*

**contentServers** An object containing keys which are the (short) hostnames of the *Edge-tier cache servers* in the CDN; the values corresponding to those keys are routing information for said servers

**cacheGroup** A string that is the *Name* of the *Cache Group* to which the server belongs

**capabilities** An array of this *Cache Server's Server Capabilities*. If the Cache Server has no Server Capabilities, this field is omitted.

**deliveryServices** An object containing keys which are the names of *Delivery Services* to which this *cache server* is assigned; the values corresponding to those keys are arrays of FQDNs (Fully Qualified Domain Names) that resolve to this *cache server*

---

**Note:** Only *Edge-tier cache servers* can be assigned to a *Delivery Service*, and therefore this field will only be present when type is "EDGE".

---

**fqdn** The server's FQDN

**hashCount** The number of servers to be placed into a single “hash ring” in Traffic Router

**hashId** A unique string to be used as the key for hashing servers  
- as of version 3.0.0 of Traffic Control, this is always the same as the server’s (short) hostname and only still exists for legacy compatibility reasons

**httpsPort** The port on which the *cache server* listens for incoming HTTPS requests

**interfaceName** The name of the main network interface device used by this *cache server*

**ip6** The server’s IPv6 address

**ip** The server’s IPv4 address

**locationId** This field is exactly the same as *cacheGroup* and only exists for legacy compatibility reasons

**port** The port on which this *cache server* listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by the *cache server*

**routingDisabled** An integer representing the boolean concept of whether or not Traffic Routers should route client traffic to this *cache server*; one of:

**0** Do not route traffic to this server

**1** Route traffic to this server normally

**status** This *cache server*’s status

**See also:**

*Health Protocol*

**type** The *Type* of this *cache server*; which ought to be one of (but in practice need not be in certain special circumstances):

**EDGE** This is an *Edge-tier cache server*

**MID** This is a *Mid-tier cache server*

**deliveryServices** An object containing keys which are the *xml\_ids* of all of the *Delivery Services* within the CDN

**anonymousBlockingEnabled** A string containing a boolean that tells whether or not *Anonymous Blocking* is set on this *Delivery Service*; one of:

**“true”** Anonymized IP addresses are blocked by this *Delivery Service*

“false” Anonymized IP addresses are not blocked by this *Delivery Service*

See also:

*Configure Anonymous Blocking*

**consistentHashQueryParameters** A set of query parameters that Traffic Router should consider when determining a consistent hash for a given client request.

**consistentHashRegex** An optional regular expression that will ensure clients are consistently routed to a *cache server* based on matches to it.

**coverageZoneOnly** A string containing a boolean that tells whether or not this *Delivery Service* routes traffic based only on its *Coverage Zone File*

See also:

*Geo Limit*

**deepCachingType** A string that defines the *Deep Caching* setting of this *Delivery Service*

**dispersion** An object describing the “dispersion” - or number of *cache servers* within a single *Cache Group* across which the same content is spread - within the *Delivery Service*

**limit** The maximum number of *cache servers* in which the response to a single request URL will be stored

---

**Note:** If this is greater than the number of *cache servers* in the *Cache Group* chosen to service the request, then content will be spread across all of them. That is, it causes no problems.

---

**shuffled** A string containing a boolean that tells whether the *cache servers* chosen for content dispersion are chosen randomly or based on a consistent hash of the request URL; one of:

“false” *cache servers* will be chosen consistently

“true” *cache servers* will be chosen at random

**domains** An array of domains served by this *Delivery Service*

**ecsEnabled** A string containing a boolean from *EDNS0 Client Subnet Enabled* that tells whether EDNS0 client subnet is enabled on this *Delivery Service*; one of:

“false” EDNS0 client subnet is not enabled on this *Delivery Service*

“true” EDNS0 client subnet is enabled on this *Delivery Service*

**geolocationProvider** The name of a provider for IP-to-geographic-location mapping services - currently the only valid value is "maxmindGeolocationService"

**ip6RoutingEnabled** A string containing a boolean that defines the *IPv6 Routing Enabled* setting for this *Delivery Service*; one of:

“false” IPv6 traffic will not be routed by this *Delivery Service*

“true” IPv6 traffic will be routed by this *Delivery Service*

**matchList** An array of methods used by Traffic Router to determine whether or not a request can be serviced by this *Delivery Service*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integral, unique identifier for the set of types to which the **type** field belongs

**type** The name of the *Type* of match performed using **pattern** to determine whether or not to use this *Delivery Service*

**HOST\_REGEX** Use the *Delivery Service* if **pattern** matches the *Host* HTTP header of an HTTP request, or the name requested for resolution in a DNS request

**HEADER\_REGEX** Use the *Delivery Service* if **pattern** matches an HTTP header (both the name and value) in an HTTP request<sup>1</sup>

**PATH\_REGEX** Use the *Delivery Service* if **pattern** matches the

---

<sup>1</sup> These only apply to HTTP-routed *Delivery Services*



request path of this *Delivery Service*'s URL Page 455, 1

**STEERING\_REGEX** Use the *Delivery Service* if `pattern` matches the *xml\_id* of one of this *Delivery Service*'s “Steering” target *Delivery Services*

**missLocation** An object representing the default geographic coordinates to use for a client when lookup of their IP has failed in both the *Coverage Zone File* (and/or possibly the *Deep Coverage Zone File*) and the IP-to-geographic-location database

**lat** Geographic latitude as a floating point number

**long** Geographic longitude as a floating point number

**protocol** An object that describes how the *Delivery Service* ought to handle HTTP requests both with and without TLS encryption

**acceptHttps** A string containing a boolean that tells whether HTTPS requests should be normally serviced by this *Delivery Service*; one of:

“false” Refuse to service HTTPS requests

“true” Service HTTPS requests normally

**redirectToHttps** A string containing a boolean that tells whether HTTP requests ought to be re-directed to use HTTPS; one of:

“false” Do not redirect unencrypted traffic; service it normally

“true” Respond to HTTP requests with instructions to use HTTPS instead

See also:

*Protocol*

**regionalGeoBlocking** A string containing a boolean that defines the *Regional Geoblocking* setting of this *Delivery Service*; one of:

“false” Regional Geographic Blocking is not used by this *Delivery Service*

“true” Regional Geographic Blocking is used by this *Delivery Service*

See also:

*Configure Regional Geo-blocking (RGB)*

**requiredCapabilities** An array of this Delivery Service’s *required capabilities*. If there are no required capabilities, this field is omitted.

**routingName** A string that is this *Delivery Service’s Routing Name*

**soa** An object defining the SOA record for the *Delivery Service’s* TLDs (Top-Level Domains) (defined in *domains*)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

See also:

The Wikipedia page on Start of Authority records.

**sslEnabled** A string containing a boolean that tells whether this *Delivery Service* uses SSL; one of:

“false” SSL is not used by this *Delivery Service*

“true” SSL is used by this *Delivery Service*

See also:

*Protocol*

**topology** The name of the *Topology* that this *Delivery Service* is assigned to. If the Delivery Service is not assigned to a topology, this field is omitted.

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

---

**Note:** This overrides `config.ttls`.

---

**edgeLocations** An object containing keys which are the names of Edge-Tier *Cache Groups* within the CDN

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines the *Fallback to Closest* behavior of this *Cache Group*; one of:

“false” Do not fall back on the closest available *Cache Group*

“true” Fall back on the closest available *Cache Group*

**list** If this *Cache Group* has any *Fallbacks*, this key will appear and will be an array of those *Cache Groups' Names*

**latitude** A floating point number that defines this *Cache Group's Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

**monitors** An object containing keys which are the (short) hostnames of Traffic Monitors within this CDN

**fqdn** The FQDN of this Traffic Monitor

**httpsPort** The port number on which this Traffic Monitor listens for incoming HTTPS requests

**ip6** This Traffic Monitor's IPv6 address

**ip** This Traffic Monitor's IPv4 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Monitor belongs

**port** The port number on which this Traffic Monitor listens for incoming HTTP requests

**profile** A string which is the *Name* of the *Profile* used by this Traffic Monitor

---

**Note:** For legacy reasons, this must always start with "RASCAL-".

---

**status** The health status of this Traffic Monitor

**See also:**

*Health Protocol*

**stats** An object containing metadata information regarding the CDN

**CDN\_name** The name of this CDN

**date** The UNIX epoch timestamp date in the Traffic Ops server's own timezone

**tm\_host** The FQDN of the Traffic Ops server

Deprecated since version ATCv6: This information should never be used; instead all tools and (especially) components **must** use the documented API. This field was removed in APIv4

**tm\_path** A path relative to the root of the Traffic Ops server where a request may be replaced to have this *Snapshot* overwritten by the current *configured state* of the CDN

**tm\_user** The username of the currently logged-in user

**tm\_version** The full version number of the Traffic Ops server, including release number, git commit hash, and supported Enterprise Linux version

**topologies** An array of *Topologies* where each key is the name of that Topology.

**nodes** An array of the names of the *Edge-Tier Cache Groups* in this *Topology*. *Mid-Tier* Cache Groups in the topology are not included.

**trafficRouterLocations** An object containing keys which are the *names of Cache Groups* within the CDN which contain Traffic Routers

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines this *Cache Group's Fallback to Closest* setting; one of:

**“false”** Do not fall back on the closest available *Cache Group*

**“true”** Fall back on the closest available *Cache Group*

**latitude** A floating point number that defines this *Cache Group's Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

## 259: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 27 May 2020 18:33:17
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
    ↳B5qdN9URIfu1lgQxPZ8YaaMvy2HMrzsnrpt6vF037yv60QiKCRyrUMX6wYs7QW4YVaeUrvmS2ya5l2YC0l
X-Server-Name: traffic_ops_golang/
Date: Wed, 27 May 2020 17:33:17 GMT
Content-Length: 1360

{
  "response": {
```

(continues on next page)

(continued from previous page)

```

"config": {
  "api.cache-control.max-age": "10",
  "certificates.polling.interval": "300000",
  "consistent.dns.routing": "true",
  "coveragezone.polling.interval": "3600000",
  "coveragezone.polling.url": "https://trafficops.infra.ciab.
→test:443/coverage-zone.json",
  "dnssec.dynamic.response.expiration": "300s",
  "dnssec.enabled": "false",
  "domain_name": "mycdn.ciab.test",
  "federationmapping.polling.interval": "60000",
  "federationmapping.polling.url": "https://${toHostname}/
→api/3.0/federations/all",
  "geolocation.polling.interval": "86400000",
  "geolocation.polling.url": "https://trafficops.infra.ciab.
→test:443/GeoLite2-City.mmdb.gz",
  "keystore.maintenance.interval": "300",
  "neustar.polling.interval": "86400000",
  "neustar.polling.url": "https://trafficops.infra.ciab.
→test:443/neustar.tar.gz",
  "soa": {
    "admin": "twelve_monkeys",
    "expire": "604800",
    "minimum": "30",
    "refresh": "28800",
    "retry": "7200"
  },
  "steeringmapping.polling.interval": "60000",
  "ttls": {
    "A": "3600",
    "AAAA": "3600",
    "DNSKEY": "30",
    "DS": "30",
    "NS": "3600",
    "SOA": "86400"
  },
  "zonemanager.cache.maintenance.interval": "300",
  "zonemanager.threadpool.scale": "0.50"
},
"contentRouters": {
  "trafficrouter": {
    "api.port": "3333",
    "fqdn": "trafficrouter.infra.ciab.test",
    "httpsPort": 443,
    "ip": "172.26.0.15",
    "ip6": "",

```

(continues on next page)

(continued from previous page)

```

        "location": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "CCR_CIAB",
        "secure.api.port": "3443",
        "status": "ONLINE"
    },
},
"contentServers": {
    "edge": {
        "cacheGroup": "CDN_in_a_Box_Edge",
        "capabilities": [
            "RAM_DISK_STORAGE"
        ],
        "fqdn": "edge.infra.ciab.test",
        "hashCount": 999,
        "hashId": "edge",
        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.3",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "ATS_EDGE_TIER_CACHE",
        "routingDisabled": 0,
        "status": "REPORTED",
        "type": "EDGE"
    },
    "mid": {
        "cacheGroup": "CDN_in_a_Box_Mid",
        "capabilities": [
            "RAM_DISK_STORAGE"
        ],
        "fqdn": "mid.infra.ciab.test",
        "hashCount": 999,
        "hashId": "mid",
        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.4",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Mid",
        "port": 80,
        "profile": "ATS_MID_TIER_CACHE",
        "routingDisabled": 0,
        "status": "REPORTED",
        "type": "MID"
    }
}

```

(continues on next page)

(continued from previous page)

```
},
"deliveryServices": {
  "demo1": {
    "anonymousBlockingEnabled": "false",
    "consistentHashQueryParams": [
      "abc",
      "pdq",
      "xxx",
      "zyx"
    ],
    "coverageZoneOnly": "false",
    "deepCachingType": "NEVER",
    "dispersion": {
      "limit": 1,
      "shuffled": "true"
    },
    "domains": [
      "demo1.mycdn.ciab.test"
    ],
    "ecsEnabled": "false",
    "geolocationProvider": "maxmindGeolocationService",
    "ip6RoutingEnabled": "true",
    "matchsets": [
      {
        "matchlist": [
          {
            "match-type": "HOST",
            "regex": ".*\\.demo1\\.\\.\\.*"
          }
        ],
        "protocol": "HTTP"
      }
    ],
    "missLocation": {
      "lat": 42,
      "long": -88
    },
    "protocol": {
      "acceptHttps": "true",
      "redirectToHttps": "false"
    },
    "regionalGeoBlocking": "false",
    "requiredCapabilities": [
      "RAM_DISK_STORAGE"
    ],
    "routingName": "video",
```

(continues on next page)



(continued from previous page)

```

    "soa": {
      "admin": "traffic_ops",
      "expire": "604800",
      "minimum": "30",
      "refresh": "28800",
      "retry": "7200"
    },
    "sslEnabled": "true",
    "topology": "my-topology",
    "ttls": {
      "A": "",
      "AAAA": "",
      "NS": "3600",
      "SOA": "86400"
    }
  },
  "edgeLocations": {
    "CDN_in_a_Box_Edge": {
      "backupLocations": {
        "fallbackToClosest": "true"
      },
      "latitude": 38.897663,
      "localizationMethods": [
        "GEO",
        "CZ",
        "DEEP_CZ"
      ],
      "longitude": -77.036574
    }
  },
  "monitors": {
    "trafficmonitor": {
      "fqdn": "trafficmonitor.infra.ciab.test",
      "httpsPort": 443,
      "ip": "172.26.0.14",
      "ip6": "",
      "location": "CDN_in_a_Box_Edge",
      "port": 80,
      "profile": "RASCAL-Traffic_Monitor",
      "status": "ONLINE"
    }
  },
  "stats": {
    "CDN_name": "CDN-in-a-Box",
    "date": 1590600715,

```

(continues on next page)

(continued from previous page)

```

        "tm_host": "trafficops.infra.ciab.test:443",
        "tm_path": "/api/3.0/snapshot",
        "tm_user": "admin",
        "tm_version": "development"
    },
    "topologies": {
        "my-topology": {
            "nodes": [
                "CDN_in_a_Box_Edge"
            ]
        }
    },
    "trafficRouterLocations": {
        "CDN_in_a_Box_Edge": {
            "backupLocations": {
                "fallbackToClosest": "false"
            },
            "latitude": 38.897663,
            "localizationMethods": [
                "GEO",
                "CZ",
                "DEEP_CZ"
            ],
            "longitude": -77.036574
        }
    }
}

```

`cdns/{name}/snapshot/new`

**GET**

Retrieves the *pending Snapshot* for a CDN, which represents the current *configuration* of the CDN, **not** the current *operating state* of the CDN. The contents of this *Snapshot* are currently used by Traffic Monitor and Traffic Router.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 127: Request Path Parameters

Name	Description
name	The name of the CDN for which a <i>Snapshot</i> shall be returned

260: Request Example

```
GET /api/3.0/cdns/CDN-in-a-Box/snapshot/new HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**config** An object containing basic configurations on the actual CDN object

**api.cache-control.max-age** A string containing an integer which specifies the value of `max-age` in the `Cache-Control` header of some HTTP responses, likely the *Traffic Router API* responses

**certificates.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for updated SSL certificates

**consistent.dns.routing** A string containing a boolean which indicates whether DNS routing will use a consistent hashing method or “round-robin”

“false” The “round-robin” method will be used to define DNS routing

“true” A consistent hashing method will be used to define DNS routing

**coveragezone.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should check for a new Coverage Zone file

**coveragezone.polling.url** The URL where a *Coverage Zone File* may be requested by Traffic Routers

**dnssec.dynamic.response.expiration** A string containing a number and unit suffix that specifies the length of time

for which dynamic responses to DNSSEC lookup queries should remain valid

**dnssec.dynamic.concurrencylevel** An integer that defines the size of the concurrency level (threads) of the Guava cache used by ZoneManager to store zone material

**dnssec.dynamic.initialcapacity** An integer that defines the initial size of the Guava cache, default is 10000. Too low of a value can lead to expensive resizing

**dnssec.init.timeout** An integer that defines the number of minutes to allow for zone generation, this bounds the zone priming activity

**dnssec.enabled** A string that tells whether or not the CDN uses DNSSEC; one of:

“false” DNSSEC is not used within this CDN

“true” DNSSEC is used within this CDN

**domain\_name** A string that is the TLD served by the CDN

**federationmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new federation mappings

**federationmapping.polling.url** The URL where Traffic Control components can request federation mappings

**geolocation.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new IP-to-geographic-location mapping databases

**geolocation.polling.url** The URL where Traffic Control components can request IP-to-geographic-location mapping database files

**keystore.maintenance.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should refresh their zone caches

**neustar.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new “Neustar” databases

**neustar.polling.url** The URL where Traffic Control components can request “Neustar” databases

**soa** An object defining the SOA for the CDN’s TLD (defined in domain\_name)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

**See also:**

[The Wikipedia page on Start of Authority records.](#)

**steeringmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Control components should check for new steering mappings

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

**zonemanager.cache.maintenance.interval** A configuration option for the ZoneManager Java class of Traffic Router

**zonemanager.threadpool.scale** A configuration option for the ZoneManager Java class of Traffic Router

**contentRouters** An object containing keys which are the (short) hostnames of the Traffic Routers that serve requests for *Delivery Services* in this CDN

**api.port** A string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTP

**secure.api.port** An optionally present string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTPS

**fqdn** This Traffic Router's FQDN

**httpsPort** The port number on which this Traffic Router listens for incoming HTTPS requests

**ip** This Traffic Router's IPv4 address

**ip6** This Traffic Router's IPv6 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Router belongs

**port** The port number on which this Traffic Router listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by this Traffic Router

**status** The health status of this Traffic Router

**See also:**

*Health Protocol*

**contentServers** An object containing keys which are the (short) hostnames of the *Edge-tier cache servers* in the CDN; the values corresponding to those keys are routing information for said servers

**cacheGroup** A string that is the *Name* of the *Cache Group* to which the server belongs

**capabilities** An array of this *Cache Server's Server Capabilities*. If the Cache Server has no Server Capabilities, this field is omitted.

**deliveryServices** An object containing keys which are the names of *Delivery Services* to which this *cache server* is assigned; the values corresponding to those keys are arrays of FQDNs that resolve to this *cache server*

---

**Note:** Only *Edge-tier cache servers* can be assigned to a *Delivery Service*, and therefore this field will only be present when *type* is "EDGE".

---

**fqdn** The server's FQDN

**hashCount** The number of servers to be placed into a single “hash ring” in Traffic Router

**hashId** A unique string to be used as the key for hashing servers  
- as of version 3.0.0 of Traffic Control, this is always the same as the server’s (short) hostname and only still exists for legacy compatibility reasons

**httpsPort** The port on which the *cache server* listens for incoming HTTPS requests

**interfaceName** The name of the main network interface device used by this *cache server*

**ip6** The server’s IPv6 address

**ip** The server’s IPv4 address

**locationId** This field is exactly the same as *cacheGroup* and only exists for legacy compatibility reasons

**port** The port on which this *cache server* listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by the *cache server*

**routingDisabled** An integer representing the boolean concept of whether or not Traffic Routers should route client traffic to this *cache server*; one of:

**0** Do not route traffic to this server

**1** Route traffic to this server normally

**status** This *cache server*’s status

**See also:**

*Health Protocol*

**type** The *Type* of this *cache server*; which ought to be one of (but in practice need not be in certain special circumstances):

**EDGE** This is an *Edge-tier cache server*

**MID** This is a *Mid-tier cache server*

**deliveryServices** An object containing keys which are the *xml\_ids* of all of the *Delivery Services* within the CDN

**anonymousBlockingEnabled** A string containing a boolean that tells whether or not *Anonymous Blocking* is set on this *Delivery Service*; one of:

**“true”** Anonymized IP addresses are blocked by this *Delivery Service*

“false” Anonymized IP addresses are not blocked by this *Delivery Service*

See also:

*Configure Anonymous Blocking*

**consistentHashQueryParameters** A set of query parameters that Traffic Router should consider when determining a consistent hash for a given client request.

**consistentHashRegex** An optional regular expression that will ensure clients are consistently routed to a *cache server* based on matches to it.

**coverageZoneOnly** A string containing a boolean that tells whether or not this *Delivery Service* routes traffic based only on its *Coverage Zone File*

See also:

*Geo Limit*

**deepCachingType** A string that defines the *Deep Caching* setting of this *Delivery Service*

**dispersion** An object describing the “dispersion” - or number of *cache servers* within a single *Cache Group* across which the same content is spread - within the *Delivery Service*

**limit** The maximum number of *cache servers* in which the response to a single request URL will be stored

---

**Note:** If this is greater than the number of *cache servers* in the *Cache Group* chosen to service the request, then content will be spread across all of them. That is, it causes no problems.

---

**shuffled** A string containing a boolean that tells whether the *cache servers* chosen for content dispersion are chosen randomly or based on a consistent hash of the request URL; one of:

“false” *cache servers* will be chosen consistently

“true” *cache servers* will be chosen at random

**domains** An array of domains served by this *Delivery Service*



**ecsEnabled** A string containing a boolean from *EDNS0 Client Subnet Enabled* that tells whether EDNS0 client subnet is enabled on this *Delivery Service*; one of:

“false” EDNS0 client subnet is not enabled on this *Delivery Service*

“true” EDNS0 client subnet is enabled on this *Delivery Service*

**geolocationProvider** The name of a provider for IP-to-geographic-location mapping services - currently the only valid value is "maxmindGeolocationService"

**ip6RoutingEnabled** A string containing a boolean that defines the *IPv6 Routing Enabled* setting for this *Delivery Service*; one of:

“false” IPv6 traffic will not be routed by this *Delivery Service*

“true” IPv6 traffic will be routed by this *Delivery Service*

**matchList** An array of methods used by Traffic Router to determine whether or not a request can be serviced by this *Delivery Service*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integral, unique identifier for the set of types to which the **type** field belongs

**type** The name of the *Type* of match performed using **pattern** to determine whether or not to use this *Delivery Service*

**HOST\_REGEX** Use the *Delivery Service* if **pattern** matches the *Host* HTTP header of an HTTP request, or the name requested for resolution in a DNS request

**HEADER\_REGEX** Use the *Delivery Service* if **pattern** matches an HTTP header (both the name and value) in an HTTP request<sup>1</sup>

**PATH\_REGEX** Use the *Delivery Service* if **pattern** matches the

---

<sup>1</sup> These only apply to HTTP-routed *Delivery Services*

request path of this *Delivery Service*'s URL Page 455, 1

**STEERING\_REGEX** Use the *Delivery Service* if `pattern` matches the *xml\_id* of one of this *Delivery Service*'s “Steering” target *Delivery Services*

**missLocation** An object representing the default geographic coordinates to use for a client when lookup of their IP has failed in both the *Coverage Zone File* (and/or possibly the *Deep Coverage Zone File*) and the IP-to-geographic-location database

**lat** Geographic latitude as a floating point number

**long** Geographic longitude as a floating point number

**protocol** An object that describes how the *Delivery Service* ought to handle HTTP requests both with and without TLS encryption

**acceptHttps** A string containing a boolean that tells whether HTTPS requests should be normally serviced by this *Delivery Service*; one of:

“false” Refuse to service HTTPS requests

“true” Service HTTPS requests normally

**redirectToHttps** A string containing a boolean that tells whether HTTP requests ought to be re-directed to use HTTPS; one of:

“false” Do not redirect unencrypted traffic; service it normally

“true” Respond to HTTP requests with instructions to use HTTPS instead

See also:

*Protocol*

**regionalGeoBlocking** A string containing a boolean that defines the *Regional Geoblocking* setting of this *Delivery Service*; one of:

“false” Regional Geographic Blocking is not used by this *Delivery Service*

“true” Regional Geographic Blocking is used by this *Delivery Service*

See also:

*Configure Regional Geo-blocking (RGB)*

**requiredCapabilities** An array of this Delivery Service’s *required capabilities*. If there are no required capabilities, this field is omitted.

**routingName** A string that is this *Delivery Service’s Routing Name*

**soa** An object defining the SOA record for the *Delivery Service’s* TLDs (defined in domains)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

See also:

The Wikipedia page on Start of Authority records.

**sslEnabled** A string containing a boolean that tells whether this *Delivery Service* uses SSL; one of:

“false” SSL is not used by this *Delivery Service*

“true” SSL is used by this *Delivery Service*

See also:

*Protocol*

**topology** The name of the *Topology* that this *Delivery Service* is assigned to. If the Delivery Service is not assigned to a topology, this field is omitted.

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

---

**Note:** This overrides `config.ttls`.

---

**edgeLocations** An object containing keys which are the names of Edge-Tier *Cache Groups* within the CDN

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines the *Fallback to Closest* behavior of this *Cache Group*; one of:

“false” Do not fall back on the closest available *Cache Group*

“true” Fall back on the closest available *Cache Group*

**list** If this *Cache Group* has any *Fallbacks*, this key will appear and will be an array of those *Cache Groups' Names*

**latitude** A floating point number that defines this *Cache Group's Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

**monitors** An object containing keys which are the (short) hostnames of Traffic Monitors within this CDN

**fqdn** The FQDN of this Traffic Monitor

**httpsPort** The port number on which this Traffic Monitor listens for incoming HTTPS requests

**ip6** This Traffic Monitor's IPv6 address

**ip** This Traffic Monitor's IPv4 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Monitor belongs

**port** The port number on which this Traffic Monitor listens for incoming HTTP requests

**profile** A string which is the *Name* of the *Profile* used by this Traffic Monitor

---

**Note:** For legacy reasons, this must always start with "RASCAL-".

---

**status** The health status of this Traffic Monitor

**See also:**

*Health Protocol*

**stats** An object containing metadata information regarding the CDN

**CDN\_name** The name of this CDN

**date** The UNIX epoch timestamp date in the Traffic Ops server's own timezone

**tm\_host** The FQDN of the Traffic Ops server

**tm\_path** A path relative to the root of the Traffic Ops server where a request may be replaced to have this *Snapshot* overwritten by the current *configured state* of the CDN

Deprecated since version ATCv6: This information should never be used; instead all tools and (especially) components **must** use the documented API. This field was removed in APIv4

**tm\_user** The username of the currently logged-in user

**tm\_version** The full version number of the Traffic Ops server, including release number, git commit hash, and supported Enterprise Linux version

**topologies** An array of *Topologies* where each key is the name of that Topology.

**nodes** An array of the names of the *Edge-Tier Cache Groups* in this *Topology*. *Mid-Tier* Cache Groups in the topology are not included.

**trafficRouterLocations** An object containing keys which are the *names of Cache Groups* within the CDN which contain Traffic Routers

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines this *Cache Group's Fallback to Closest* setting; one of:

**“false”** Do not fall back on the closest available *Cache Group*

**“true”** Fall back on the closest available *Cache Group*

**latitude** A floating point number that defines this *Cache Group's Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

#### 261: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 27 May 2020 20:31:13.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→M6uhE2oPpjpTUR7gALsP0nM2CepD+VCAj4dj5Xnppo0G5zL31PQgiteD23q67r7/bq/
→JJpMvIvdaENVYFtrqQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 27 May 2020 19:31:13 GMT
Content-Length: 1374

{
  "response": {
```

(continues on next page)

(continued from previous page)

```

"config": {
  "api.cache-control.max-age": "10",
  "certificates.polling.interval": "300000",
  "consistent.dns.routing": "true",
  "coveragezone.polling.interval": "3600000",
  "coveragezone.polling.url": "https://trafficops.infra.ciab.
→test:443/coverage-zone.json",
  "dnssec.dynamic.response.expiration": "300s",
  "dnssec.enabled": "false",
  "domain_name": "mycdn.ciab.test",
  "federationmapping.polling.interval": "60000",
  "federationmapping.polling.url": "https://${toHostname}/
→api/3.0/federations/all",
  "geolocation.polling.interval": "86400000",
  "geolocation.polling.url": "https://trafficops.infra.ciab.
→test:443/GeoLite2-City.mmdb.gz",
  "keystore.maintenance.interval": "300",
  "neustar.polling.interval": "86400000",
  "neustar.polling.url": "https://trafficops.infra.ciab.
→test:443/neustar.tar.gz",
  "soa": {
    "admin": "twelve_monkeys",
    "expire": "604800",
    "minimum": "30",
    "refresh": "28800",
    "retry": "7200"
  },
  "steeringmapping.polling.interval": "60000",
  "ttls": {
    "A": "3600",
    "AAAA": "3600",
    "DNSKEY": "30",
    "DS": "30",
    "NS": "3600",
    "SOA": "86400"
  },
  "zonemanager.cache.maintenance.interval": "300",
  "zonemanager.threadpool.scale": "0.50"
},
"contentServers": {
  "edge": {
    "cacheGroup": "CDN_in_a_Box_Edge",
    "capabilities": [
      "RAM_DISK_STORAGE"
    ],
    "fqdn": "edge.infra.ciab.test",

```

(continues on next page)

(continued from previous page)

```

        "hashCount": 999,
        "hashId": "edge",
        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.3",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "ATS_EDGE_TIER_CACHE",
        "status": "REPORTED",
        "type": "EDGE",
        "routingDisabled": 0
    },
    "mid": {
        "cacheGroup": "CDN_in_a_Box_Mid",
        "capabilities": [
            "RAM_DISK_STORAGE"
        ],
        "fqdn": "mid.infra.ciab.test",
        "hashCount": 999,
        "hashId": "mid",
        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.4",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Mid",
        "port": 80,
        "profile": "ATS_MID_TIER_CACHE",
        "status": "REPORTED",
        "type": "MID",
        "routingDisabled": 0
    }
},
"contentRouters": {
    "trafficrouter": {
        "api.port": "3333",
        "fqdn": "trafficrouter.infra.ciab.test",
        "httpsPort": 443,
        "ip": "172.26.0.15",
        "ip6": "",
        "location": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "CCR_CIAB",
        "secure.api.port": "3443",
        "status": "ONLINE"
    }
}

```

(continues on next page)



(continued from previous page)

```

},
"deliveryServices": {
  "demo1": {
    "anonymousBlockingEnabled": "false",
    "consistentHashQueryParams": [
      "abc",
      "pdq",
      "xxx",
      "zyx"
    ],
    "coverageZoneOnly": "false",
    "deepCachingType": "NEVER",
    "dispersion": {
      "limit": 1,
      "shuffled": "true"
    },
    "domains": [
      "demo1.mycdn.ciab.test"
    ],
    "ecsEnabled": "false",
    "geolocationProvider": "maxmindGeolocationService",
    "ip6RoutingEnabled": "true",
    "matchsets": [
      {
        "protocol": "HTTP",
        "matchlist": [
          {
            "regex": ".*\\.demo1\\.\\.\\.\"",
            "match-type": "HOST"
          }
        ]
      }
    ],
    "missLocation": {
      "lat": 42,
      "long": -88
    },
    "protocol": {
      "acceptHttps": "true",
      "redirectToHttps": "false"
    },
    "regionalGeoBlocking": "false",
    "requiredCapabilities": [
      "RAM_DISK_STORAGE"
    ],
    "routingName": "video",

```

(continues on next page)

(continued from previous page)

```

    "soa": {
        "admin": "traffic_ops",
        "expire": "604800",
        "minimum": "30",
        "refresh": "28800",
        "retry": "7200"
    },
    "sslEnabled": "true",
    "topology": "my-topology",
    "ttls": {
        "A": "",
        "AAAA": "",
        "NS": "3600",
        "SOA": "86400"
    }
},
"edgeLocations": {
    "CDN_in_a_Box_Edge": {
        "latitude": 38.897663,
        "longitude": -77.036574,
        "backupLocations": {
            "fallbackToClosest": "true"
        },
        "localizationMethods": [
            "GEO",
            "CZ",
            "DEEP_CZ"
        ]
    }
},
"trafficRouterLocations": {
    "CDN_in_a_Box_Edge": {
        "latitude": 38.897663,
        "longitude": -77.036574,
        "backupLocations": {
            "fallbackToClosest": "false"
        },
        "localizationMethods": [
            "GEO",
            "CZ",
            "DEEP_CZ"
        ]
    }
},
"monitors": {

```

(continues on next page)

(continued from previous page)

```

    "trafficmonitor": {
      "fqdn": "trafficmonitor.infra.ciab.test",
      "httpsPort": 443,
      "ip": "172.26.0.14",
      "ip6": "",
      "location": "CDN_in_a_Box_Edge",
      "port": 80,
      "profile": "RASCAL-Traffic_Monitor",
      "status": "ONLINE"
    },
    "stats": {
      "CDN_name": "CDN-in-a-Box",
      "date": 1590607873,
      "tm_host": "trafficops.infra.ciab.test:443",
      "tm_path": "/api/3.0/cdns/CDN-in-a-Box/snapshot/new",
      "tm_user": "admin",
      "tm_version": "development"
    },
    "topologies": {
      "my-topology": {
        "nodes": [
          "CDN_in_a_Box_Edge"
        ]
      }
    }
  }
}

```

## cdns/routing

### GET

Retrieves the aggregated routing percentages across all CDNs. This is accomplished by polling stats from all online Traffic Routers via the `/crs/stats` route.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

No parameters available

## Response Structure

- cz** The percent of requests to online Traffic Routers that were satisfied by a *Coverage Zone File*
- deepCz** The percent of requests to online Traffic Routers that were satisfied by a *Deep Coverage Zone File*
- dsr** The percent of requests to online Traffic Routers that were satisfied by sending the client to an overflow *Delivery Service*
- err** The percent of requests to online Traffic Routers that resulted in an error
- fed** The percent of requests to online Traffic Routers that were satisfied by sending the client to a federated CDN
- geo** The percent of requests to online Traffic Routers that were satisfied using 3rd party geographic IP mapping
- miss** The percent of requests to online Traffic Routers that could not be satisfied
- regionalAlternate** The percent of requests to online Traffic Routers that were satisfied by sending the client to the alternate, Regional Geo-blocking URL
- regionalDenied** The percent of requests to online Traffic Routers that were denied due to geographic location policy
- staticRoute** The percent of requests to online Traffic Routers that were satisfied with *Static DNS Entries*

### 262: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 21:29:32 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
↪GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
↪7LjytwKyRzSKM4cRIol40MIJxApFpTWJaSK73rbtUIQdASZjI64XxLVzZP00GRU7XeJ22YKUyQ30qbKHD
```

(continues on next page)

(continued from previous page)

Content-Length: 130

```
{ "response": {
  "cz": 79,
  "deepCz": 0.50,
  "dsr": 0,
  "err": 0,
  "fed": 0.25,
  "geo": 20,
  "miss": 0.25,
  "regionalAlternate": 0,
  "regionalDenied": 0,
  "staticRoute": 0
}}
```

## consistenthash

Test Pattern-Based Consistent Hashing for a *Delivery Service* using a regular expression and a request path

### POST

Queries database for an active Traffic Router on a given CDN and sends GET request to get the resulting path to consistent hash with a given regex and request path.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

**regex** The regular expression to apply to the request path to get a resulting path that will be used for consistent hashing

**requestPath** The request path to use to test the regular expression against

**cdnId** The unique identifier of a CDN that will be used to query for an active Traffic Router

### 263: Request Example

```
POST /api/3.0/consistenthash HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.54.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
Content-Length: 80
Content-Type: application/x-www-form-urlencoded

{"regex":"/.*?(/.*?/).*(m3u8)","requestPath":"/test/path/asset.m3u8",
  ↪ "cdnId":2}
```

## Response Structure

**resultingPathToConsistentHash** The resulting path that Traffic Router will use for consistent hashing

**consistentHashRegex** The regex used by Traffic Router derived from POST 'regex' parameter

**requestPath** The request path used by Traffic Router to test regex against

### 264: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, ↵
  ↪ Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54. ↵
  ↪ GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ↵
  ↪ QMDFOnUfqH4TcZ4YnUQyqnXDier0YiUMIfwBGdCT7ySjw9uASBGsLQW35lpnKFi4as0vYlHuSSGpe4hHG
X-Server-Name: traffic_ops_golang/
Date: Tue, 12 Feb 2019 21:32:05 GMT
Content-Length: 142

{ "response": {
  "resultingPathToConsistentHash":"/path/m3u8",
  "consistentHashRegex":"/.*?(/.*?/).*(m3u8)",
  "requestPath":"/test/path/asset.m3u8"
}}
```

## coordinates

### GET

Gets a list of all coordinates in the Traffic Ops database

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

Table 128: Request Query Parameters

Name	Re- quired	Description
id	no	Return only coordinates that have this integral, unique identifier
name	no	Return only coordinates with this name
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### Response Structure

**id** Integral, unique, identifier for this coordinate pair

**lastUpdated** The time and date at which this entry was last updated, in a ctime-like format

**latitude** Latitude of the coordinate

**longitude** Longitude of the coordinate

**name** The name of the coordinate - typically this just reflects the name of the Cache Group for which the coordinate was created

#### 265: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Y2vxC3hpxIg6aRNBBT7i2hbAViIJp+dJoqHIzu3acFM+vGay/
    ↳I5E+eZYOC9RY8hcJPrKNXysZOD8DOb9KsFgaw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:32:28 GMT
Content-Length: 942
```

```
{ "response": [
  {
    "id": 1,
    "name": "from_cachegroup_TRAFFIC_ANALYTICS",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 2,
    "name": "from_cachegroup_TRAFFIC_OPS",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 3,
    "name": "from_cachegroup_TRAFFIC_OPS_DB",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 4,
    "name": "from_cachegroup_TRAFFIC_PORTAL",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 5,
    "name": "from_cachegroup_TRAFFIC_STATS",
    "latitude": 38.897663,
```

(continues on next page)



(continued from previous page)

```

    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 6,
    "name": "from_cachegroup_CDN_in_a_Box_Mid",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 7,
    "name": "from_cachegroup_CDN_in_a_Box_Edge",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:05+00"
  }
]

```

## POST

Creates a new coordinate pair

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**name** The name of the new coordinate

**latitude** The desired latitude of the new coordinate (must be on the interval [-180, 180])

**longitude** The desired longitude of the new coordinate (must be on the interval [-90, 90])

### 266: Request Example

```

POST /api/3.0/coordinates HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 47

```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
```

```
{"name": "test", "latitude": 0, "longitude": 0}
```

## Response Structure

**id** Integral, unique, identifier for the newly created coordinate pair

**lastUpdated** The time and date at which this entry was last updated, in a ctime-like format

**latitude** Latitude of the newly created coordinate

**longitude** Longitude of the newly created coordinate

**name** The name of the coordinate

### 267: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 7pWdeZyIIXE1P7o/
→JVon+5eSCbDw+FGamAzdXzWHXJ8IhF+Vh+/
→tWfCkzHYw3rP2kBVwZu+gqLffjQpBCMjt7A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 17:48:55 GMT
Content-Length: 165

{ "alerts": [
  {
    "text": "coordinate was created.",
    "level": "success"
  }
],
"response": {
  "id": 9,
  "name": "test",
  "latitude": 0,
  "longitude": 0,
  "lastUpdated": "2018-11-15 17:48:55+00"
```

(continues on next page)

(continued from previous page)

```
}}
```

## PUT

Updates a coordinate

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 129: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the coordinate to edit

**name** The name of the new coordinate

**latitude** The desired new latitude of the coordinate (must be on the interval [-180, 180])

**longitude** The desired new longitude of the coordinate (must be on the interval [-90, 90])

## 268: Request Example

```
PUT /api/3.0/coordinates?id=9 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 48
Content-Type: application/json

{"name": "quest", "latitude": 0, "longitude": 0}
```

## Response Structure

- id** Integral, unique, identifier for the coordinate pair
- lastUpdated** The time and date at which this entry was last updated, in a ctime-like format
- latitude** Latitude of the coordinate
- longitude** Longitude of the coordinate
- name** The name of the coordinate

## 269: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳zd03Uvbnv8EbSZZ75Xp5tnnYStZsZTdyPxXnoqK4QZ5WhELLPL8iHlRf0aiLTbrUWUeJ8ue2HRz6aBS/
    ↳iXCCGA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 17:54:30 GMT
Content-Length: 166

{ "alerts": [
  {
    "text": "coordinate was updated.",
    "level": "success"
  }
]}
```

(continues on next page)

(continued from previous page)

```
],
"response": {
  "id": 9,
  "name": "quest",
  "latitude": 0,
  "longitude": 0,
  "lastUpdated": "2018-11-15 17:54:30+00"
}}
```

## DELETE

Deletes a coordinate

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 130: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the coordinate to delete

## Response Structure

### 270: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 82x/Wdckqgk4LN5LI1ZfBJ26xkDrUVUGDjs5QFa/
  ↳Lzap7dU30Zkjbv8XW41xeFYj8PDmxHIpb7hiV0bvLaxnEDA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 17:57:42 GMT
Content-Length: 65
```

(continues on next page)

(continued from previous page)

```
{ "alerts": [  
  {  
    "text": "coordinate was deleted.",  
    "level": "success"  
  }  
]}
```

### current\_stats

An API endpoint that returns current statistics for each CDN and an aggregate across them all.

#### GET

Retrieves current stats for each CDN. Also includes aggregate stats across them.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

No parameters available.

271: Request Example

```
GET /api/3.0/current_stats HTTP/1.1  
Host: trafficops.infra.ciab.test  
User-Agent: curl/7.47.0  
Accept: */*  
Cookie: mojolicious=...
```

### Response Structure

**cdn** The name of the CDN

**connections** Current number of TCP connections maintained

**capacity** 85 percent capacity of the CDN in Gps

**bandwidth** The total amount of bandwidth in Gbs

**Note:** If `cdn` name is `total` and capacity is omitted it represents the aggregate stats across CDNs

## 272: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=; Path=/; HttpOnly
Whole-Content-Sha512:
    ↳Rs3wgd7v5dP0b0Qs4I3J1q6mnWIMSM2AKSAWirK1kymvDY0oFISArF7Kkypgy10I34yn7FtFdMh6U7ABa
X-Server-Name: traffic_ops_golang/
Date: Thu, 14 Nov 2019 15:35:31 GMT
Content-Length: 138

{"response": {
  "currentStats": [
    {
      "bandwidth": null,
      "capacity": null,
      "cdn": "ALL",
      "connections": null
    },
    {
      "bandwidth": 0.000104,
      "capacity": 17,
      "cdn": "CDN-in-a-Box",
      "connections": 4
    },
    {
      "bandwidth": 0.000104,
      "cdn": "total",
      "connections": 4
    }
  ]
}}
```

### dbdump

**Caution:** This is an extremely dangerous thing to do, as it exposes the entirety of the database, including possibly sensitive information. Administrators and systems engineers are advised to instead use database-specific tools to make server transitions more securely.

Dumps the Traffic Ops database as an SQL script that should recreate its schema and contents exactly.

---

#### Implementation Detail

The script is output using the *pg\_dump(1)* utility, and is thus compatible for use with the *pg\_restore(1)* utility.

---

### GET

Fetches the database dump.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** undefined - outputs an SQL script, not JSON

### Request Structure

No parameters available



### 273: Request Example

```
GET /api/3.0/dbdump HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 274: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/sql
Content-Disposition: attachment
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳YwvPB0ZToyzT8ilBnDlWWdwV+E3f2Xgus10KrkNaipQqgrw5zGwq0rC1U9TZ8Zl6kAGcRZgCYnr1EWfHX
X-Server-Name: traffic_ops_golang/
Date: Mon, 09 Sep 2019 21:08:28 GMT
Transfer-Encoding: chunked

-- Actual text omitted - it's huge
```

## deliveryservice\_request\_comments

### GET

Gets delivery service request comments.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 131: Request Query Parameters

Name	Re-quired	Description
author	no	Filter for <i>Delivery Service Request</i> comments submitted by the user identified by this username
authorId	no	Filter for <i>Delivery Service Request</i> comments submitted by the user identified by this integral, unique identifier
deliveryServiceRequestId	no	Filter for <i>Delivery Service Request</i> comments submitted for the delivery service identified by this integral, unique identifier
id	no	Filter for the <i>Delivery Service Request</i> comment identified by this integral, unique identifier

### 275: Request Example

```
GET /api/3.0/deliveryservice_request_comments HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- author** The username of the user who created the comment.
- authorId** The integral, unique identifier of the user who created the comment.
- deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.
- id** The integral, unique identifier of the *DSR* comment.
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- value** The text of the comment that was posted.
- xmlId** This is the xmlId value that you provided in the request.

### 276: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:00:26.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→RaJZS1XFJ4oIxVKyyDjTuoQY7gPOmm5EuIL4AgHpyQpuaaNviw0XhGC4V/AKf/
→Ws6zXLgIUc40yvMsTxnriIww==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:00:26 GMT
Content-Length: 207

{
  "response": [
    {
      "authorId": 2,
      "author": "admin",
      "deliveryServiceRequestId": 2,
      "id": 3,
      "lastUpdated": "2020-02-24 19:59:46+00",
      "value": "Changing to a different origin for now.",
      "xmlId": "demo1"
    },
    {
      "authorId": 2,
      "author": "admin",
      "deliveryServiceRequestId": 2,
      "id": 4,
      "lastUpdated": "2020-02-24 19:59:55+00",
      "value": "Using HTTPS.",
      "xmlId": "demo1"
    }
  ]
}
```

## POST

Allows user to create a *Delivery Service Request* comment.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object

## Request Structure

**deliveryServiceRequestId** The integral, unique identifier of the delivery service that you are commenting on.

**value** The comment text itself.

**xmlId** This can be any string. It is not validated or used, though it is returned in the response.

### 277: Request Example

```
POST /api/3.0/deliveryservice_request_comments HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 111

{
  "deliveryServiceRequestId": 2,
  "value": "Does anyone have time to review my delivery service_
  ↪request?"
}
```

## Response Structure

**author** The username of the user who created the comment.

**authorId** The integral, unique identifier of the user who created the comment.

**deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.

**id** The integral, unique identifier of the *DSR* comment.

**lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*

**value** The text of the comment that was posted.

**xmlId** This is the xmlId value that you provided in the request.

### 278: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↪Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:02:20 GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: LiakFP6L7PrnF05kLXftx7WQoKn3bGpIJT5N15PvNG2sHridRMV3k23eRJM66ET0LcRfMOrQgRiydE+Xg
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:02:20 GMT
Content-Length: 223

{
  "alerts": [
    {
      "text": "deliveryservice_request_comment was created.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": 2,
    "author": null,
    "deliveryServiceRequestId": 2,
    "id": 6,
    "lastUpdated": "2020-02-24 20:02:20+00",
    "value": "Does anyone have time to review my delivery service request?",
    "xmlId": null
  }
}
```

## PUT

Updates a delivery service request comment.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object

## Request Structure

**deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.

**value** The comment text itself.

**xmlId** This can be any string. It is not validated or used, though it is returned in the response.

Table 132: Request Query Parameters

Parameter	Required	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> comment that you wish to update.

### 279: Request Example

```
PUT /api/3.0/deliveryservice_request_comments?id=6 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 166

{
  "deliveryServiceRequestId": 2,
  "value": "Update: We no longer need this, feel free to reject.\n\
→nDoes anyone have time to review my delivery service request?"
}
```

## Response Structure

**author** The username of the user who created the comment.

**authorId** The integral, unique identifier of the user who created the comment.

**deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.

**id** The integral, unique identifier of the *DSR* comment.

**lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*

**value** The text of the comment that was posted.

**xmlId** This is the xmlId value that you provided in the request.

## 280: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:05:46
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳RalS34imPw7c42nlnu5eTuv6FSxuGcAvxEdeIyNma1zpE3ZojAMFbhj8qi1s+hOVDYybfFPzMz82c+xc1
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:05:46 GMT
Content-Length: 255

{
  "alerts": [
    {
      "text": "deliveryservice_request_comment was updated.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": null,
    "author": null,
    "deliveryServiceRequestId": 2,
    "id": 6,
    "lastUpdated": "2020-02-24 20:05:46+00",
    "value": "Update: We no longer need this, feel free to reject.\n
    ↳\n\nDoes anyone have time to review my delivery service request?",
    "xmlId": null
  }
}

```

## DELETE

Deletes a delivery service request comment.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** undefined

## Request Structure

Table 133: Request Query Parameters

Parameter	Required	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> comment that you wish to delete.

### 281: Request Example

```
DELETE /api/3.0/deliveryservice_request_comments?id=6 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

### 282: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:07:40
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 10pGzqeIh/1jAx85mz3MI/
→5A1i1g5beTSLtfvgcfQmCjNKQvOMs/4TLviuVzOCRRrEIPmNcjy35tmvfxwlv7RMQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:07:40 GMT
Content-Length: 101

{
  "alerts": [
    {
      "text": "deliveryservice_request_comment was deleted.",
      "level": "success"
    }
  ]
}
```

(continues on next page)



(continued from previous page)

```
    ]  
  }
```

**deliveryservice\_requests**

**GET**

Retrieves *Delivery Service Requests*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 134: Request Query Parameters

Name	Re-quired	Description
as-signee	no	Filter for <i>Delivery Service Requests</i> that are assigned to the user identified by this username.
as-signeeId	no	Filter for <i>Delivery Service Requests</i> that are assigned to the user identified by this integral, unique identifier
author	no	Filter for <i>Delivery Service Requests</i> submitted by the user identified by this username
authorId	no	Filter for <i>Delivery Service Requests</i> submitted by the user identified by this integral, unique identifier
changeType	no	Filter for <i>Delivery Service Requests</i> of the change type specified. Can be create, update, or delete.
createdAt	no	Filter for <i>Delivery Service Requests</i> created on a certain date/time. Value must be <b>RFC 3339</b> compliant. Eg. 2019-09-19T19:35:38.828535Z
id	no	Filter for the <i>Delivery Service Request</i> identified by this integral, unique identifier.
status	no	Filter for <i>Delivery Service Requests</i> whose status is the status specified. The status can be draft, submitted, pending, rejected, or complete.
xmlId	no	Filter for <i>Delivery Service Requests</i> that have the given <i>xml_id</i> .
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

New in version ATCv6: The createdAt query parameter was added to this in endpoint across all API versions in ATC version 6.0.0.

### 283: Request Example

```
GET /api/3.0/deliveryservice_requests?status=draft HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- author** The username of the user who created the Delivery Service Request.
- authorId** The integral, unique identifier assigned to the author
- changeType** The change type of the *DSR*. It can be create, update, or delete....
- createdAt** The date and time at which the *DSR* was created, in *Traffic Ops's Custom Date/Time Format*.
- deliveryService** The delivery service that the *DSR* is requesting to update.
  - active** A boolean that defines *Active*.
  - anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*
  - cacheurl** A *Cache URL Expression*  
 Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later
  - ccrDnsTtl** The *DNS TTL* - named "ccrDnsTtl" for legacy reasons
  - cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs
  - cdnName** Name of the *CDN* to which the *Delivery Service* belongs
  - checkPath** A *Check Path*
  - consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*
  - consistentHashRegex** A *Consistent Hashing Regular Expression*
  - deepCachingType** The *Deep Caching* setting for this *Delivery Service*
  - displayName** The *Display Name*
  - dnsBypassCname** A *DNS Bypass CNAME*
  - dnsBypassIp** A *DNS Bypass IP*
  - dnsBypassIp6** A *DNS Bypass IPv6*
  - dnsBypassTtl** The *DNS Bypass TTL*
  - dscp** A *DSCP* to be used within the *Delivery Service*
  - ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>1</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL* Page 455, 1

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** An optional field containing the 2nd long description of this *Delivery Service*

**longDesc2** An optional field containing the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

---

<sup>1</sup> These fields must be defined if and only if **geoLimit** is non-zero

- setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
- type** The type of match performed using pattern.
- maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*
- maxOriginConnections** The *Max Origin Connections*
- midHeaderRewrite** A set of *Mid Header Rewrite Rules*
- missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*
- missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*
- multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*
- orgServerFqdn** The *Origin Server Base URL*
- originShield** A *Origin Shield* string
- profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated
- profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated
- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- routingName** The *Routing Name* of this *Delivery Service*

- signed** true if and only if `signingAlgorithm` is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenant** The name of the *Tenant* who owns this *Origin*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*
- id** The integral, unique identifier assigned to the *DSR*
- lastEditedBy** The username of user who last edited this *DSR*
- lastEditedById** The integral, unique identifier assigned to the user who last edited this *DSR*
- lastUpdated** The date and time at which the *DSR* was last updated, in *Traffic Ops's Custom Date/Time Format*.
- status** The status of the request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

#### 284: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➦Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:14:07 GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UBp3nklJr2x2cAW/
TKbhXMVJH6+OduxUaEBGbX4P7IahDk3VkaTd9LsQj01zgFEnZLwHrikpwFfn1UO32RAZOA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:14:07 GMT
Content-Length: 872
```

```
{
  "response": [
    {
      "authorId": 2,
      "author": "admin",
      "changeType": "update",
      "createdAt": "2020-02-24 19:11:12+00",
      "id": 1,
      "lastEditedBy": "admin",
      "lastEditedById": 2,
      "lastUpdated": "2020-02-24 19:11:12+00",
      "deliveryService": {
        "active": false,
        "anonymousBlockingEnabled": false,
        "cacheurl": null,
        "ccrDnsTtl": null,
        "cdnId": 2,
        "cdnName": "CDN-in-a-Box",
        "checkPath": null,
        "displayName": "Demo 1",
        "dnsBypassCname": null,
        "dnsBypassIp": null,
        "dnsBypassIp6": null,
        "dnsBypassTtl": null,
        "dscp": 0,
        "edgeHeaderRewrite": null,
        "firstHeaderRewrite": null,
        "geoLimit": 0,
        "geoLimitCountries": null,
        "geoLimitRedirectURL": null,
        "geoProvider": 0,
        "globalMaxMbps": null,
        "globalMaxTps": null,
        "httpBypassFqdn": null,
        "id": 1,
        "infoUrl": null,
        "initialDispersion": 1,
        "innerHeaderRewrite": null,

```

(continues on next page)

(continued from previous page)

```

"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "0001-01-01 00:00:00+00",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"longDesc1": null,
"longDesc2": null,
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"topology": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
  "http://video.demo1.mycdn.ciab.test",
  "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",

```

(continues on next page)



(continued from previous page)

```

        "trResponseHeaders": null,
        "trRequestHeaders": null,
        "consistentHashRegex": null,
        "consistentHashQueryParams": [
            "abc",
            "pdq",
            "xxx",
            "zyx"
        ],
        "maxOriginConnections": 0,
        "ecsEnabled": false
    },
    "status": "draft"
}
]
}

```

## POST

**Note:** This route does NOT do the same thing as *POST deliveryservices/request*.

Creates a new *Delivery Service Request*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object

## Request Structure

**changeType** The action that you want to perform on the delivery service. It can be “create”, “update”, or “delete”.

**status** The status of your request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

**deliveryService** The *Delivery Service* that you have submitted for review as part of this request.

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>Page 455, 1</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>Page 455, 1</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

- httpBypassFqdn** A *HTTP Bypass FQDN*
- id** An integral, unique identifier for this *Delivery Service*
- infoUrl** An *Info URL*
- initialDispersion** The *Initial Dispersion*
- innerHeaderRewrite** A set of *Inner Header Rewrite Rules*
- ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*
- lastHeaderRewrite** A set of *Last Header Rewrite Rules*
- lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*
- logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*
- longDesc** The *Long Description* of this *Delivery Service*
- longDesc1** An optional field containing the 2nd long description of this *Delivery Service*
- longDesc2** An optional field containing the 3rd long description of this *Delivery Service*
- matchList** The *Delivery Service's Match List*
- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
  - setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
  - type** The type of match performed using pattern.
- maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*
- maxOriginConnections** The *Max Origin Connections*
- midHeaderRewrite** A set of *Mid Header Rewrite Rules*
- missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*
- missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

**routingName** The *Routing Name* of this *Delivery Service*

**signed** true if and only if *signingAlgorithm* is not null, false otherwise

**signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*

**sslKeyVersion** This integer indicates the *SSL Key Version*

**tenant** The name of the *Tenant* who owns this *Origin*

**tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*

**topology** The unique name of the *Topology* that this *Delivery Service* is assigned to

**trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*

**trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*

**type** The *Type* of this *Delivery Service*

**typeId** The integral, unique identifier of the *Type* of this *Delivery Service*

**xmlId** This *Delivery Service*'s *xml\_id*

#### 285: Request Example

```
POST /api/3.0/deliveryservice_requests HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 1979

{
  "changeType": "update",
  "status": "draft",
  "deliveryService": {
    "active": false,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "displayName": "Demo 1",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "firstHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
    "globalMaxTps": null,
    "httpBypassFqdn": null,
    "id": 1,
    "infoUrl": null,
```

(continues on next page)

(continued from previous page)

```

"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "2020-02-13 16:43:54+00",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"longDesc1": null,
"longDesc2": null,
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
  "http://video.demo1.mycdn.ciab.test",
  "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,

```

(continues on next page)

(continued from previous page)

```

    "tenant": "root",
    "topology": null,
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": [
        "abc",
        "pdq",
        "xxx",
        "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false
}

```

## Response Structure

- author** The username of the user who created the Delivery Service Request.
- authorId** The integral, unique identifier assigned to the author
- changeType** The change type of the *DSR*. It can be create, update, or delete....
- createdAt** The date and time at which the *DSR* was created, in *Traffic Ops's Custom Date/Time Format*.
- deliveryService** The delivery service that the *DSR* is requesting to update.
  - active** A boolean that defines *Active*.
  - anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*
  - cacheurl** A *Cache URL Expression*
    - Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later
  - ccrDnsTtl** The *DNS TTL* - named "ccrDnsTtl" for legacy reasons
  - cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs
  - cdnName** Name of the *CDN* to which the *Delivery Service* belongs
  - checkPath** A *Check Path*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>Page 455, 1</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>Page 455, 1</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*



- lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*
- logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*
- longDesc** The *Long Description* of this *Delivery Service*
- longDesc1** An optional field containing the 2nd long description of this *Delivery Service*
- longDesc2** An optional field containing the 3rd long description of this *Delivery Service*
- matchList** The *Delivery Service's Match List*
- pattern** A regular expression - the use of this pattern is dependent on the *type* field (backslashes are escaped)
  - setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
  - type** The type of match performed using pattern.
- maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*
- maxOriginConnections** The *Max Origin Connections*
- midHeaderRewrite** A set of *Mid Header Rewrite Rules*
- missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*
- missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*
- multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*
- orgServerFqdn** The *Origin Server Base URL*
- originShield** A *Origin Shield* string
- profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated
- profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated

- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- routingName** The *Routing Name* of this *Delivery Service*
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenant** The name of the *Tenant* who owns this *Origin*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*
- id** The integral, unique identifier assigned to the *DSR*
- lastEditedBy** The username of user who last edited this *DSR*

**lastEditedById** The integral, unique identifier assigned to the user who last edited this *DSR*

**lastUpdated** The date and time at which the *DSR* was last updated, in *Traffic Ops's Custom Date/Time Format*.

**status** The status of the request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

#### 286: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:11:12.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: aWIrFTwUGnLq56WNZPL/FgOi/
→NwAVUtOy4iqjFPwx4gj7RMZ6+nd++bQKIiasBl8ytAY0WmFvNnmm30Fq9mLpA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:11:12 GMT
Content-Length: 901

{
  "alerts": [
    {
      "text": "deliveryservice_request was created.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": 2,
    "author": null,
    "changeType": "update",
    "createdAt": null,
    "id": 1,
    "lastEditedBy": null,
    "lastEditedById": 2,
    "lastUpdated": "2020-02-24 19:11:12+00",
    "deliveryService": {
      "active": false,
      "anonymousBlockingEnabled": false,
      "cacheurl": null,
      "ccrDnsTtl": null,
      "cdnId": 2,
```

(continues on next page)

(continued from previous page)

```
"cdnName": "CDN-in-a-Box",
"checkPath": null,
"displayName": "Demo 1",
"dnsBypassCname": null,
"dnsBypassIp": null,
"dnsBypassIp6": null,
"dnsBypassTtl": null,
"dscp": 0,
"edgeHeaderRewrite": null,
"firstHeaderRewrite": null,
"geoLimit": 0,
"geoLimitCountries": null,
"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "0001-01-01 00:00:00+00",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"longDesc1": null,
"longDesc2": null,
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
```

(continues on next page)

(continued from previous page)

```

    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "video",
    "signed": false,
    "sslKeyVersion": 1,
    "tenantId": 1,
    "topology": null,
    "type": "HTTP",
    "typeId": 1,
    "xmlId": "demo1",
    "exampleURLs": [
        "http://video.demo1.mycdn.ciab.test",
        "https://video.demo1.mycdn.ciab.test"
    ],
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": "root",
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": [
        "abc",
        "pdq",
        "xxx",
        "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false
},
"status": "draft"
}
}

```

## PUT

Updates an existing *Delivery Service Request*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object

## Request Structure

**changeType** The change type of the *DSR*. It can be create, update, or delete....

**deliveryService** The delivery service that the *DSR* is requesting to update.

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>Page 455, 1</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>Page 455, 1</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** An optional field containing the 2nd long description of this *Delivery Service*

**longDesc2** An optional field containing the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using pattern.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*



- remapText** *Raw Remap Text*
- routingName** The *Routing Name* of this *Delivery Service*
- signed** true if and only if **signingAlgorithm** is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenant** The name of the *Tenant* who owns this *Origin*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*
- id** The integral, unique identifier assigned to the *DSR*
- status** The status of the request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

Table 135: Request Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> that you want to update.

#### 287: Request Example

```
PUT /api/3.0/deliveryservice_requests?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
```

(continues on next page)

(continued from previous page)

Cookie: mojolicious=...

Content-Length: 2256

```
{
  "authorId": 2,
  "author": "admin",
  "changeType": "update",
  "createdAt": "2020-02-24 19:11:12+00",
  "id": 1,
  "lastEditedBy": "admin",
  "lastEditedById": 2,
  "lastUpdated": "2020-02-24 19:33:26+00",
  "deliveryService": {
    "active": false,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "displayName": "Demo 1",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "firstHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
    "globalMaxTps": null,
    "httpBypassFqdn": null,
    "id": 1,
    "infoUrl": null,
    "initialDispersion": 1,
    "innerHeaderRewrite": null,
    "ipv6RoutingEnabled": true,
    "lastHeaderRewrite": null,
    "lastUpdated": "0001-01-01 00:00:00+00",
    "logsEnabled": true,
    "longDesc": "Apachecon North America 2018",
    "longDesc1": null,
    "longDesc2": null,
  }
}
```

(continues on next page)

(continued from previous page)

```
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"topology": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
  "http://video.demo1.mycdn.ciab.test",
  "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"trResponseHeaders": "",
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [
  "abc",
  "pdq",
  "xxx",
```

(continues on next page)

(continued from previous page)

```

        "zyx"
      ],
      "maxOriginConnections": 0,
      "ecsEnabled": false
    },
    "status": "submitted"
  }

```

## Response Structure

**changeType** The change type of the *DSR*. It can be create, update, or delete....

**deliveryService** The delivery service that the *DSR* is requesting to update.

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>Page 455, 1</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>Page 455, 1</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** An optional field containing the 2nd long description of this *Delivery Service*

**longDesc2** An optional field containing the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using pattern.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

- remapText** *Raw Remap Text*
- routingName** The *Routing Name* of this *Delivery Service*
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenant** The name of the *Tenant* who owns this *Origin*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*
- id** The integral, unique identifier assigned to the *DSR*
- lastEditedBy** The username of user who last edited this *DSR*
- lastEditedById** The integral, unique identifier assigned to the user who last edited this *DSR*
- lastUpdated** The date and time at which the *DSR* was last updated, in *Traffic Ops's Custom Date/Time Format*.
- status** The status of the request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

#### 288: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:36:16.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +W0vFm96yFkZUJqa0GAX7uzIpRKh/
→ohyBm0uH3egpiERTcxy50fVVtoP3h8Ee2teLu8KFooDYXJ6rpQg6UhbNQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:36:16 GMT
Content-Length: 913
```

```
{
  "alerts": [
    {
      "text": "deliveryservice_request was updated.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": 0,
    "author": "admin",
    "changeType": "update",
    "createdAt": "0001-01-01 00:00:00+00",
    "id": 1,
    "lastEditedBy": "admin",
    "lastEditedById": 2,
    "lastUpdated": "2020-02-24 19:36:16+00",
    "deliveryService": {
      "active": false,
      "anonymousBlockingEnabled": false,
      "cacheurl": null,
      "ccrDnsTtl": null,
      "cdnId": 2,
      "cdnName": "CDN-in-a-Box",
      "checkPath": null,
      "displayName": "Demo 1",
      "dnsBypassCname": null,
      "dnsBypassIp": null,
      "dnsBypassIp6": null,
      "dnsBypassTtl": null,
      "dscp": 0,
      "edgeHeaderRewrite": null,
      "firstHeaderRewrite": null,
      "geoLimit": 0,
      "geoLimitCountries": null,
      "geoLimitRedirectURL": null,
```

(continues on next page)



(continued from previous page)

```
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "0001-01-01 00:00:00+00",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"longDesc1": null,
"longDesc2": null,
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.*)"
  ]
},
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"topology": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
```

(continues on next page)

(continued from previous page)

```

    "exampleURLs": [
      "http://video.demo1.mycdn.ciab.test",
      "https://video.demo1.mycdn.ciab.test"
    ],
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": "root",
    "trResponseHeaders": "",
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": [
      "abc",
      "pdq",
      "xxx",
      "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false
  },
  "status": "submitted"
}

```

## DELETE

Deletes a *Delivery Service Request*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** undefined

## Request Structure

Table 136: Request Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> that you want to delete.

#### 289: Request Example

```
DELETE /api/3.0/deliveryservice_requests?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

### Response Structure

#### 290: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:48:55
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳jNCbNo8Tw+JMMaWpAYQgntSXPq2Xuj+n2zSEVRaDQFWMV1SYbT9djes6SPdwiBoKq6W0lNE04h0E92jBV
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:48:55 GMT
Content-Length: 96

{
  "alerts": [
    {
      "text": "deliveryservice_request was deleted.",
      "level": "success"
    }
  ]
}
```

## deliveryservice\_requests/{{ID}}/assign

Assign a *Delivery Service Request* to a user.

### PUT

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

**id** The integral, unique identifier assigned to the *DSR*

**assignee** The username of the user to whom the *Delivery Service Request* is assigned.

#### 291: Request Example

```
PUT /api/3.0/deliveryservice_requests/1/assign HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 28

{
  "id": 1,
  "assigneeId": 2
}
```

### Response Structure

**assignee** The username of the user to whom the *Delivery Service Request* is assigned.

**author** The author of the *Delivery Service Request*

**authorId** The integral, unique identifier assigned to the author

**changeType** The change type of the *DSR*. It can be create, update, or delete....

**createdAt** The date and time at which the *DSR* was created, in *Traffic Ops’s Custom Date/Time Format*.

**deliveryService** The delivery service that the *DSR* is requesting to update.

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

- geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>1</sup>
- geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>Page 455, 1</sup>
- geoProvider** The *Geolocation Provider*
- globalMaxMbps** The *Global Max Mbps*
- globalMaxTps** The *Global Max TPS*
- httpBypassFqdn** A *HTTP Bypass FQDN*
- id** An integral, unique identifier for this *Delivery Service*
- infoUrl** An *Info URL*
- initialDispersion** The *Initial Dispersion*
- innerHeaderRewrite** A set of *Inner Header Rewrite Rules*
- ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*
- lastHeaderRewrite** A set of *Last Header Rewrite Rules*
- lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*
- logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*
- longDesc** The *Long Description* of this *Delivery Service*
- longDesc1** An optional field containing the 2nd long description of this *Delivery Service*
- longDesc2** An optional field containing the 3rd long description of this *Delivery Service*
- matchList** The *Delivery Service's Match List*
- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
  - setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
  - type** The type of match performed using pattern.

---

<sup>1</sup> These fields must be defined if and only if `geoLimit` is non-zero

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

**routingName** The *Routing Name* of this *Delivery Service*

**signed** true if and only if *signingAlgorithm* is not null, false otherwise

**signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*

**sslKeyVersion** This integer indicates the *SSL Key Version*

**tenant** The name of the *Tenant* who owns this *Origin*

- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*
- id** The integral, unique identifier assigned to the *DSR*
- lastEditedBy** The username of user who last edited this *DSR*
- lastEditedById** The integral, unique identifier assigned to the user who last edited this *DSR*
- lastUpdated** The date and time at which the *DSR* was last updated, in *Traffic Ops's Custom Date/Time Format*.
- status** The status of the request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

#### 292: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 14:45:51
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳h7uBZHLQtRYbOSOR5AtQQRZ4uMeEWivWNT74fCf6WtLbAMwGpRrMjNmBYKduv48DEnRqG6WVM/
    ↳4nBu3AkCUqPw==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 13:45:51 GMT
Content-Length: 931
    
```

(continues on next page)



(continued from previous page)

```
{
  "alerts": [
    {
      "text": "deliveryservice_request was updated.",
      "level": "success"
    }
  ],
  "response": {
    "assigneeId": 2,
    "assignee": "admin",
    "authorId": 2,
    "author": "admin",
    "changeType": "update",
    "createdAt": "2020-02-23 11:06:00+00",
    "id": 1,
    "lastEditedBy": "admin",
    "lastEditedById": 2,
    "lastUpdated": "2020-02-23 13:45:51+00",
    "deliveryService": {
      "active": true,
      "anonymousBlockingEnabled": false,
      "cacheurl": null,
      "ccrDnsTtl": null,
      "cdnId": 2,
      "cdnName": "CDN-in-a-Box",
      "checkPath": null,
      "displayName": "Demo 2",
      "dnsBypassCname": null,
      "dnsBypassIp": null,
      "dnsBypassIp6": null,
      "dnsBypassTtl": null,
      "dscp": 0,
      "edgeHeaderRewrite": null,
      "firstHeaderRewrite": null,
      "geoLimit": 0,
      "geoLimitCountries": null,
      "geoLimitRedirectURL": null,
      "geoProvider": 0,
      "globalMaxMbps": null,
      "globalMaxTps": null,
      "httpBypassFqdn": null,
      "id": 1,
      "infoUrl": null,
      "initialDispersion": 1,
      "innerHeaderRewrite": null,
      "ipv6RoutingEnabled": true,
```

(continues on next page)

(continued from previous page)

```

"lastHeaderRewrite": null,
"lastUpdated": "0001-01-01 00:00:00+00",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"longDesc1": null,
"longDesc2": null,
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": null,
"tenantId": 1,
"topology": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
    "http://video.demo1.mycdn.ciab.test",
    "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"trResponseHeaders": null,

```

(continues on next page)

(continued from previous page)

```

        "trRequestHeaders": null,
        "consistentHashRegex": null,
        "consistentHashQueryParams": [
            "abc",
            "pdq",
            "xxx",
            "zyx"
        ],
        "maxOriginConnections": 0,
        "ecsEnabled": false
    },
    "status": "submitted"
}

```

## deliveryservice\_requests/{{ID}}/status

Sets the status of a *Delivery Service Request*.

### PUT

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object

### Request Structure

**id** The integral, unique identifier assigned to the *DSR*

**status** The status of the *DSR* <*Delivery Service Request*>. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

#### 293: Request Example

```

PUT /api/3.0/deliveryservice_requests/1/status HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 28

{

```

(continues on next page)

(continued from previous page)

```
"id": 1,
"status": "rejected"
}
```

## Response Structure

**assignee** The username of the user to whom the *Delivery Service Request* is assigned.

**assigneeId** The integral, unique identifier of the user to whom the *Delivery Service Request* is assigned.

**author** The author of the *Delivery Service Request*

**authorId** The integral, unique identifier assigned to the author

**changeType** The change type of the *DSR*. It can be create, update, or delete....

**createdAt** The date and time at which the *DSR* was created, in *Traffic Ops's Custom Date/Time Format*.

**deliveryService** The delivery service that the *DSR* is requesting to update.

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named "ccrDnsTtl" for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>1</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL* Page 455, 1

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

---

<sup>1</sup> These fields must be defined if and only if `geoLimit` is non-zero

**longDesc1** An optional field containing the 2nd long description of this *Delivery Service*

**longDesc2** An optional field containing the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service*'s *Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using pattern.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- routingName** The *Routing Name* of this *Delivery Service*
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenant** The name of the *Tenant* who owns this *Origin*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*
- id** The integral, unique identifier assigned to the *DSR*
- lastEditedBy** The username of user who last edited this *DSR*
- lastEditedById** The integral, unique identifier assigned to the user who last edited this *DSR*
- lastUpdated** The date and time at which the *DSR* was last updated, in *Traffic Ops's Custom Date/Time Format*.
- status** The status of the request. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

294: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 15:54:53
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳C8Nhciy1jv5X7CGgHwAnLp1qmLizHq+4dv1AApb3cFSz5V2dABl7+N1Z4ndzB7GertB7rNLP31pVcat8v
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 14:54:53 GMT
Content-Length: 930

{
  "alerts": [
    {
      "text": "deliveryservice_request was updated.",
      "level": "success"
    }
  ],
  "response": {
    "assigneeId": 2,
    "assignee": "admin",
    "authorId": 2,
    "author": "admin",
    "changeType": "update",
    "createdAt": "2020-02-23 11:06:00+00",
    "id": 1,
    "lastEditedBy": "admin",
    "lastEditedById": 2,
    "lastUpdated": "2020-02-23 14:54:53+00",
    "deliveryService": {
      "active": true,
      "anonymousBlockingEnabled": false,
      "cacheurl": null,
      "ccrDnsTtl": null,
      "cdnId": 2,
      "cdnName": "CDN-in-a-Box",
      "checkPath": null,
      "displayName": "Demo 2",
      "dnsBypassCname": null,
      "dnsBypassIp": null,
    }
  }
}

```

(continues on next page)



(continued from previous page)

```

"dnsBypassIp6": null,
"dnsBypassTtl": null,
"dscp": 0,
"edgeHeaderRewrite": null,
"firstHeaderRewrite": null,
"geoLimit": 0,
"geoLimitCountries": null,
"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "0001-01-01 00:00:00+00",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"longDesc1": null,
"longDesc2": null,
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,

```

(continues on next page)

(continued from previous page)

```

        "routingName": "video",
        "signed": false,
        "sslKeyVersion": null,
        "tenantId": 1,
        "topology": null,
        "type": "HTTP",
        "typeId": 1,
        "xmlId": "demo1",
        "exampleURLs": [
            "http://video.demo1.mycdn.ciab.test",
            "https://video.demo1.mycdn.ciab.test"
        ],
        "deepCachingType": "NEVER",
        "fqPacingRate": null,
        "signingAlgorithm": null,
        "tenant": "root",
        "trResponseHeaders": null,
        "trRequestHeaders": null,
        "consistentHashRegex": null,
        "consistentHashQueryParams": [
            "abc",
            "pdq",
            "xxx",
            "zyx"
        ],
        "maxOriginConnections": 0,
        "ecsEnabled": false
    },
    "status": "rejected"
}

```

## deliveryservice\_stats

### GET

Retrieves time-aggregated statistics on a specific *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Object

<sup>1</sup> This endpoint respects *Tenancy*, and users whose *Tenant* does not have access to a *Delivery Service* will be unable to view the statistics of said *Delivery Service*.



## Request Structure

Table 137: Request Query Parameters

Name	Required	Description
deliveryService	yes <sup>2</sup>	Either the <i>xml_id</i> of a <i>Delivery Service</i> for which statistics will be aggregated or the integral, unique identifier of said <i>Delivery Service</i>
deliveryServiceName	yes <sup>?</sup>	The <i>xml_id</i> of the <i>Delivery Service</i> for which statistics will be aggregated
endDate	yes	The date and time until which statistics shall be aggregated in <b>RFC 3339</b> format (with or without sub-second precision), the number of nanoseconds since the Unix Epoch, or in the same, proprietary format as the <i>lastUpdated</i> fields prevalent throughout the Traffic Ops API
exclude	no	Either “series” to omit the data series from the result, or “summary” to omit the summary data from the result - directly corresponds to fields in the <i>Response Structure</i>
interval	no	Specifies the interval within which data will be “bucketed”; e.g. when requesting data from 2019-07-25T00:00:00Z to 2019-07-25T23:59:59Z with an interval of “1m”, the resulting data series (assuming it is not excluded) should contain $24 \frac{\text{hours}}{\text{day}} \times 60 \frac{\text{minutes}}{\text{hour}} \times 1 \text{day} \times 1 \frac{\text{minute}}{\text{datapoint}} = 1440 \text{data points}$ The allowed values for this parameter are valid InfluxQL duration literal strings matching <code>^d+[mhdw]\$</code>
limit	no	A natural number indicating the maximum amount of data points should be returned in the series
metricType	yes	The metric type being reported - one of: <b>kbps</b> The total traffic rate

## 295: Request Example

```
GET /api/3.0/deliveryservice_stats?deliveryServiceName=demo1&
→startDate=2019-07-22T17:55:00Z&endDate=2019-07-22T17:56:00.000Z&
→metricType=tps_total HTTP/1.1
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: application/json;timestamp=unix, application/json;
→timestamp=rfc;q=0.9, application/json;q=0.8, */*;q=0.7
Connection: keep-alive
Cookie: mojolicious=...
```

## Content Format

It's important to note in *Request Example* the use of a complex “Accept” header. This endpoint accepts two special media types in the “Accept” header that instruct it on how to format the timestamps associated with the returned data. Specifically, Traffic Ops will recognize the special, optional, non-standard parameter of *application/json: timestamp*. The values of this parameter are restricted to one of

**rfc** Returned timestamps will be formatted according to **RFC 3339** (no sub-second precision).

**unix** Returned timestamps will be formatted as the number of nanoseconds since the Unix Epoch (midnight on January 1<sup>st</sup> 1970 UTC).

## Implementation Detail

The endpoint passes back nanoseconds, specifically, because that is the form used both by InfluxDB, which is used to store the data being served, and Go's standard library. Clients may need to convert the value to match their own standard libraries - e.g. the `Date()` class in Javascript expects milliseconds.

The default behavior - when only e.g. *application/json* or *\*/\** is given - is to use **RFC 3339** formatting. It will, however, respect quality parameters. It is suggested that clients request timestamps they can handle specifically, rather than relying on this default behavior, as it **is subject to change** and is in fact **expected to invert in the next major release** as string-based time formats become deprecated.

## See also:

For more information on the “Accept” HTTP header, consult [its dedicated page on MDN](#).

<sup>2</sup> Either `deliveryServiceName` or `deliveryService` *must* be present, but if both are `deliveryServiceName` will be used and `deliveryService` will be ignored.

## Response Structure

**series** An object containing the actual data series and information necessary for working with it.

**columns** This is an array of names of the columns of the data contained in the “values” array - should always be `["time", "sum_count"]`

**count** The number of data points contained in the “values” array

**name** The name of the data set. Should always match `metric.ds.1min` where `metric` is the requested `metricType`

**values** The actual array of data points. Each represents a length of time specified by the `interval` query parameter

**time** The time at which the measurement was taken. This corresponds to the *beginning* of the interval. This time comes in the format of either an [RFC 3339](#)-formatted string, or a number containing the number of nanoseconds since the Unix Epoch depending on the “Accept” header sent by the client, according to the rules outlined in [Content Format](#).

**value** The value of the requested `metricType` at the time given by `time`. This will always be a floating point number, unless no data is available for the data interval, in which case it will be `null`

**source** A legacy field meant only for plugins that override this endpoint to name themselves. Should always be “TrafficStats”.

Deprecated since version 1.4: As this has no known purpose, developers are advised it will be removed in the future.

**summary** An object containing summary statistics describing the data series

**average** The arithmetic mean of the data’s values

**count** The number of measurements taken within the requested timespan. This is, in general, **not** the same as the `count` field of the `series` object, as it reflects the number of underlying, un-“bucketed” data points, and is therefore dependent on the implementation of Traffic Stats.

**fifthPercentile** Data points with values less than or equal to this number constitute the “bottom” 5% of the data set

**max** The maximum value that can be found in the requested data set

**min** The minimum value that can be found in the requested data set

**ninetyEighthPercentile** Data points with values greater than or equal to this number constitute the “top” 2% of the data set

**ninetyFifthPercentile** Data points with values greater than or equal to this number constitute the “top” 5% of the data set

**totalBytes** When the `metricType` requested is `kbps`, this will contain the total number of bytes transferred by the *Delivery Service* within the requested time window. Note that fractional amounts are possible, as the data transfer rate will almost certainly not be cleanly divided by the requested time range.

**totalTransactions** When the `metricType` requested is `not kbps`, this will contain the total number of transactions completed by the *Delivery Service* within the requested time window. Note that fractional amounts are possible, as the transaction rate will almost certainly not be cleanly divided by the requested time range.

**version** A legacy field that seems to have been meant to indicate the API version used. Will always be “1.2”

Deprecated since version 1.4: As this has no known purpose, developers are advised it will be removed in the future.

## 296: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: zXJGjcYuu6HxWINVp8HA1gL31J3ukry5wCsTDNxtP/
    ↳waC6rSD8h10KJ9jEAtrZJ9owOSVPvKaA/2bRu/QeuCpQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 22 Jul 2019 17:57:14 GMT
Transfer-Encoding: chunked
```

(continues on next page)

(continued from previous page)

```
{ "response": {
  "series": {
    "columns": [
      "time",
      "sum_count"
    ],
    "count": 2,
    "name": "tps_total.ds.1min",
    "tags": {
      "cachegroup": "total"
    },
    "values": [
      [
        15638181000000000000,
        0
      ],
      [
        15638181600000000000,
        0
      ]
    ]
  },
  "source": "TrafficStats",
  "summary": {
    "average": 0,
    "count": 2,
    "fifthPercentile": 0,
    "max": 0,
    "min": 0,
    "ninetyEighthPercentile": 0,
    "ninetyFifthPercentile": 0,
    "totalBytes": null,
    "totalTransactions": 0
  },
  "version": "1.2"
}}
```



## deliveryservices

### GET

Retrieves *Delivery Services*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

---

<sup>1</sup> Only those *Delivery Services* assigned to *Tenants* that are the requesting user's *Tenant* or children thereof will appear in the output of a GET request, and the same constraints are placed on the allowed values of the `tenantId` field of a POST request to create a new *Delivery Service*

## Request Structure

Table 138: Request Query Parameters

Name	Re-quired	Description
cdn	no	Show only the <i>Delivery Services</i> belonging to the <i>CDN</i> identified by this integral, unique identifier
id	no	Show only the <i>Delivery Service</i> that has this integral, unique identifier
logsEnabled	no	Show only the <i>Delivery Services</i> that have <i>Logs Enabled</i> set or not based on this boolean
profile	no	Return only <i>Delivery Services</i> using the <i>Profile</i> that has this <i>ID</i>
tenant	no	Show only the <i>Delivery Services</i> belonging to the <i>Tenant</i> identified by this integral, unique identifier
topology	no	Show only the <i>Delivery Services</i> assigned to the <i>Topology</i> identified by this unique name
type	no	Return only <i>Delivery Services</i> of the <i>Delivery Service Type</i> identified by this integral, unique identifier
accessibleTo	no	Return the <i>Delivery Services</i> accessible from a <i>Tenant</i> or it's children identified by this integral, unique identifier
service-Category	no	Show only the <i>Delivery Services</i> belonging to the <i>Service Category</i> that has this name
xmlId	no	Show only the <i>Delivery Service</i> that has this text-based, unique identifier
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.
active	no	Show only the <i>Delivery Services</i> that have <i>Active</i> set or not based on this boolean (whether or not they are active)

## Response Structure

- active** A boolean that defines *Active*.
- anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*
- cacheurl** A *Cache URL Expression*  
 Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later
- ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons
- cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs
- cdnName** Name of the *CDN* to which the *Delivery Service* belongs
- checkPath** A *Check Path*
- consistentHashRegex** A *Consistent Hashing Regular Expression*
- consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*
- deepCachingType** The *Deep Caching* setting for this *Delivery Service*
- displayName** The *Display Name*
- dnsBypassCname** A *DNS Bypass CNAME*
- dnsBypassIp** A *DNS Bypass IP*
- dnsBypassIp6** A *DNS Bypass IPv6*
- dnsBypassTtl** The *DNS Bypass TTL*
- dscp** A *DSCP* to be used within the *Delivery Service*
- ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*
- edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*
- exampleURLs** An array of *Example URLs*
- firstHeaderRewrite** A set of *First Header Rewrite Rules*
- fqPacingRate** The *Fair-Queuing Pacing Rate Bps*
- geoLimit** An integer that defines the *Geo Limit*
- geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*
- geoLimitRedirectUrl** A *Geo Limit Redirect URL*
- geoProvider** The *Geolocation Provider*
- globalMaxMbps** The *Global Max Mbps*
- globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** The 2nd long description of this *Delivery Service*

**longDesc2** the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
- setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
- type** The type of match performed using **pattern**.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

- profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated
- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

#### 297: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: mCLMjvACRKHNGP/
→OSx4javkOtxxzyiDdQzsV78IamUhVmvKyKaCeOKRmpsG69w+nhh30kPZ6e9MMeJpcJSKcA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 19:04:29 GMT
Transfer-Encoding: chunked
```

```
{ "response": [{
  "active": true,
  "anonymousBlockingEnabled": false,
  "cacheurl": null,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "checkPath": null,
  "displayName": "Demo 1",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "edgeHeaderRewrite": null,
  "firstHeaderRewrite": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectURL": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "id": 1,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": true,
  "lastHeaderRewrite": null,
  "lastUpdated": "2019-05-15 14:32:05+00",
  "logsEnabled": true,
  "longDesc": "Apachecon North America 2018",
  "longDesc1": null,
  "longDesc2": null,
  "matchList": [
```

(continues on next page)

(continued from previous page)

```
{
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
}
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": null,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
    "http://video.demo1.mycdn.ciab.test",
    "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"trResponseHeaders": null,
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [
    "abc",
    "pdq",
    "xxx",
    "zyx"
],
],
```

(continues on next page)

(continued from previous page)

```
"maxOriginConnections": 0,
"maxRequestHeaderBytes": 131072,
"ecsEnabled": false,
"rangeSliceBlockSize": null,
"topology": null
"serviceCategory": null
}}}
```

## POST

Allows users to create *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 455, 1</sup>

**Response Type** Array

## Request Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*



**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>2</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>?</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** An optional field containing the 2nd long description of this *Delivery Service*

**longDesc2** An optional field containing the 3rd long description of this *Delivery Service*

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

---

<sup>2</sup> These fields must be defined if and only if `geoLimit` is non-zero

- orgServerFqdn** The *Origin Server Base URL*
- originShield** A *Origin Shield* string
- profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated - or null if there is to be no such category
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3. It can only be between (inclusive) 262144 (256KB) - 33554432 (32MB).
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

298: Request Example

```
POST /api/3.0/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 761
Content-Type: application/json

{
  "active": false,
  "anonymousBlockingEnabled": false,
  "cdnId": 2,
  "deepCachingType": "NEVER",
  "displayName": "test",
  "dscp": 0,
  "ecsEnabled": true,
  "geoLimit": 0,
  "geoProvider": 0,
  "initialDispersion": 1,
  "ipv6RoutingEnabled": false,
  "logsEnabled": true,
  "longDesc": "A Delivery Service created expressly for API_
→documentation examples",
  "missLat": 0,
  "missLong": 0,
  "maxOriginConnections": 0,
  "maxRequestHeaderBytes": 131072,
  "multiSiteOrigin": false,
  "orgServerFqdn": "http://origin.infra.ciab.test",
  "protocol": 0,
  "qstringIgnore": 0,
  "rangeRequestHandling": 0,
  "regionalGeoBlocking": false,
  "routingName": "test",
  "serviceCategory": null,
  "signed": false,
  "tenant": "root",
  "tenantId": 1,
  "typeId": 1,
  "xmlId": "test"
}
```

## Response Structure

- active** A boolean that defines *Active*.
- anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*
- cacheurl** A *Cache URL Expression*  
 Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later
- ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons
- cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs
- cdnName** Name of the *CDN* to which the *Delivery Service* belongs
- checkPath** A *Check Path*
- consistentHashRegex** A *Consistent Hashing Regular Expression*
- consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*
- deepCachingType** The *Deep Caching* setting for this *Delivery Service*
- displayName** The *Display Name*
- dnsBypassCname** A *DNS Bypass CNAME*
- dnsBypassIp** A *DNS Bypass IP*
- dnsBypassIp6** A *DNS Bypass IPv6*
- dnsBypassTtl** The *DNS Bypass TTL*
- dscp** A *DSCP* to be used within the *Delivery Service*
- ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*
- edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*
- exampleURLs** An array of *Example URLs*
- firstHeaderRewrite** A set of *First Header Rewrite Rules*
- fqPacingRate** The *Fair-Queuing Pacing Rate Bps*
- geoLimit** An integer that defines the *Geo Limit*
- geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*
- geoLimitRedirectUrl** A *Geo Limit Redirect URL*
- geoProvider** The *Geolocation Provider*
- globalMaxMbps** The *Global Max Mbps*
- globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** The 2nd long description of this *Delivery Service*

**longDesc2** the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using **pattern**.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

- profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated
- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

#### 299: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: SVveQ5hGwfPv8N5APUskwLOzwrTUVA+z8wuFLsSLCr1/
→vVnFJJ0VQOGMUctg1NbqhAuQ795MJmuuAaAwR8dSOQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 19 Nov 2018 19:45:49 GMT
Content-Length: 1404

{ "alerts": [
  {
    "text": "Deliveryservice creation was successful.",
    "level": "success"
  }
],
"response": [
  {
    "active": false,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "displayName": "test",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "firstHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
    "globalMaxTps": null,
    "httpBypassFqdn": null,
    "id": 2,
    "infoUrl": null,
    "initialDispersion": 1,
    "innerHeaderRewrite": null,
    "ipv6RoutingEnabled": false,
```

(continues on next page)

(continued from previous page)

```

    "lastHeaderRewrite": null,
    "lastUpdated": "2018-11-19 19:45:49+00",
    "logsEnabled": true,
    "longDesc": "A Delivery Service created expressly for API_
→documentation examples",
    "longDesc1": null,
    "longDesc2": null,
    "matchList": [
        {
            "type": "HOST_REGEX",
            "setNumber": 0,
            "pattern": ".*\\.test\\.\\.*"
        }
    ],
    "maxDnsAnswers": null,
    "maxOriginConnections": 0,
    "maxRequestHeaderBytes": 131072,
    "midHeaderRewrite": null,
    "missLat": -1,
    "missLong": -1,
    "multiSiteOrigin": false,
    "originShield": null,
    "orgServerFqdn": "http://origin.infra.ciab.test",
    "profileDescription": null,
    "profileId": null,
    "profileName": null,
    "protocol": 0,
    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "test",
    "serviceCategory": null,
    "signed": false,
    "sslKeyVersion": null,
    "tenantId": 1,
    "type": "HTTP",
    "typeId": 1,
    "xmlId": "test",
    "exampleURLs": [
        "http://test.test.mycdn.ciab.test"
    ],
    "deepCachingType": "NEVER",
    "signingAlgorithm": null,
    "tenant": "root",

```

(continues on next page)



(continued from previous page)

```

    "ecsEnabled": true,
    "rangeSliceBlockSize": null,
    "topology": null
  }
}]

```

## deliveryservices/dnsseckey/generate

### POST

Generates ZSK and KSK keypairs for a CDN and all associated *Delivery Services*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object (string)

### Request Structure

**effectiveDate** UNIX epoch start date for the signing keys

**key** Name of the CDN

**kskExpirationDays** Expiration (in days) for the KSKs

**name** Domain name used by the CDN

**ttl** Time for which the keypairs shall remain valid

**zskExpirationDays** Expiration (in days) for the ZSKs

### Response Structure

### 300: Response Example

```
{
  "response": "Successfully created dnssec keys for cdn1"
}
```

**deliveryservices/{ID}**

**PUT**

Allows users to edit an existing *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Response Type** **NOT PRESENT** - Despite returning a 200 OK response (rather than e.g. a 204 NO CONTENT response), this endpoint does **not** return a representation of the modified resource in its payload, and instead returns nothing - not even a success message.

### Request Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

---

**Note:** If the Delivery Service has SSL Keys, then cdnId is not allowed to change as that would invalidate the SSL Key

---

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

---

<sup>1</sup> Only those *Delivery Services* assigned to *Tenants* that are the requesting user’s *Tenant* or children thereof will appear in the output of a GET request, and the same constraints are placed on the allowed values of the tenantId field of a PUT request to update a new *Delivery Service*. Furthermore, the only *Delivery Services* a user may delete are those assigned to a *Tenant* that is either the same *Tenant* as the user’s *Tenant*, or a descendant thereof.

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*<sup>2</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>?</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** An optional field containing the 2nd long description of this *Delivery Service*

**longDesc2** An optional field containing the 3rd long description of this *Delivery Service*

---

<sup>2</sup> These fields must be defined if and only if `geoLimit` is non-zero

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileId** An optional *ID* of the *Profile* with which this *Delivery Service* will be associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

**routingName** The *Routing Name* of this *Delivery Service*

---

**Note:** If the Delivery Service has SSL Keys, then routingName is not allowed to change as that would invalidate the SSL Key

---

**signed** true if and only if signingAlgorithm is not null, false otherwise

**signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*

**rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if rangeRequestHandling is set to 3. It can only be between (inclusive) 262144 (256KB) - 33554432 (32MB).

**sslKeyVersion** This integer indicates the *SSL Key Version*

**tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*

**topology** The unique name of the *Topology* that this *Delivery Service* is assigned to

**trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*

**trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*

**typeId** The integral, unique identifier of the *Type* of this *Delivery Service*

**xmlId** This *Delivery Service*'s *xml\_id*

---

**Note:** While this field **must** be present, it is **not** allowed to change; this must be the same as the `xml_id` the *Delivery Service* already has. This should almost never be different from the *Delivery Service*'s `displayName`.

---

### 301: Request Example

```
PUT /api/3.0/deliveryservices/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 761
Content-Type: application/json

{
  "active": true,
  "anonymousBlockingEnabled": false,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "deepCachingType": "NEVER",
  "displayName": "demo",
  "dscp": 0,
  "ecsEnabled": true,
  "geoLimit": 0,
  "geoProvider": 0,
  "initialDispersion": 1,
  "ipv6RoutingEnabled": false,
  "lastUpdated": "2018-11-14 18:21:17+00",
  "logsEnabled": true,
  "longDesc": "A Delivery Service created expressly for API_
↪documentation examples",
  "missLat": -1,
  "missLong": -1,
  "multiSiteOrigin": false,
  "orgServerFqdn": "http://origin.infra.ciab.test",
  "protocol": 0,
  "qstringIgnore": 0,
  "rangeRequestHandling": 0,
```

(continues on next page)

(continued from previous page)

```
"regionalGeoBlocking": false,
"routingName": "video",
"signed": false,
"tenant": "root",
"tenantId": 1,
"typeId": 1,
"xmlId": "demo1"
}
```

## Response Structure

### 302: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: z4PhNX7vuL3xVChQ1m2AB9Yg5AULVxXcg/
    ↳SpIdNs6c5H0NE8XYXysP+DGNKHfuwvY7kxvUdBeoGlODJ6+SfaPg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 20 Nov 2018 14:12:25 GMT
Content-Length: 0
Content-Type: text/plain; charset=utf-8
```

## DELETE

Deletes the target *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 455, 1</sup>

**Response Type** undefined

## Request Structure

Table 139: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> to be deleted

### 303: Request Example

```
DELETE /api/3.0/deliveryservices/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 304: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: w9NlQpJJEl56r6iYq/
→fk8o5WfAXeUS5XR9yDHvKUgPO8lYEo8YyftaSF0MPFseeOk60dk6kQo+MLYTDIAhhRxw==
X-Server-Name: traffic_ops_golang/
Date: Tue, 20 Nov 2018 14:56:37 GMT
Content-Length: 57

{ "alerts": [
  {
    "text": "ds was deleted.",
    "level": "success"
  }
]}
```

`deliveryservices/{{ID}}/capacity`

See also:

*Health Protocol*

GET

Retrieves the usage percentages of a servers associated with a *Delivery Service*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Object

## Request Structure

Table 140: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the <i>Delivery Service</i> of interest

## Response Structure

**availablePercent** The percent of servers assigned to this *Delivery Service* that is available - the allowed traffic level in terms of data per time period for all *cache servers* that remains unused

**unavailablePercent** The percent of servers assigned to this *Delivery Service* that is unavailable - the allowed traffic level in terms of data per time period for all *cache servers* that can't be used because the servers are deemed unhealthy

**utilizedPercent** The percent of servers assigned to this *Delivery Service* that is currently in use - the allowed traffic level in terms of data per time period that is currently devoted to servicing requests

**maintenancePercent** The percent of servers assigned to this *Delivery Service* that is unavailable due to server maintenance - the allowed traffic level in terms of data per time period that is unavailable because servers have intentionally been marked offline by administrators

### 305: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept
```

(continues on next page)

<sup>1</sup> Users will only be able to see capacity details for the *Delivery Services* their *Tenant* is allowed to see.



(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 14:41:27 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→++dFR9V1c60CHGNwMjX6JSFEjHreXcL4QnhT03hiv04ByY379aLpL40rOzX2bPgJgpR94+f6jZ0+iDIyT
Content-Length: 134

{ "response": {
  "availablePercent": 99.9993696969697,
  "unavailablePercent": 0,
  "utilizedPercent": 0.00063030303030303,
  "maintenancePercent": 0
}}
```

## deliveryservices/{{ID}}/health

See also:

*Health Protocol*

### GET

Retrieves the health of all *Cache Groups* assigned to a particular *Delivery Service*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Object

<sup>1</sup> Users will only be able to see *Cache Group* health details for the *Delivery Services* their *Tenant* is allowed to see.

## Request Structure

Table 141: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the Delivery service for which <i>Cache Groups</i> will be displayed

## Response Structure

**cachegroups** An array of objects that represent the health of each *Cache Group* assigned to this *Delivery Service*

**name** A string that is the *name of the Cache Group* represented by this object

**offline** The number of OFFLINE *cache servers* within this *Cache Group*

**online** The number of ONLINE *cache servers* within this *Cache Group*

**totalOffline** Total number of OFFLINE *cache servers* assigned to this *Delivery Service*

**totalOnline** Total number of ONLINE *cache servers* assigned to this *Delivery Service*

### 306: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 14:43:43 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→KpXViXeAgch58ueQqdyU8NuINBw1EUedE6Rv2ewcLUajJp6kowdbVynpwW7XiSvAyHdtClIOuT30khIm
Content-Length: 115

{ "response": {
  "totalOffline": 0,
```

(continues on next page)

(continued from previous page)

```

    "totalOnline": 1,
    "cachegroups": [
      {
        "offline": 0,
        "name": "CDN_in_a_Box_Edge",
        "online": 1
      }
    ]
  }
}
```

## deliveryservices/{{ID}}/regexes

### GET

Retrieves routing regular expressions for a specific *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

### Request Structure

Table 142: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected

Table 143: Request Query Parameters

Name	Re- quired	Description
id	no	Show only the <i>Delivery Service</i> regular expression that has this integral, unique identifier
limit	no	Choose the maximum number of results to return
off- set	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <sup>n</sup> th page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

<sup>1</sup> Users will only be able to view and create regular expressions for the *Delivery Services* their *Tenant* is allowed to see.

### 307: Request Example

```
GET /api/3.0/deliveryservices/1/regexes HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- id** The integral, unique identifier of this regular expression
- pattern** The actual regular expression - \s are escaped
- setNumber** The order in which the regular expression is evaluated against requests
- type** The integral, unique identifier of the *Type* of this regular expression
- typeName** The *Type* of regular expression - determines that against which it will be evaluated

### 308: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: fw9Fde4WRpp2ShRAC41P9s/PhU71LI/
↳SEzHgYjGqfzhk45wq0kpaWy76JvPfLpowY8eDTp8Y8TL5rNGEc+bM+A==
X-Server-Name: traffic_ops_golang/
Date: Tue, 27 Nov 2018 20:56:43 GMT
Content-Length: 100

{ "response": [
  {
    "id": 1,
    "type": 31,
    "typeName": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
]
}]
```

## POST

Creates a routing regular expression for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 455, 1</sup>

**Response Type** Object

## Request Structure

Table 144: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected

**pattern** The actual regular expression

**Warning:** Be sure that \s are escaped, or the expression may not work as intended!

**setNumber** The order in which this regular expression should be checked

**type** The integral, unique identifier of a routing regular expression type

### 309: Request Example

```
POST /api/3.0/deliveryservices/1/regexes HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 55
Content-Type: application/json

{
  "pattern": ".*\\.foo-bar\\.\\.\\.\"",
  "type": 31,
  "setNumber": 1
}
```

## Response Structure

- id** The integral, unique identifier of this regular expression
- pattern** The actual regular expression - \s are escaped
- setNumber** The order in which the regular expression is evaluated against requests
- type** The integral, unique identifier of the type of this regular expression
- typeName** The type of regular expression - determines that against which it will be evaluated

### 310: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: kS5dRzAhFKE7vfzHK7XVIwpMOjztksk9MU+qtj5YU/
  ↳1oxVHmqNbJ12Fe00IJsZJCXbYlnBS04sCI95Sz5wed1Q==
X-Server-Name: traffic_ops_golang/
Date: Wed, 28 Nov 2018 17:00:42 GMT
Content-Length: 188

{ "alerts": [
  {
```

(continues on next page)

(continued from previous page)

```

        "text": "Delivery service regex creation was successful.",
        "level": "success"
    }
],
"response": {
    "id": 2,
    "type": 31,
    "typeName": "HOST_REGEX",
    "setNumber": 1,
    "pattern": ".*\\.foo-bar\\.\\.\\.*"
}}

```

deliveryservices/{{ID}}/regexes/{{rID}}

PUT

Updates a routing regular expression.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Response Type** Object

## Request Structure

Table 145: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected
rID	The integral, unique identifier of the routing regular expression being inspected

**pattern** The actual regular expression

**Warning:** Be sure that \s are escaped, or the expression may not work as intended!

**setNumber** The order in which this regular expression should be checked

**type** The integral, unique identifier of a routing regular expression type

<sup>1</sup> Users will only be able to view, delete and update regular expressions for the *Delivery Services* their *Tenant* is allowed to see.

### 311: Request Example

```
PUT /api/3.0/deliveryservices/1/regexes/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 55
Content-Type: application/json

{
  "pattern": ".*\\.foo-bar\\.\\.\\.\"",
  "type": 33,
  "setNumber": 1
}
```

## Response Structure

- id** The integral, unique identifier of this regular expression
- pattern** The actual regular expression - \s are escaped
- setNumber** The order in which the regular expression is evaluated against requests
- type** The integral, unique identifier of the type of this regular expression
- typeName** The type of regular expression - determines that against which it will be evaluated

### 312: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: kS5dRzAhFKE7vfzHK7XVIwpMOjztksk9MU+qtj5YU/
  ↳1oxVHmqNbJ12Fe00IJsZJCXbYlnBS04sCI95Sz5wed1Q==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 17:54:58 GMT
Content-Length: 188

{ "alerts": [
  {
```

(continues on next page)



(continued from previous page)

```

        "text": "Delivery service regex creation was successful.",
        "level": "success"
    }
],
"response": {
    "id": 2,
    "type": 33,
    "typeName": "PATH_REGEX",
    "setNumber": 1,
    "pattern": ".*\\.foo-bar\\.\\.*"
}}

```

## DELETE

Deletes a routing regular expression.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 455, 1</sup>

**Response Type** undefined

## Request Structure

Table 146: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected
rID	The integral, unique identifier of the routing regular expression being inspected

### 313: Request Example

```
DELETE /api/3.0/deliveryservices/1/regexes/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 314: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 8oEa78x7f/o39LIS98W6G+UqE6cX/
    ↳Iw4v3mMHvbAsliWHALuDYZRz3V0tA6jzfGQKpB040m8qaVG+zWRrBVoCmQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 18:44:00 GMT
Content-Length: 76

{ "alerts": [
  {
    "text": "deliveryservice_regex was deleted.",
    "level": "success"
  }
]}
```

## deliveryservices/{{ID}}/routing

### GET

Retrieves the aggregated routing percentages for a given *Delivery Service*. This is accomplished by polling stats from all online Traffic Routers via the */crs/stats* route.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

---

<sup>1</sup> Users will only be able to view routing details for the *Delivery Services* their *Tenant* is allowed to see.

## Response Type Object

**Note:** Only HTTP or DNS *Delivery Services* can be requested.

## Request Structure

Table 147: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the <i>Delivery Service</i> of interest

### 315: Request Example

```
GET /api/3.0/deliveryservices/1/routing HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cz** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by a *Coverage Zone File*
- deepCz** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by a *Deep Coverage Zone File*
- dsr** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by sending the client to an overflow *Delivery Service*
- err** The percent of requests to online Traffic Routers for this *Delivery Service* that resulted in an error
- fed** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by sending the client to a federated CDN
- geo** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied using 3rd party geographic IP mapping
- miss** The percent of requests to online Traffic Routers for this *Delivery Service* that could not be satisfied
- regionalAlternate** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by sending the client to the alternate, Regional Geo-blocking URL
- regionalDenied** The percent of requests to online Traffic Routers for this *Delivery Service* that were denied due to geographic location policy

**staticRoute** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied with *Static DNS Entries*

### 316: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Fri, 30 Nov 2018 15:08:07 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: UgPziRC/5u4+CfkZ9xm0EkEzjjJVu6cwBrFd/n3xH/
→ZmlkaXkQaa1y4+B7DyE46vxFLYE0OD0cQchyn7JkoQOg==
Content-Length: 132

{ "response": {
  "cz": 79,
  "deepCz": 0.50,
  "dsr": 0,
  "err": 0,
  "fed": 0.25,
  "geo": 20,
  "miss": 0.25,
  "regionalAlternate": 0,
  "regionalDenied": 0,
  "staticRoute": 0
}}
```

`deliveryservices/{{ID}}/safe`

**PUT**

Allows a user to edit metadata fields of a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

<sup>1</sup> Only those *Delivery Services* assigned to *Tenants* that are the requesting user's *Tenant* or children thereof may be modified with this endpoint.

## Request Structure

Table 148: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being modified

**displayName** A string that is the *Display Name*

**infoUrl** An optional<sup>2</sup> string containing the *Info URL*

**longDesc** An optional<sup>2</sup> string containing the *Long Description* of this *Delivery Service*

**longDesc1** An optional<sup>2</sup> string containing the 2nd long description of this *Delivery Service*

### 317: Request Example

```
PUT /api/3.0/deliveryservices/1/safe HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 132

{
  "displayName": "test",
  "infoUrl": "this is not even a real URL",
  "longDesc": "longDesc1 is implicitly set to null in this example"
}
```

## Response Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**cacheurl** A *Cache URL Expression*

Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

<sup>2</sup> If these fields are not present in the request body they are *implicitly set to null*.

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** The 2nd long description of this *Delivery Service*

**longDesc2** the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using **pattern**.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

**signed** true if and only if `signingAlgorithm` is not null, false otherwise

**signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*

**rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if `rangeRequestHandling` is set to 3.

**sslKeyVersion** This integer indicates the *SSL Key Version*

**tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*

**topology** The unique name of the *Topology* that this *Delivery Service* is assigned to

**trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*

**trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*

**type** The *Type* of this *Delivery Service*

**typeId** The integral, unique identifier of the *Type* of this *Delivery Service*

**xmlId** This *Delivery Service*'s *xml\_id*

### 318: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Feb 2020 16:33:03.
→GMT; Max-Age=3600; HttpOnly
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Feb 2020 15:33:03 GMT
Content-Length: 853

{ "alerts": [
  {
    "text": "Delivery Service safe update successful.",
    "level": "success"
  }
],
"response": [
  {
    "active": true,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
```

(continues on next page)



(continued from previous page)

```

"cdnId": 2,
"cdnName": "CDN-in-a-Box",
"checkPath": null,
"displayName": "test",
"dnsBypassCname": null,
"dnsBypassIp": null,
"dnsBypassIp6": null,
"dnsBypassTtl": null,
"dscp": 0,
"edgeHeaderRewrite": null,
"firstHeaderRewrite": null,
"geoLimit": 0,
"geoLimitCountries": null,
"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": "this is not even a real URL",
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "2020-02-10 15:33:03+00",
"logsEnabled": true,
"longDesc": "longDesc1 is implicitly set to null in this_
→example",
"longDesc1": null,
"longDesc2": null,
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,

```

(continues on next page)

(continued from previous page)

```
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
    "http://video.demo1.mycdn.ciab.test",
    "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"trResponseHeaders": null,
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [
    "abc",
    "pdq",
    "xxx",
    "zyx"
],
"maxOriginConnections": 0,
"ecsEnabled": false,
"rangeSliceBlockSize": null,
"topology": null
}
]}
```

`deliveryservices/{{id}}/server_types/{{type}}/metric_types/start_date/{{start}}/end_date/{{end}}`

**Danger:** This endpoint doesn't appear to work, and so its use is strongly discouraged! The below documentation cannot be verified, therefore it may be inaccurate and/or incomplete!

## GET

Retrieves detailed and summary metrics for Mid-tier and Edge-tier caches assigned to a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 149: Request Path Parameters

Name	Re-quired	Description
id	yes	The delivery service id.
server_types	yes	EDGE or MID.
metric_types	yes	One of the following: “kbps”, “tps”, “tps_2xx”, “tps_3xx”, “tps_4xx”, “tps_5xx”.
start_date	yes	UNIX time
end_date	yes	UNIX time

Table 150: Request Query Parameters

Name	Required	Description
stats	no	Flag used to return only summary metrics

## Response Structure

Parameter	Type	Description
stats	hash	
>>count	int	
>>98thPercentile	number	
>>min	number	
>>max	number	
>>5thPercentile	number	
>>95thPercentile	number	
>>median	number	
>>mean	number	
>>stddev	number	
>>sum	number	
data	array	
>>item	array	
>>time	number	
>>value	number	
label	string	

319: Response Example

```
{ "response": [
  {
    "stats": {
      "count": 988,
      "98thPercentile": 16589105.55958,
      "min": 3185442.975,
      "max": 17124754.257,
      "5thPercentile": 3901253.95445,
      "95thPercentile": 16013210.034,
      "median": 8816895.576,
      "mean": 8995846.31741194,
      "stddev": 3941169.83683573,
      "sum": 333296106.060112
    },
    "data": [
      [
        1414303200000,
        12923518.466
      ],
      [
        1414303500000,
        12625139.65
      ]
    ]
  }
],
```

(continues on next page)

(continued from previous page)

```

    "label": "MID Kbps"
  }
}]

```

## deliveryservices/{{ID}}/servers

**Caution:** It's often much easier to use *deliveryservices/{{xml\_id}}/servers* instead

### GET

Retrieves properties of Edge-Tier servers assigned to a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

Table 151: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the Delivery service for which servers will be displayed

### Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** An integral, unique identifier the CDN to which the server belongs

**cdnName** The name of the CDN to which the server belongs

**domainName** The domain name part of the FQDN of the server

**guid** Optionally represents an identifier used to uniquely identify the server

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS requests - 443 in most cases

**id** An integral, unique identifier for the server

**iloIpAddress** The IPv4 address of the lights-out-management port<sup>1</sup>

**iloIpGateway** The IPv4 gateway address of the lights-out-management port<sup>Page 455, 1</sup>

**iloIpNetmask** The IPv4 subnet mask of the lights-out-management port<sup>Page 455, 1</sup>

**iloPassword** The password of the of the lights-out-management user - displays as \*\*\*\*\* unless the requesting user has the ‘admin’ role)<sup>Page 455, 1</sup>

**iloUsername** The user name for lights-out-management<sup>Page 455, 1</sup>

**interfaces** An array of interface and IP address information

**max\_bandwidth** The maximum allowed bandwidth for this interface to be considered “healthy” by Traffic Monitor. This has no effect if *monitor* is not true. Values are in kb/s. The value *null* means “no limit”.

**monitor** A boolean indicating if Traffic Monitor should monitor this interface

**mtu** The MTU to configure for *interfaceName*

**See also:**

[The Wikipedia article on Maximum Transmission Unit](#)

**name** The network interface name used by the server.

**ipAddresses** An array of the IP address information for the interface

**address** The IPv4 or IPv6 address and subnet mask of the server - applicable for the interface name

**gateway** The IPv4 or IPv6 gateway address of the server - applicable for the interface name

**service\_address** A boolean determining if content will be routed to the IP address

**lastUpdated** The time and date at which this server was last updated, in *Traffic Ops’s Custom Date/Time Format*

**mgmtIpAddress** The IPv4 address of the server’s management port

**mgmtIpGateway** The IPv4 gateway of the server’s management port

**mgmtIpNetmask** The IPv4 subnet mask of the server’s management port

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status (will be empty if not offline)

---

<sup>1</sup> See [the Wikipedia article on Out-of-Band Management](#) for more information.

**physLocation** The name of the *Physical Location* at which the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* at which the server resides

**profile** The *Name* of the *Profile* assigned to this server

**profileDesc** A *Description* of the *Profile* assigned to this server

**profileId** The *ID* of the *Profile* assigned to this server

**rack** A string indicating “rack” location

**routerHostName** The human-readable name of the router

**routerPortName** The human-readable name of the router port

**status** The Status of the server

**See also:**

*Health Protocol*

**statusId** An integral, unique identifier for the status of the server

**See also:**

*Health Protocol*

**tcpPort** The default port on which the main application listens for incoming TCP connections - 80 in most cases

**type** The name of the type of this server

**typeId** An integral, unique identifier for the type of this server

**updPending** true if the server has updates pending, false otherwise

### 320: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: MaIva080Sjysr4bCkuXFEMf3o6mOqga1aM4IHN/
→tcP2aa1iXEmA5IrHB7DaqNX/2vGHLXvN+01FEAR/lRNqr1w==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:28:23 GMT
Content-Length: 891

{ "response": [
```

(continues on next page)

(continued from previous page)

```

{
  "cachegroup": "CDN_in_a_Box_Edge",
  "cachegroupId": 7,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "edge",
  "httpsPort": 443,
  "id": 10,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "lastUpdated": "2018-11-14 21:08:44+00",
  "mgmtIpAddress": "",
  "mgmtIpGateway": "",
  "mgmtIpNetmask": "",
  "offlineReason": "",
  "physLocation": "Apachecon North America 2018",
  "physLocationId": 1,
  "profile": "ATS_EDGE_TIER_CACHE",
  "profileDesc": "Edge Cache - Apache Traffic Server",
  "profileId": 9,
  "rack": "",
  "routerHostName": "",
  "routerPortName": "",
  "status": "REPORTED",
  "statusId": 3,
  "tcpPort": 80,
  "type": "EDGE",
  "typeId": 11,
  "updPending": false,
  "interfaces": [{
    "ipAddresses": [
      {
        "address": "172.16.239.100",
        "gateway": "172.16.239.1",
        "service_address": true
      },
      {
        "address": "fc01:9400:1000:8::100",
        "gateway": "fc01:9400:1000:8::1",
        "service_address": true
      }
    ]
  }]
}

```

(continues on next page)



(continued from previous page)

```

    ],
    "max_bandwidth": 0,
    "monitor": true,
    "mtu": 1500,
    "name": "eth0"
  }]
}
[]}

```

## deliveryservices/{{ID}}/servers/eligible

**Caution:** This endpoint may not work as advertised, and its use is therefore discouraged!

### GET

Retrieves properties of *Edge-tier cache servers* eligible for assignment to a particular *Delivery Service*. Eligibility is determined based on the following properties:

- The name of the server's *Type* must match one of the glob patterns `EDGE*`, `ORG*`
- The server and *Delivery Service* must belong to the same CDN
- If the *Delivery Service* has *Required Capabilities*, an *Edge-tier cache server* must have all of those defined capabilities

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

Table 152: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the Delivery service for which servers will be displayed

## Response Structure

- cachegroup** A string which is the *Name of the Cache Group* to which the server belongs
- cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs
- cdnId** An integral, unique identifier the CDN to which the server belongs
- cdnName** The name of the CDN to which the server belongs
- domainName** The domain name part of the FQDN of the server
- guid** Optionally represents an identifier used to uniquely identify the server
- hostName** The (short) hostname of the server
- httpsPort** The port on which the server listens for incoming HTTPS requests - 443 in most cases
- id** An integral, unique identifier for the server
- iloIpAddress** The IPv4 address of the lights-out-management port<sup>1</sup>
- iloIpGateway** The IPv4 gateway address of the lights-out-management port<sup>Page 455, 1</sup>
- iloIpNetmask** The IPv4 subnet mask of the lights-out-management port<sup>Page 455, 1</sup>
- iloPassword** The password of the of the lights-out-management user - displays as \*\*\*\*\* unless the requesting user has the ‘admin’ role)<sup>Page 455, 1</sup>
- iloUsername** The user name for lights-out-management<sup>Page 455, 1</sup>
- interfaces** An array of interface and IP address information
  - max\_bandwidth** The maximum allowed bandwidth for this interface to be considered “healthy” by Traffic Monitor. This has no effect if *monitor* is not true. Values are in kb/s. The value *null* means “no limit”.
  - monitor** A boolean indicating if Traffic Monitor should monitor this interface
  - mtu** The MTU to configure for *interfaceName*
  - See also:**
    - [The Wikipedia article on Maximum Transmission Unit](#)
  - name** The network interface name used by the server.
  - ipAddresses** An array of the IP address information for the interface

---

<sup>1</sup> See [the Wikipedia article on Out-of-Band Management](#) for more information.

**address** The IPv4 or IPv6 address and subnet mask of the server - applicable for the interface name

**gateway** The IPv4 or IPv6 gateway address of the server - applicable for the interface name

**service\_address** A boolean determining if content will be routed to the IP address

**lastUpdated** The time and date at which this server was last updated, in *Traffic Ops's Custom Date/Time Format*

**mgmtIpAddress** The IPv4 address of the server's management port

**mgmtIpGateway** The IPv4 gateway of the server's management port

**mgmtIpNetmask** The IPv4 subnet mask of the server's management port

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status (will be empty if not offline)

**physLocation** The name of the *Physical Location* at which the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* at which the server resides

**profile** The *Name* of the *Profile* assigned to this server

**profileDesc** A *Description* of the *Profile* assigned to this server

**profileId** The *ID* of the *Profile* assigned to this server

**rack** A string indicating "rack" location

**routerHostName** The human-readable name of the router

**routerPortName** The human-readable name of the router port

**status** The Status of the server

**See also:**

*Health Protocol*

**statusId** An integral, unique identifier for the status of the server

**See also:**

*Health Protocol*

**tcpPort** The default port on which the main application listens for incoming TCP connections - 80 in most cases

**type** The name of the *Type* of this server

**typeId** An integral, unique identifier for the *Type* of this server

**updPending** true if the server has updates pending, false otherwise

### 321: Response Example

```
{ "response": [
  {
    "cachegroup": "CDN_in_a_Box_Edge",
    "cachegroupId": 7,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "domainName": "infra.ciab.test",
    "guid": null,
    "hostName": "edge",
    "httpsPort": 443,
    "id": 10,
    "iloIpAddress": "",
    "iloIpGateway": "",
    "iloIpNetmask": "",
    "iloPassword": "",
    "iloUsername": "",
    "lastUpdated": "2018-10-30 16:01:12+00",
    "mgmtIpAddress": "",
    "mgmtIpGateway": "",
    "mgmtIpNetmask": "",
    "offlineReason": "",
    "physLocation": "Apachecon North America 2018",
    "physLocationId": 1,
    "profile": "ATS_EDGE_TIER_CACHE",
    "profileDesc": "Edge Cache - Apache Traffic Server",
    "profileId": 9,
    "rack": "",
    "routerHostName": "",
    "routerPortName": "",
    "status": "REPORTED",
    "statusId": 3,
    "tcpPort": 80,
    "type": "EDGE",
    "typeId": 11,
    "updPending": false,
    "interfaces": [{
      "ipAddresses": [
        {
          "address": "172.16.239.100",
          "gateway": "172.16.239.1",
          "service_address": true
        }
      ],
      {
        "address": "fc01:9400:1000:8::100",
        "gateway": "fc01:9400:1000:8::1",

```

(continues on next page)

(continued from previous page)

```

        "service_address": true
    }
],
"max_bandwidth": 0,
"monitor": true,
"mtu": 1500,
"name": "eth0"
}]
}
}]

```

`deliveryservices/{{ID}}/urlkeys`

GET

See also:

`deliveryservices/xmlId/{{xmlid}}/urlkeys`

Retrieves URL signing keys for a *Delivery Service*.

**Caution:** This method will return the *Delivery Service*'s **PRIVATE** URL signing keys! Be wary of using this endpoint and **NEVER** share the output with anyone who would be unable to see it on their own.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 153: Request Path Parameters

Name	Description
id	Filter for the <i>Delivery Service</i> identified by this integral, unique identifier

322: Request Example

```

GET /api/3.0/deliveryservices/1/urlkeys HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*

```

(continues on next page)

(continued from previous page)

```
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**key<N>** The private URL signing key for this *Delivery Service* as a base-64-encoded string, where <N> is the “generation” of the key e.g. the first key will always be named "key0". Up to 16 concurrent generations are retained at any time (<N> is always on the interval [0,15])

### 323: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 16:34:56
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: cTc5OPE3hM+CiyQPCy36zD2tsQcfkvIqQ7/
    ↳D82WGMWHm+ACW3YbcKhgPnSQU6+Tuj4jya52Kx9+nw5+OonFvPQ==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 15:34:56 GMT
Content-Length: 533

{
  "response": {
    "key0": "...",
    "key1": "...",
    "key2": "...",
    "key3": "...",
    "key4": "...",
    "key5": "...",
    "key6": "...",
    "key7": "...",
    "key8": "...",
    "key9": "...",
    "key10": "...",
    "key11": "...",
    "key12": "...",
    "key13": "...",
    "key14": "...",
```

(continues on next page)

(continued from previous page)

```

    "key15": "...",
  }
}

```

## deliveryservices\_regexes

### GET

Retrieves routing regular expressions for all *Delivery Services*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

### Request Structure

No parameters available

### Response Structure

**dsName** The name of the *Delivery Service* represented by this object

**regexes** An array of objects that represent various routing regular expressions used by dsName

**pattern** The actual regular expression - \s are escaped

**setNumber** The order in which the regular expression is evaluated against requests

**type** The type of regular expression - determines that against which it will be evaluated

#### 324: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json

```

(continues on next page)

<sup>1</sup> If tenancy is used, then users (regardless of role) will only be able to see the routing regular expressions used by *Delivery Services* their tenant has permissions to see.

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +2MI+Q/NJqTizlMR/MhPAL+yu6/z/
→Yqvo5fD08F593RM0mK6dX/A14wARbEG+HQaJNgSCRPSiLVATusrmnnCMA==
X-Server-Name: traffic_ops_golang/
Date: Tue, 27 Nov 2018 19:22:59 GMT
Content-Length: 110

{ "response": [
  {
    "regexes": [
      {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.\\.*"
      }
    ],
    "dsName": "demo1"
  }
]}
```

## deliveryservices/request

---

**Note:** This route does NOT do the same thing as *POST deliveryservice\_requests*.

---

Deprecated since version ATCv6: This endpoint does not appear in Traffic Ops API version 4.0 - released with Apache Traffic Control version 6.0 - or later.

## POST

Submits an emailed requesting that a *Delivery Service* be created.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** undefined



## Request Structure

**details** An object describing the actual parameters for the Delivery Service request

**customer** Name of the customer associated with the *Delivery Service* - must only contain alphanumeric characters and the characters @, !, #, \$, %, ^, &, \*, (, ), [, ], ., , and -

Changed in version ATCv6: Prior to ATC version 6, this field had no restrictions.

**deepCachingType** An optional string describing when to do Deep Caching for this *Delivery Service* - one of:

**NEVER** Never use deep caching (default)

**ALWAYS** Always use deep caching

**deliveryProtocol** The protocol used to retrieve content from the CDN - one of:

- http
- https
- http/https

**hasNegativeCachingCustomization** true if any customization is required for negative caching, false otherwise

**hasOriginACLWhitelist** true if access to the origin is restricted using an Access Control List (ACL or “whitelist”) of IP addresses

**hasOriginDynamicRemap** If true, this *Delivery Service* can dynamically map to multiple origin URLs

**hasSignedURLs** If true, this *Delivery Service*’s URLs are signed

**headerRewriteEdge** An optional string containing a header rewrite rule to be used at the Edge tier

**headerRewriteMid** An optional string containing a header rewrite rule to be used at the Mid tier

**headerRewriteRedirectRouter** An optional string containing a header re-write rule to be used by the Traffic Router

**maxLibrarySizeEstimate** A special string that describes the estimated size of the sum total of content available through this *Delivery Service*

**negativeCachingCustomizationNote** A note remarking on the use, customization, or complications associated with negative caching for this *Delivery Service*

- notes** An optional string containing additional instructions or notes regarding the Request
- originHeaders** An optional, comma-separated string of header values that must be passed to requests to the *Delivery Service*'s origin
- originTestFile** A URL path to a test file available on the *Delivery Service*'s origin server
- originURL** The URL of the *Delivery Service*'s origin server
- otherOriginSecurity** An optional string describing any and all other origin security measures that need to be considered for access to the *Delivery Service*'s origin
- overflowService** An optional string containing the IP address or URL of an overflow point (used if rate limits are met or exceeded)
- peakBPSEstimate** A special string describing the estimated peak data transfer rate of the *Delivery Service* in Bytes Per Second (BPS)
- peakTPSEstimate** A special string describing the estimated peak transaction rate of the *Delivery Service* in Transactions Per Second (TPS)
- queryStringHandling** A special string describing how the *Delivery Service* should treat URLs containing query parameters
- rangeRequestHandling** A special string describing how the *Delivery Service* should handle range requests
- rateLimitingGBPS** An optional field which, if defined, should contain the maximum allowed data transfer rate for the *Delivery Service* in GigaBytes Per Second (GBPS)
- rateLimitingTPS** An optional field which, if defined, should contain the maximum allowed transaction rate for the *Delivery Service* in Transactions Per Second (TPS)
- routingName** An optional field which, if defined, should contain the routing name for the *Delivery Service*, e.g. `SomeRoutingName.DeliveryService_xml_id.CDNName.com`
- routingType** The *Delivery Service*'s routing type, should be one of:
- HTTP** The Traffic Router re-directs clients to *cache servers* using the HTTP 302 REDIRECT response code

**DNS** The Traffic Router responds to requests for name resolution of the *Delivery Service*'s routing name with IP addresses of *cache servers*

**STEERING** This *Delivery Service* routes clients to other *Delivery Services* - which will in turn (generally) route them to clients

**ANY\_MAP** Some kind of undocumented black magic is used to get clients to... content, probably?

**serviceAliases** An optional array of aliases for this *Delivery Service*

**serviceDesc** A description of the *Delivery Service*

**emailTo** The email to which the Delivery Service request will be sent

### 325: Request Example

```
{ "emailTo": "foo@bar.com",
  "details": {
    "customer": "XYZ Corporation",
    "contentType": "static",
    "deepCachingType": "NEVER",
    "deliveryProtocol": "http",
    "routingType": "http",
    "routingName": "demo1",
    "serviceDesc": "service description goes here",
    "peakBPSEstimate": "less-than-5-Gbps",
    "peakTPSEstimate": "less-than-1000-TPS",
    "maxLibrarySizeEstimate": "less-than-200-GB",
    "originURL": "http://myorigin.com",
    "hasOriginDynamicRemap": false,
    "originTestFile": "http://origin.infra.ciab.test",
    "hasOriginACLWhitelist": false,
    "originHeaders": "",
    "otherOriginSecurity": "",
    "queryStringHandling": "ignore-in-cache-key-and-pass-up",
    "rangeRequestHandling": "range-requests-not-used",
    "hasSignedURLs": false,
    "hasNegativeCachingCustomization": false,
    "negativeCachingCustomizationNote": "",
    "serviceAliases": [],
    "rateLimitingGBPS": 50,
    "rateLimitingTPS": 5000,
    "overflowService": null,
    "headerRewriteEdge": "",
    "headerRewriteMid": "",
    "headerRewriteRedirectRouter": "",
```

(continues on next page)

(continued from previous page)

```
"notes": ""  
}}
```

## Response Structure

326: Response Example

```
{ "alerts": [{  
  "level": "success",  
  "text": "Delivery Service request sent to foo@bar.com."  
}]}
```

### deliveryservices\_required\_capabilities

#### GET

Gets all associations of *Server Capability* to *Delivery Services*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 154: Request Query Parameters

Name	Re-quired	Description
deliv-ery-Servi-ceID	no	Filter <i>Server Capability</i> associations by <i>Delivery Service</i> integral, unique identifier
xmlID	no	Filter <i>Server Capability</i> associations by <i>Delivery Service xml_id</i>
re-quired-Capa-bility	no	Filter <i>Server Capability</i> associations by <i>Server Capability</i> name
or-derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

### 327: Request Example

```
GET /api/3.0/deliveryservices_required_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**deliveryServiceID** The associated *Delivery Service*’s integral, unique identifier

**xmlID** The associated *Delivery Service*’s *xml\_id*

**lastUpdated** The date and time at which this association between the *Delivery Service* and the *Server Capability* was last updated, in *Traffic Ops’s Custom Date/Time Format*

**requiredCapability** The *Server Capability*’s name

### 328: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
    ↳jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
    ↳DbBwhMAOJoeqDIImoDlrLnrVjQG04AooA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 396

{
  "response": [
    {
      "deliveryServiceID": 1,
      "lastUpdated": "2019-10-07 22:05:31+00",
      "requiredCapability": "ram",
      "xmlId": "example_ds-1"
    },
    {
      "deliveryServiceID": 2,
      "lastUpdated": "2019-10-07 22:05:31+00",
      "requiredCapability": "disk",
      "xmlId": "example_ds-2"
    }
  ]
}

```

## POST

Associates a *Server Capability* with a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

---

**Note:** A *Server Capability* can only be made required on a *Delivery Service* if its associated

Servers already have that *Server Capability* assigned.

## Request Structure

**deliveryServiceID** The integral, unique identifier of the *Delivery Service* to be associated

**requiredCapability** The name of the *Server Capability* to be associated

### 329: Request Example

```
POST /api/3.0/deliveryservices_required_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "deliveryServiceID": 1,
  "requiredCapability": "disk"
}
```

## Response Structure

**deliveryServiceID** The newly associated *Delivery Service*'s integral, unique identifier

**lastUpdated** The date and time at which this association between the *Delivery Service* and the *Server Capability* was last updated, in *Traffic Ops's Custom Date/Time Format*

**requiredCapability** The newly associated *Server Capability*'s name

### 330: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512:↵
↵eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
↵Cfh2X9i3G+MQ36eH95gukJq0BOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 287

{
  "alerts": [
    {
      "level": "success",
      "text": "deliveryservice.RequiredCapability was created."
    }
  ],
  "response": {
    "deliveryServiceID": 1,
    "lastUpdated": "2019-10-07 22:15:11+00",
    "requiredCapability": "disk"
  }
}
```

## DELETE

Dissociate a *Server Capability* from a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

**deliveryServiceID** The integral, unique identifier of the *Delivery Service* from which a *Server Capability* will be dissociated

**requiredCapability** The name of the *Server Capability* to dissociate

331: Request Example

```
POST /api/3.0/deliveryservices_required_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
```

(continues on next page)



(continued from previous page)

```
Content-Type: application/json

{
  "deliveryServiceID": 1,
  "requiredCapability": "disk"
}
```

## Response Structure

### 332: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
→Cfh2X9i3G+MQ36eH95gukJqOBOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 127

{
  "alerts": [
    {
      "level": "success",
      "text": "deliveryservice.RequiredCapability was deleted."
    }
  ]
}
```

## deliveryservices/sslkeys/add

### See also:

In most cases it is preferable to allow Traffic Ops to generate the keys via *deliveryservices/sslkeys/generate*, rather than uploading them manually using this endpoint.

### POST

Allows user to upload an SSL certificate, csr, and private key for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object (string)

### Request Structure

**cdn** The name of the CDN to which the *Delivery Service* belongs

**certificate** An object that contains the actual components of the SSL key

**crt** The certificate for the *Delivery Service* identified by key

**csr** The csr file for the *Delivery Service* identified by key

**key** The private key for the *Delivery Service* identified by key

**key** The *xml\_id* of the *Delivery Service* to which these keys will be assigned

**version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

### 333: Request Example

```
POST /api/3.0/deliveryservices/sslkeys/add HTTP/1.1
Host: trafficops.infra.ciab.test
Content-Type: application/json

{
  "key": "ds-01",
  "version": "1",
  "certificate": {
    "key": "some_key",
    "csr": "some_csr",
    "crt": "some_crt"
  }
}
```

## Response Structure

### 334: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "response": "Successfully added ssl keys for ds-01"
}
```

## deliveryservices/sslkeys/generate

### POST

Generates an SSL certificate, csr, and private key for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object (string)

## Request Structure

**city** An optional field which, if present, will represent the resident city of the generated SSL certificate

**country** An optional field which, if present, will represent the resident country of the generated SSL certificate

**hostname** The desired hostname of the *Delivery Service*

---

**Note:** In most cases, this must be the same as the *Delivery Service* URL’

---

**key** The *xml\_id* of the *Delivery Service* for which keys will be generated

**organization** An optional field which, if present, will represent the organization for which the SSL certificate was generated

**state** An optional field which, if present, will represent the resident state or province of the generated SSL certificate

**businessUnit** An optional field which, if present, will represent the business unit for which the SSL certificate was generated

**version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

335: Request Example

```
POST /api/3.0/deliveryservices/sslkeys/generate HTTP/1.1
Content-Type: application/json

{
  "key": "ds-01",
  "businessUnit": "CDN Engineering",
  "version": "3",
  "hostname": "tr.ds-01.ott.kabletown.com",
  "country": "US",
  "organization": "Kabletown",
  "city": "Denver",
  "state": "Colorado"
}
```

## Response Structure

336: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": "Successfully created ssl keys for ds-01" }
```

**deliveryservices/sslkeys/generate/letsencrypt**

**POST**

Generates an SSL certificate and private key using Let's Encrypt for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object (string)

## Request Structure

**key** The *xml\_id* of the *Delivery Service* for which keys will be generated<sup>1</sup>

**deliveryservice** The *xml\_id* of the *Delivery Service* for which keys will be generated<sup>Page 655, 1</sup>

**version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

**hostname** The desired hostname of the *Delivery Service*

---

**Note:** In most cases, this must be the same as the *Example URLs*.

---

**cdn** The name of the CDN of the *Delivery Service* for which the certs will be generated

### 337: Request Example

```
POST /api/3.0/deliveryservices/sslkeys/generate/letsencrypt HTTP/1.1
Content-Type: application/json

{
  "key": "ds-01",
  "deliveryservice": "ds-01",
  "version": "3",
  "hostname": "tr.ds-01.ott.kabletown.com",
  "cdn": "test-cdn"
}
```

## Response Structure

---

<sup>1</sup> Either the key or the deliveryservice field must be provided. If both are provided, then they must match.

### 338: Response Example

```
{ "alerts": [{
  "level": "success",
  "text": "Beginning async ACME call for demo1 using Lets Encrypt.
→ This may take a few minutes. Status updates can be found here: /api/
→ 4.0/async_status/1"
}]}
```

`deliveryservices/{xml_id}/servers`

**POST**

Assigns *cache servers* to a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Response Type** Object

### Request Structure

Table 155: Request Path Parameters

Name	Description
xml_id	The ‘xml_id’ of the <i>Delivery Service</i> whose server assignments are being edited

**serverNames** An array of hostname of *cache servers* to assign to this *Delivery Service*

### 339: Request Example

```
POST /api/3.0/deliveryservices/test/servers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 24
Content-Type: application/json

{ "serverNames": [ "edge" ] }
```

<sup>1</sup> Users can only assign servers to *Delivery Services* that are visible to their *Tenant*.

## Response Structure

**xml\_id** The *xml\_id* of the *Delivery Service* to which the servers in `serverNames` have been assigned

**serverNames** An array of hostnames of *cache servers* assigned to *Delivery Service* identified by `xml_id`

### 340: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→zTpLrWiLM4xRsm8mlBQFB5KzT478AjloSyXHgtyWhebCv1YIwWltnmkjr0HFgc3GMGZODt+fyzk0Yy5Zl/
→yBtJw==
X-Server-Name: traffic_ops_golang/
Date: Tue, 20 Nov 2018 15:21:50 GMT
Content-Length: 52

{ "response": {
  "serverNames": [
    "edge"
  ],
  "xmlId": "test"
}}
```

`deliveryservices/{xml_id}/urisignkeys`

## DELETE

Deletes URISigning objects for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** admin<sup>1</sup>

**Response Type** undefined

<sup>1</sup> URI Signing Keys can only be created, viewed, deleted, or modified on *Delivery Services* that either match the requesting user's *Tenant* or are descendants thereof.

## Request Structure

Table 156: Request Path Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Response Structure

TBD

### GET

Retrieves one or more URISigning objects for a delivery service.

**Auth. Required** Yes

**Roles Required** admin [Page 655, 1](#)

**Response Type** undefined

## Request Structure

Table 157: Request Route Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Response Structure

**Issuer** a string describing the issuer of the URI signing object. Multiple URISigning objects may be returned in a response, see example

**renewal\_kid** a string naming the jwt key used for renewals

**keys** json array of jwt symmetric keys

**alg** this parameter repeats for each jwt key in the array and specifies the jwa encryption algorithm to use with this key, [RFC 7518](#)

**kid** this parameter repeats for each jwt key in the array and specifies the unique id for the key as defined in [RFC 7516](#)

**key** this parameter repeats for each jwt key in the array and specifies the key type as defined in [RFC 7516](#)

**k** this parameter repeats for each jwt key in the array and specifies the base64 encoded symmetric key see [RFC 7516](#)



### 341: Response Example

```
{ "Kabletown URI Authority": {
  "renewal_kid": "Second Key",
  "keys": [
    {
      "alg": "HS256",
      "kid": "First Key",
      "kty": "oct",
      "k": "Kh_RkUMj-fzbD37qBnDf_3e_RvQ3RP9PaSmVEpE24AM"
    },
    {
      "alg": "HS256",
      "kid": "Second Key",
      "kty": "oct",
      "k": "fZBpDBNbk2GqhwoB_DGBAsBxqQZVix04rIoLJ7p_RlE"
    }
  ]
}}
```

## POST

Assigns URISigning objects to a delivery service.

**Auth. Required** Yes

**Roles Required** admin [Page 655, 1](#)

**Response Type** undefined

## Request Structure

Table 158: Request Path Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Request Structure

**Issuer** a string describing the issuer of the URI signing object. Multiple URISigning objects may be returned in a response, see example

**renewal\_kid** a string naming the jwt key used for renewals

**keys** json array of jwt symmetric keys

- alg** this parameter repeats for each jwt key in the array and specifies the jwa encryption algorithm to use with this key, [RFC 7518](#)
- kid** this parameter repeats for each jwt key in the array and specifies the unique id for the key as defined in [RFC 7516](#)
- kt** this parameter repeats for each jwt key in the array and specifies the key type as defined in [RFC 7516](#)
- k** this parameter repeats for each jwt key in the array and specifies the base64 encoded symmetric key see [RFC 7516](#)

#### 342: Request Example

```
{ "Kabletown URI Authority": {
  "renewal_kid": "Second Key",
  "keys": [
    {
      "alg": "HS256",
      "kid": "First Key",
      "kty": "oct",
      "k": "Kh_RkUMj-fzbD37qBnDf_3e_RvQ3RP9PaSmVEpE24AM"
    },
    {
      "alg": "HS256",
      "kid": "Second Key",
      "kty": "oct",
      "k": "fZBpDBNbk2GqhwoB_DGBAsBxqQZVix04rIoLJ7p_RlE"
    }
  ]
}}
```

## PUT

updates URISigning objects on a delivery service.

**Auth. Required** Yes

**Roles Required** admin Page 655, 1

**Response Type** undefined

## Request Structure

Table 159: Request Path Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Request Structure

**Issuer** a string describing the issuer of the URI signing object. Multiple URISigning objects may be returned in a response, see example

**renewal\_kid** a string naming the jwt key used for renewals

**keys** json array of jwt symmetric keys

**alg** this parameter repeats for each jwt key in the array and specifies the jwa encryption algorithm to use with this key, [RFC 7518](#)

**kid** this parameter repeats for each jwt key in the array and specifies the unique id for the key as defined in [RFC 7516](#)

**kty** this parameter repeats for each jwt key in the array and specifies the key type as defined in [RFC 7516](#)

**k** this parameter repeats for each jwt key in the array and specifies the base64 encoded symmetric key see [RFC 7516](#)

### 343: Request Example

```
{ "Kabletown URI Authority": {
  "renewal_kid": "Second Key",
  "keys": [
    {
      "alg": "HS256",
      "kid": "First Key",
      "kty": "oct",
      "k": "Kh_RkUMj-fzbD37qBnDf_3e_RvQ3RP9PaSmVEpE24AM"
    },
    {
      "alg": "HS256",
      "kid": "Second Key",
      "kty": "oct",
      "k": "fZBpDBNbk2GqhwoB_DGBAsBxqQZVix04rIoLJ7p_RlE"
    }
  ]
}}
```

`deliveryservices/xmlId/{XMLID}/sslkeys`

**GET**

Retrieves SSL keys for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 160: Request Path Parameters

Name	Description
XMLID	The ‘xml_id’ of the desired <i>Delivery Service</i>

Table 161: Request Query Parameters

Name	Re-quired	Description
ver-sion	no	The version number of the SSL keys to retrieve
de-code	no	If <code>true</code> , the returned keys will be decoded - if <code>false</code> , they will not be decoded

**Caution:** There’s almost certainly no good reason to request the private key! Even when “base 64-encoded” do not let **ANYONE** see this who would be unable to request it themselves!

## Response Structure

**businessUnit** An optional field which, if present, contains the business unit entered by the user when generating the SSL certificate<sup>1</sup>

**certificate** An object containing the actual generated key, certificate, and signature of the SSL keys

**crt** Base 64-encoded (or not if the decode query parameter was given and `true`) certificate for the *Delivery Service* identified by `deliveryservice`

---

<sup>1</sup> These optional fields will be present in the response if and only if they were specified during key generation; they are optional during key generation and thus cannot be guaranteed to exist or not exist.

**csr** Base 64-encoded (or not if the decode query parameter was given and true) csr file for the *Delivery Service* identified by deliveryservice

**key** Base 64-encoded (or not if the decode query parameter was given and true) private key for the *Delivery Service* identified by deliveryservice

**Caution:** There's almost certainly no good reason to request the private key! Even when "base 64-encoded" do not let **ANYONE** see this who would be unable to request it themselves!

**cdn** The CDN of the *Delivery Service* for which the certs were generated

**city** An optional field which, if present, contains the city entered by the user when generating the SSL certificate<sup>Page 676, 1</sup>

**country** An optional field which, if present, contains the country entered by the user when generating the SSL certificate<sup>Page 676, 1</sup>

**deliveryservice** The 'xml\_id' of the *Delivery Service* for which the certificate was generated

**hostname** The hostname generated by Traffic Ops that is used as the common name when generating the certificate - this will be a FQDN for DNS *Delivery Services* and a wildcard URL for HTTP *Delivery Services*

**organization** An optional field which, if present, contains the organization entered by the user when generating certificate<sup>Page 676, 1</sup>

**state** An optional field which, if present, contains the state entered by the user when generating certificate<sup>Page 676, 1</sup>

**version** An integer that defines the "version" of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

**expiration** The expiration date of the certificate for the *Delivery Service* in **RFC 3339** format

### 344: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": {
  "certificate": {
    "crt": "crt",
    "key": "key",
    "csr": "csr"
  },

```

(continues on next page)

(continued from previous page)

```
"deliveryservice": "my-ds",
"cdn": "qa",
"businessUnit": "CDN_Eng",
"city": "Denver",
"organization": "KableTown",
"hostname": "foober.com",
"country": "US",
"state": "Colorado",
"version": "1",
"expiration": "2020-08-18T13:53:06Z"
}}
```

## DELETE

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object (string)

## Request Structure

Table 162: Request Path Parameters

Name	Required	Description
xmlId	yes	The <i>xml_id</i> of the desired <i>Delivery Service</i>

Table 163: Request Query Parameters

Name	Required	Description
version	no	The version number of the SSL keys that shall be retrieved

## Response Structure

### 345: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 18 Mar 2020 17:36:10 GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Pj+zCo0Xg19nGNxcSkjib2iDjG062Y3RcEEV+0YnwbGIsLcpa0BKZleY/qJOKT5DkSoX2qQkckUxUqdDxjVorQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 18 Mar 2020 16:36:10 GMT
Content-Length: 79

{
  "response": "Successfully deleted ssl keys for demo1"
}
```

`deliveryservices/xmlId/{XMLID}/sslkeys/renew`

POST

Uses ACME protocol to renew SSL keys for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 164: Request Path Parameters

Name	Description
XMLID	The ‘xml_id’ of the desired <i>Delivery Service</i>

## Request Structure

No parameters available

## Response Structure

346: Response Example

```
{ "alerts": [{
  "level": "success",
  "text": "Certificate for test-xml-id successfully renewed."
}]}
```

`deliveryservices/xmlId/{xmlid}/urlkeys`

GET

See also:

*deliveryservices/{ID}/urlkeys*

Retrieves URL signing keys for a *Delivery Service*.

**Caution:** This method will return the *Delivery Service*'s **PRIVATE** URL signing keys! Be wary of using this endpoint and **NEVER** share the output with anyone who would be unable to see it on their own.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 165: Request Path Parameters

Name	Description
xmlid	The 'xml_id' of the desired <i>Delivery Service</i>

## Response Structure

**key<N>** The private URL signing key for this *Delivery Service* as a base-64-encoded string, where <N> is the “generation” of the key e.g. the first key will always be named "key0". Up to 16 concurrent generations are retained at any time (<N> is always on the interval [0,15])



### 347: Response Example

```
{ "response": {
  "key9": "ZvVQNYpPVQWQV8tjQnUl6osm4y7xK4zD",
  "key6": "JhGdpw5X9o8TqHfgezCm0bqb9SQPASWL",
  "key8": "ySXdp1T8IeDEE10CMftzZb9EIw_20wwq",
  "key0": "D4AYzJ1AE2nYisA9MxMtY03TPDCHji9C",
  "key3": "W90YHlGc_kYlYw5_I0LrkvV9J0zSIneI",
  "key12": "ZbtMb3mrKqfS8hnx9_xWBIP_OPWlUpzc",
  "key2": "0qgEoD07sUsugIQemZbwmMt0tNCwB1sf",
  "key4": "aFJ2Gb7atmxVB8uv7T9S60aDml3ycpGf",
  "key1": "wnWNR1mCz104C7EFPtcqHd0xUMQyNFhA",
  "key11": "k6HMzlBH1x6htKkypRFfWQhAndQqe50e",
  "key10": "zYONfdD7fGYKj4kLvIj4U0918csuZ00d",
  "key15": "3360cGaIip_layZMc_0hI2teJbazxTQh",
  "key5": "SIwv3G0hWN7EE9wSwPFj18qE4M07sFxN",
  "key13": "SqQKBR6LqEOzp8AewZUCVtBcW_8YFc1g",
  "key14": "DtXsu8nsw04YhT0kNoKBhu2G3P9WRpQJ",
  "key7": "cmKoIIxXGAxUMdCsWvnGLOIMGmNiuT5I"
}}
```

deliveryservices/xmlId/{xml\_id}/urlkeys/copyFromXmlId/  
{{copyFrom\_xml\_id}}

### POST

Allows a user to copy URL signing keys from a specified *Delivery Service* to another *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object (string)

### Request Structure

Table 166: Request Path Parameters

Name	Description
xml_id	The <i>xml_id</i> of the <i>Delivery Service</i> to which keys will be copied
copyFrom_xml_id	The <i>xml_id</i> of the <i>Delivery Service</i> from which keys will be copied

## Response Structure

348: Response Example

```
{
  "response": "Successfully copied and stored keys"
}
```

`deliveryservices/xmlId/{xml_id}/urlkeys/generate`

**POST**

Generates URL signing keys for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object (string)

## Request Structure

Table 167: Request Path Parameters

Name	Description
xml_id	The <i>xml_id</i> of the desired <i>Delivery Service</i>

## Response Structure

349: Response Example

```
{
  "response": "Successfully generated and stored keys"
}
```

`deliveryservicesserver`

**GET**

Retrieve information about the assignment of servers to *Delivery Services*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

---

<sup>1</sup> While no roles are required, this endpoint *does* respect tenancy permissions (pending [GitHub Issue #2978](#)).

**Response Type** Array

## Request Structure

Table 168: Request Query Parameters

Name	Required	Default	Description
cdn	no	None	Limit the results to delivery service servers for the given CDN name
page	no	0	The page number for use in pagination - 0 means “no pagination”
limit	no	20	Limits the results to a maximum of this number - if pagination is used, this defines the number of results per page
orderby	no	“deliveryService”	Choose the ordering of the results - the value must either be the name of one of the fields of the objects in the response array or be empty to skip ordering altogether

### 350: Request Example

```
GET /api/3.0/deliveryserviceserver?page=1&limit=2&orderBy=lastUpdated.
→ HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

Unlike most API endpoints, this will return a JSON response body containing both a “response” object as well as other, top-level fields (besides the optional “alerts” field). For this reason, this section contains a “response” key, which normally is implicit.

### See also:

#### *Response Structure*

**limit** The maximum size of the response array, also indicative of the number of results per page using the pagination requested by the query parameters (if any) - this should be the same as the `limit` query parameter (if given)

**orderBy** A string that names the field by which the elements of the response array are ordered - should be the same as the `orderBy` request query parameter (if given)

**response** An array of objects, each of which represents a server’s *Delivery Service* assignment

**deliveryService** The integral, unique identifier of the *Delivery Service* to which the server identified by `server` is assigned

**lastUpdated** The date and time at which the server's assignment to a *Delivery Service* was last updated

**server** The integral, unique identifier of a server which is assigned to the *Delivery Service* identified by `deliveryService`

**size** The page number - if pagination was requested in the query parameters, else 0 to indicate no pagination - of the results represented by the response array. This is named "size" for legacy reasons

### 351: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: J7sK8PohQWyTpTrMjJrWdlJwPj+Zyep/
    ↳xutM25uVosL6cHgi30nXa6VMYOC5Y3vd9r5KLES8rTgR+qUQcZcJ/A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 01 Nov 2018 14:27:45 GMT
Content-Length: 129

{ "orderBy": "lastUpdated",
  "response": [
    {
      "server": 8,
      "deliveryService": 1,
      "lastUpdated": "2018-11-01 14:10:38+00"
    }
  ],
  "size": 1,
  "limit": 2
}
```

## POST

Assign a set of one or more servers to a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>2</sup>

**Response Type** Object

## Request Structure

**dsId** The integral, unique identifier of the *Delivery Service* to which the servers identified in the `servers` array will be assigned

**replace** If true, any existing assignments for a server identified in the `servers` array will be overwritten by this request

**servers** An array of integral, unique identifiers for servers which are to be assigned to the *Delivery Service* identified by `deliveryService`

### 352: Request Example

```
POST /api/3.0/deliveryserviceserver HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 46
Content-Type: application/x-www-form-urlencoded

dsId=1&replace=true&servers=12
```

## Response Structure

**dsId** The integral, unique identifier of the *Delivery Service* to which the servers identified by the elements of the `servers` array have been assigned

**replace** If true, any existing assignments for a server identified in the `servers` array have been overwritten by this request

**servers** An array of integral, unique identifiers for servers which have been assigned to the *Delivery Service* identified by `deliveryService`

<sup>2</sup> Users with the “admin” or “operations” roles will be able to modify ALL server-to-Delivery-Service assignments, whereas all other users can only assign servers to the *Delivery Services* their Tenant has permissions to edit.

### 353: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳D+HhGhozxaxvka9vZIStoa0ZUpX23nz7zZnMbpFHNRO3MawyEaSb3GVUHQyCv6sDgwhpZZjRggDmctGCw
X-Server-Name: traffic_ops_golang/
Date: Thu, 01 Nov 2018 14:12:49 GMT
Content-Length: 123

{ "alerts": [
  {
    "text": "server assignments complete",
    "level": "success"
  }
],
"response": {
  "dsId": 1,
  "replace": false,
  "servers": [ 12 ]
}}
```

`deliveryserviceserver/{DSID}/{serverID}`

#### DELETE

Removes a *cache server* from a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Response Type** undefined

<sup>1</sup> Users with the “admin” or “operations” roles will be able to delete *any Delivery Service*, whereas other users will only be able to delete *Delivery Services* that their tenant has permissions to delete.

## Request Structure

Table 169: Request Path Parameters

Name	Required	Description
dsId	yes	An integral, unique identifier for a <i>Delivery Service</i>
serverID	yes	An integral, unique identifier for a server

**Note:** The server identified by serverID must be a *cache server*, or the assignment will fail.

## Response Structure

354: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: rGD2sOMHYF0sga1zuTytyLHCUkkc3ZwQRKvZ/
    ↳HuPz0bOP4WztKTOVXB4uhs3iJqBg9zRB2TucMxONHN+3/yShQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 01 Nov 2018 14:24:34 GMT
Content-Length: 80

{ "alerts": [
  {
    "text": "Server unlinked from delivery service.",
    "level": "success"
  }
]}
```

## divisions

### GET

Returns a JSON representation of all configured *Divisions*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 170: Request Query Parameters

Name	Description
id	Filter for <i>Divisions</i> having this integral, unique identifier
name	Filter for <i>Divisions</i> with this name
orderby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops’s Custom Date/Time Format*

**name** The Division name

### 355: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)



(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→SLKi9RHa67sGoSz62IDcQsk7KZjTXKfonqMoCUFPXGcNUdhBssvUjc1G7KkWK8X1Ny16geMx2BN8Hm/
→3dQ75GA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 19:44:03 GMT
Content-Length: 139

{ "response": [
  {
    "id": 1,
    "lastUpdated": "2018-11-29 18:38:28+00",
    "name": "Quebec"
  },
  {
    "id": 2,
    "lastUpdated": "2018-11-29 18:38:28+00",
    "name": "USA"
  }
]}
```

## POST

Creates a new Division.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**name** The name of the new Division

### 356: Request Example

```
POST /api/3.0/divisions HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 16
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
{"name": "test"}
```

## Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops's Custom Date/Time Format*

**name** The Division name

### 357: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +pJm4c30+JTaSXNt+LP+u240Ba/
    ↳SsvSSDOQ4rDc6hcyZ0FIL+iY/WWrMHhpLulRGKGY88bM4YPCMaxGn3FZ9yQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 19:52:06 GMT
Content-Length: 136

{ "alerts": [
  {
    "text": "division was created.",
    "level": "success"
  }
],
"response": {
  "id": 3,
  "lastUpdated": "2018-11-29 19:52:06+00",
  "name": "test"
}}
```

`divisions/{ID}`

**PUT**

Updates a specific Division

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

## Request Structure

Table 171: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the requested Division

**name** The new name of the Division

358: Request Example

```
PUT /api/3.0/divisions/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 17
Content-Type: application/json

{"name": "quest"}
```

## Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops’s Custom Date/Time Format*

**name** The Division name

359: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: yBd8MzBR/Qbc/
→xTs44WEIFRTrqeMKZwUe2ufpm6JH6frh1UjFmYRs3/
→B7E5FTruFWRTuvEIlx5EpDmp3f9LjzA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 20:10:36 GMT
Content-Length: 137

{ "alerts": [
  {
    "text": "division was updated.",
    "level": "success"
  }
],
"response": {
  "id": 3,
  "lastUpdated": "2018-11-29 20:10:36+00",
  "name": "quest"
}}
```

## federation\_resolvers

### GET

Retrieves *Federation* Resolvers.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 172: Request Query Parameters

Name	Re-quired	Description
id	no	Return only the Federation Resolver identified by this integral, unique identifier
ipAddress	no	Return only the Federation Resolver(s) that has/have this IP Address
type	no	Return only the Federation Resolvers of this <i>Type</i>
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 360: Request Example

```
GET /api/3.0/federation_resolvers?type=RESOLVE6 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.63.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**id** The integral, unique identifier of the resolver

**ipAddress** The IP address or CIDR-notation subnet of the resolver - may be IPv4 or IPv6

**lastUpdated** The date and time at which this resolver was last updated, in *Traffic Ops’s Custom Date/Time Format*

**type** The *Type* of the resolver

### 361: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→4TLkULAOAuap47H+hpwyf2lHjDbHbSNQHLmj7BCTHtps2CQxCuq7mwctbwqmPdmAjLOUXAIRsHmvSuAp4
X-Server-Name: traffic_ops_golang/
Date: Wed, 06 Nov 2019 00:03:56 GMT
Content-Length: 101

{ "response": [
  {
    "id": 1,
    "ipAddress": "::1/1",
    "lastUpdated": "2019-11-06 00:00:40+00",
    "type": "RESOLVE6"
  }
]}
```

## POST

Creates a new federation resolver.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

**ipAddress** The IP address of the resolver - may be IPv4 or IPv6

**typeId** The integral, unique identifier of the *Type* of resolver being created

**Caution:** This field should only ever be an identifier for one of the *Types* “RESOLVE4” or “RESOLVE6”, but there is **no protection for this built into Traffic Ops** and therefore **any valid *Type* identifier will be silently accepted by Traffic Ops** and so care should be taken to ensure that these *Types* are properly identified. If any *Type* besides “RESOLVE4” or “RESOLVE6” is identified, the resulting resolver *will not work*.

**See also:**

*types* is the endpoint that can be used to determine the identifier for various

## Types

### 362: Request Example

```
POST /api/3.0/federation_resolvers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.63.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 36
Content-Type: application/json

{
  "ipAddress": "::1/1",
  "typeId": 37
}
```

## Response Structure

**id** The integral, unique identifier of the resolver

**ipAddress** The IP address or CIDR-notation subnet of the resolver - may be IPv4 or IPv6

**type** The *Type* of the resolver

**typeId** The integral, unique identifier of the *Type* of the resolver

### 363: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳e9D8JNrQb64xpuDwoBwbISSWUkDGCL2l37NuDXsXsPYof2EqmeHondD8NzxDSwWNJ8d9B9DXpZDbRUtgd
X-Server-Name: traffic_ops_golang/
Date: Wed, 06 Nov 2019 00:00:40 GMT
Content-Length: 153

{ "alerts": [
  {
    "text": "Federation Resolver created [ IP = ::1/1 ] with id: 1
  },
```

(continues on next page)

(continued from previous page)

```

        "level": "success"
      }
    ],
    "response": {
      "id": 1,
      "ipAddress": "::1/1",
      "type": "RESOLVE6",
      "typeId": 37
    }
  }
}
```

## DELETE

Deletes a federation resolver.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 173: Request Query Parameters

Name	Required	Description
id	yes	Integral, unique identifier for the federation resolver to be deleted



### 364: Request Example

```
DELETE /api/3.0/federation_resolvers?id=4 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

**id** The integral, unique identifier of the resolver

**ipAddress** The IP address or CIDR-notation subnet of the resolver - may be IPv4 or IPv6

**type** The *Type* of the resolver

### 365: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; HttpOnly
Whole-Content-Sha512:
→2v4LYQdRVhaFJVd86Iv1BWVYzNPSlzpQ222bUB7Zz+Ss8A48FNyHZjPlq5a+a4g9KAQCTUIytWnIQk+L1
X-Server-Name: traffic_ops_golang/
Date: Fri, 08 Nov 2019 23:19:01 GMT
Content-Length: 161

{ "alerts": [
  {
    "text": "Federation resolver deleted [ IP = 1.2.6.4/22 ] with
→id: 4",
    "level": "success"
  }
],
"response": {
  "id": 4,
  "ipAddress": "1.2.6.4/22",
```

(continues on next page)

(continued from previous page)

```
"type": "RESOLVE6"
}}
```

## federations

### GET

Retrieves a list of *Federation* mappings (i.e. *Federation* Resolvers) for the current user.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Array

## Request Structure

No parameters available.

## Response Structure

**deliveryService** The `xml_id` that uniquely identifies the *Delivery Service* that uses the federation mappings in mappings

**mappings** An array of objects that represent the mapping of a *Federation*’s CNAME to one or more Resolvers

**cname** The actual CNAME used by the *Federation*

**resolve4** An array of IPv4 addresses (or subnets in CIDR notation) capable of resolving the *Federation*’s CNAME

**resolve6** An array of IPv6 addresses (or subnets in CIDR notation) capable of resolving the *Federation*’s CNAME

**ttl** The TTL of the CNAME in hours

### 366: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512:↵
↵d6Llm5qNc2sfgVH9IimW7hA4wvtBUq6EzUmpJf805kB0k6v2WysNgFEWK4hBXNdAYkr8hYuKPrwDy3tCx
X-Server-Name: traffic_ops_golang/
Date: Mon, 03 Dec 2018 17:19:13 GMT
Content-Length: 136

{ "response": [
  {
    "mappings": [
      {
        "ttl": 300,
        "cname": "blah.blah.",
        "resolve4": [
          "0.0.0.0/32"
        ],
        "resolve6": [
          "::/128"
        ]
      }
    ],
    "deliveryService": "demo1"
  }
]}
```

## POST

Allows a user to create *Federation* Resolvers for *Delivery Services*, providing the *Delivery Service* is within a CDN that has some associated *Federation*.

**Warning:** Confusingly, this method of this endpoint does **not** create a new *Federation*; to do that, the *cdns/{name}/federations* endpoint must be used. Furthermore, the *Federation* must properly be assigned to a *Delivery Service* using the *federations/{ID}/deliveryservices* and assigned to the user creating Resolvers using *federations/{ID}/users*.

### See also:

The *federations/{ID}/federation\_resolvers* endpoint duplicates this functionality.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object (string)

## Request Structure

The request payload is an array of objects that describe Delivery Service *Federation* Resolver mappings. Each object in the array must be in the following format.

**deliveryService** The *xml\_id* of the *Delivery Service* which will use the *Federation* Resolvers specified in mappings

**mappings** An object containing two arrays of IP addresses (or subnets in CIDR notation) to use as *Federation* Resolvers

**resolve4** An array of IPv4 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service's Federation*

**resolve6** An array of IPv6 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service's Federation*

### 367: Request Example

```
POST /api/3.0/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 118
Content-Type: application/json

[{"deliveryService": "demo1",
  "mappings": {
    "resolve4": ["127.0.0.1", "0.0.0.0/32"],
    "resolve6": ["::1", "5efa::ff00/128"]
  }
}]
```

## Response Structure

### 368: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
  ↳GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: B7TSUOYZPRPyi3mVy+CuxiXR5k/
→d0s07w4i6kYzpWS+YL79juEfkusqfedayG/kMA809XbjkWRjcBAdxOVrdTQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 23 Oct 2019 22:28:02 GMT
Content-Length: 152

{ "alerts": [
  {
    "text": "admin successfully created federation resolvers.",
    "level": "success"
  }
],
"response": "admin successfully created federation resolvers."
}
```

## DELETE

Deletes **all** *Federation* Resolvers associated with the logged-in user's *Federations*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object (string)

## Request Structure

No parameters available

### 369: Request Example

```
DELETE /api/3.0/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 370: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→fd7P45mIiHuYqZZW6+8K+YjY1Pe504Aaw4J4Zp9AhrqLX72ERYtTqWtAp1msutzNSRUdUSC72+odNPtpv
X-Server-Name: traffic_ops_golang/
Date: Wed, 23 Oct 2019 23:34:53 GMT
Content-Length: 184

{ "alerts": [
  {
    "text": "admin successfully deleted all federation resolvers:
→[ 8.8.8.8 ]",
    "level": "success"
  }
],
"response": "admin successfully deleted all federation resolvers: [ 8.
→8.8.8 ]"
}
```

## PUT

Replaces **all** *Federations* associated with a user's *Delivery Service*(s) with those defined inside the request payload.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Response Type** Object (string)

## Request Structure

The request payload is an array of objects that describe Delivery Service *Federation* Resolver mappings. Each object in the array must be in the following format.

**deliveryService** The *xml\_id* of the *Delivery Service* which will use the *Federation* Resolvers specified in **mappings**

**mappings** An object containing two arrays of IP addresses (or subnets in CIDR notation) to use as *Federation* Resolvers

**resolve4** An array of IPv4 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service*'s *Federation*

**resolve6** An array of IPv6 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service*'s *Federation*

### 371: Request Example

```
PUT /api/3.0/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 95
Content-Type: application/json

[{"mappings": {
  "resolve4": ["8.8.8.8"],
  "resolve6": []
},
"deliveryService": "demo1"
}]
```

## Response Structure

### 372: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dQ5AvQULhc254zQwgUpBl1/CHbLr/
→clKtkbs0Ju9f1BM4xIfbb03puFNN9zaEaZ1iz0lBvHFp/PgfUqisD3QHA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 23 Oct 2019 23:22:03 GMT
Content-Length: 258
Content-Type: application/json

{ "alerts": [
  {
    "text": "admin successfully deleted all federation resolvers:
→[ 8.8.8.8 ]",
    "level": "success"
  },
  {
    "text": "admin successfully created federation resolvers.",
    "level": "success"
  }
]
```

(continues on next page)



(continued from previous page)

```
}
],
"response": "admin successfully created federation resolvers."
}
```

## federations/all

### GET

Retrieves a list of *Federation* mappings (also called *Federation* Resolvers) for the current user.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Array

### Request Structure

No parameters available.

#### 373: Request Example

```
GET /api/3.0/federations/all HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

**deliveryService** The *xml\_id* of the delivery service.

**mappings** An array of objects that represent the mapping of a *Federation*’s CNAME to one or more Resolvers

**cname** The actual CNAME used by the *Federation*

**ttd** The TTL of the CNAME in hours

#### 374: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 21:38:06.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→UQBlGVPJytYMkv0V42EAIoJUnXjBTCXnOGp0berxte6TtnX63LTAKFfD2LejBVYXkKtnCdkBbs+SzhA0H.
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 20:38:06 GMT
Content-Length: 138
```

```
{
  "response": [
    {
      "mappings": [
        {
          "ttl": 60,
          "cname": "img1.mcdn.ciab.test."
        },
        {
          "ttl": 60,
          "cname": "img2.mycdn.ciab.test."
        }
      ],
      "deliveryService": "demo1"
    },
    {
      "mappings": [
        {
          "ttl": 60,
          "cname": "static.mycdn.ciab.test."
        }
      ],
      "deliveryService": "demo2"
    }
  ]
}
```

`federations/{{ID}}/deliveryservices`

**GET**

Retrieves *Delivery Services* assigned to a *Federation*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 174: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation to be inspected

Table 175: Request Query Parameters

Name	Required	Description
dsID	no	Show only the <i>Delivery Service</i> identified by this integral, unique identifier
order-by	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 375: Request Example

```
GET /api/3.0/federations/1/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cdn** The CDN to which this *Delivery Service* Belongs
- id** The integral, unique identifier for the *Delivery Service*
- type** The routing type used by this *Delivery Service*
- xmlId** The 'xml\_id' which uniquely identifies this *Delivery Service*

### 376: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:44:13 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:44:13 GMT;
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: 7Y9Q/
→qHeXfbjJduvucRCR85wf4VRfyYhlK59sNRkzIJuwnsMhFcEfYfNqrvELwfexOum/
→VEX2f/1oa+I/edGfw==
content-length: 74

{ "response": [
  {
    "xmlId": "demo1",
    "cdn": "CDN-in-a-Box",
    "type": "HTTP",
    "id": 1
  }
]}
```

## POST

Assigns one or more *Delivery Services* to a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 176: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation to be inspected

**dsIds** An array of integral, unique identifiers for *Delivery Services* which will be assigned to this federation

**replace** An optional boolean (default: `false`) which, if `true`, will cause any conflicting assignments already in place to be overridden by this request

---

**Note:** If `replace` is not given (and/or not `true`), then any conflicts with existing assignments will cause the entire operation to fail.

---

### 377: Request Example

```
POST /api/3.0/federations/1/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 32
Content-Type: application/json

{
  "dsIds": [1],
  "replace": true
}
```

## Response Structure

**dsIds** An array of integral, unique identifiers for *Delivery Services* which are now assigned to this federation

**replace** An optional boolean (default: false) which, if true, means any conflicting assignments already in place were overridden by this request

### 378: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
→rVd0nx8G3bRI8ub1zw6FTdmwQ7jer4zoqzOZf5tC1ckrR0HEIOH1Azdcmvv0FVE5I0omCHVnrYbzab7tU
x-server-name: traffic_ops_golang/
content-length: 137
date: Wed, 05 Dec 2018 00:34:06 GMT

{ "alerts": [
  {
    "text": "1 delivery service(s) were assigned to the federation.
→1",
    "level": "success"
  }
],
```

(continues on next page)

(continued from previous page)

```
"response": {
  "dsIds": [
    1
  ],
  "replace": true
}}
```

`federations/{{ID}}/deliveryservices/{{dsID}}`

## DELETE

Removes a *Delivery Service* from a federation. A *Delivery Service* cannot be removed from a federation if it is the only *Delivery Service* assigned to said federation

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** undefined

## Request Structure

Table 177: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the federation from which the <i>Delivery Service</i> identified by dsID will be removed
dsID	The integral, unique identifier of the <i>Delivery Service</i> which will be removed from the federation identified by ID

### 379: Request Example

```
DELETE /api/3.0/federations/1/deliveryservices/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 380: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 01:06:51 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 05:06:51 GMT;
    ↳path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: NqAZuZYlF1UW0aazbj/
    ↳j4gWX7ye0kGGakRRFEkK6ShxqXvCxEOdCTyu75qiLPN2wSgr3FGQnp2Sq345sE7In9g==
content-length: 98

{ "alerts": [
  {
    "level": "success",
    "text": "federation deliveryservice was deleted."
  }
]}
```

**federations/{ID}/federation\_resolvers**

**GET**

Retrieves federation resolvers assigned to a federation.

**Auth. Required** Yes

**Roles Required** None



**Response Type** Array

## Request Structure

Table 178: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation for which resolvers will be retrieved

### 381: Request Example

```
GET /api/3.0/federations/1/federation_resolvers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**id** The integral, unique identifier of this federation resolver

**ipAddress** The IP address of the federation resolver - may be IPv4 or IPv6

**type** The type of resolver - one of:

**RESOLVE4** This resolver is for IPv4 addresses (and **ipAddress** is IPv4)

**RESOLVE6** This resolver is for IPv6 addresses (and **ipAddress** is IPv6)

### 382: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:49:50 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:49:50 GMT;
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: csC18kE3YjiILHP1wmJg7V4h/
→XWY8HUMKyPuZWnde2g7HJ4gTY51HfjCSqhyKvIJQ8Rl7uEqshF3Ey6xIMOX4A==
content-length: 63
```

(continues on next page)

(continued from previous page)

```
{ "response": [
  {
    "ipAddress": "0.0.0.0",
    "type": "RESOLVE4",
    "id": 1
  }
]}
```

## POST

Assigns one or more resolvers to a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

Table 179: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation for which resolvers will be retrieved

**fedResolverIds** An array of integral, unique identifiers for federation resolvers

**replace** An optional boolean (default: `false`) which, if `true`, will cause any conflicting assignments already in place to be overridden by this request

---

**Note:** If `replace` is not given (and/or not `true`), then any conflicts with existing assignments will cause the entire operation to fail.

---

## 383: Request Example

```
POST /api/3.0/federations/1/federation_resolvers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 41
Content-Type: application/json

{
```

(continues on next page)

(continued from previous page)

```
"fedResolverIds": [1],
"replace": true
}
```

## Response Structure

**fedResolverIds** An array of integral, unique identifiers for federation resolvers

**replace** An optionally-present boolean (default: `false`) which, if `true`, any conflicting assignments already in place were overridden by this request

### 384: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:47:47 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:47:47 GMT;
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: +JDcRByS3H06pMg3Gzkvn0w7/
→v5oRu19e+RxyFIOkJKNH0kZILyQBS+PJpxDeCgwI19+0poW5dyHPPR9SwbNCA==
content-length: 148

{ "alerts": [
  {
    "level": "success",
    "text": "1 resolver(s) were assigned to the test.quest.
→federation"
  }
],
"response": {
  "replace": true,
  "fedResolverIds": [
    1
  ]
}}
```

`federations/{{ID}}/users`

**GET**

Retrieves users assigned to a federation.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 180: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the federation for which users will be retrieved

Table 181: Request Query Parameters

Name	Re-quired	Description
userId	no	Show only the user that has this integral, unique identifier
role	no	Show only the users that have this role
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

## Response Structure

**company** The company to which the user belongs

**email** The user’s email address

**fullName** The user’s full name

**id** An integral, unique identifier for the user

**role** The user’s highest role

**username** The user’s short “username”

### 385: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:31:34 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:31:34 GMT;
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: eQQoF2xlbK2I2oTja7zrt/
→FlkLzCgwpU2zb2+rmIjHbHJ3MnmsSczSamIAAyTzs5gDaqcuUX1G35ZB8d7Bj82g==
content-length: 101

{ "response": [
  {
    "fullName": null,
    "email": null,
    "id": 2,
    "role": "admin",
    "company": null,
    "username": "admin"
  }
]}
```

## POST

Assigns one or more users to a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

**userIds** An array of integral, unique identifiers for users which will be assigned to this federation

**replace** An optional boolean (default: `false`) which, if `true`, will cause any conflicting assignments already in place to be overridden by this request

---

**Note:** If `replace` is not given (and/or not `true`), then any conflicts with existing assignments will cause the entire operation to fail.

---

### 386: Request Example

```
POST /api/3.0/federations/1/users HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 34
Content-Type: application/json

{
  "userIds": [2],
  "replace": true
}
```

## Response Structure

**userIds** An array of integral, unique identifiers for users which have been assigned to this federation

**replace** An optional boolean (default: `false`) which, if `true`, caused any conflicting assignments already in place to be overridden by this request

### 387: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
➔Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:29:19 GMT
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

(continued from previous page)

```
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:29:19 GMT;␣
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512:␣
→MvPmgOAs58aSOgVh+iEilflgOexbaexg+qE2IPrQZX0H4iSX4JvEys9adbGE9a9yaLj9uUMxg77N6ZyDhV
content-length: 137

{ "alerts": [
  {
    "level": "success",
    "text": "1 user(s) were assigned to the test.quest. federation"
  }
],
"response": {
  "userIds": [
    2
  ],
  "replace": true
}}
```

`federations/{{ID}}/users/{{userID}}`

## DELETE

Removes a user from a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** undefined

## Request Structure

Table 182: Request Path Parameters

Name	Description
ID	An integral, unique identifier for the federation from which the user identified by userID will be removed
userID	An integral, unique identifier for the user who will be removed from the federation identified by ID

### 388: Request Structure

```
DELETE /api/3.0/federations/1/users/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 389: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 01:14:04 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 05:14:04 GMT;
    ↳path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512:
    ↳xdF6l7jdd2t8au6lh4pFtDqYxTfehake2aDBuytL7I74hK9KCT7ssLuYbfvD8ejdqqF3+jjBiFk7neQ8c
content-length: 93

{ "alerts": [
  {
    "level": "success",
    "text": "Removed user [ admin ] from federation [ foo.bar. ]"
  }
]}
```

**isos**

**POST**

Generates an ISO from the requested ISO source.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”



**Response Type** undefined - ISO image as a streaming download

## Request Structure

**dhcp** A string that specifies whether the generated system image will use DHCP IP address leasing; one of:

**yes** DHCP will be used, and other network configuration keys need not be present in the request (and are ignored if they are)

**no** DHCP will not be used, and the desired network configuration **must** be specified manually in the request body

**disk** An optional string that names the block device (under /dev/) used for the boot media, e.g. “sda”

**domainName** The domain part of the system image’s Fully Qualified Domain Name (FQDN)

**hostName** The host name part of the system image’s FQDN

**interfaceMtu** A number that specifies the Maximum Transmission Unit (MTU) for the system image’s network interface card - the only valid values of which I’m aware are 1500 or 9000, and this should almost always just be 1500

**interfaceName** An optional string naming the network interface to be used by the generated system image e.g. “bond0”, “eth0”, etc. If the special name “bond0” is used, an LACP (Link Aggregation Control Protocol) binding configuration will be created and included in the system image

**See also:**

[The Link Aggregation Wikipedia page.](#)

**ip6Address** An optional string containing the IPv6 address of the generated system image

**ip6Gateway** An optional string specifying the IPv6 address of the generated system image’s network gateway - this will be ignored if **ipGateway** is specified

**ipAddress** An optional<sup>1</sup> string containing the IP address of the generated system image

**ipGateway** An optional<sup>1</sup> string specifying the IP address of the generated system image’s network gateway

**ipNetmask** An optional<sup>1</sup> string specifying the subnet mask of the generated system image

**osversionDir** The name of the directory containing the ISO source

**See also:**

[osversions](#)

---

<sup>1</sup> This optional key is required if and only if **dhcp** is “no”.

**rootPass** The password used by the generated system image's root user

### 390: Request Example

```
POST /api/3.0/isos HTTP/1.1
Host: some.trafficops.host
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 334
Content-Type: application/json

{
  "osversionDir": "centos72",
  "hostName": "test",
  "domainName": "quest",
  "rootPass": "twelve",
  "dhcp": "no",
  "interfaceMtu": 1500,
  "ipAddress": "1.3.3.7",
  "ipNetmask": "255.255.255.255",
  "ipGateway": "8.0.0.8",
  "ip6Address": "1::3:3:7",
  "ip6Gateway": "8::8",
  "interfaceName": "eth0",
  "disk": "hda"
}
```

## Response Structure

ISO image as a streaming download.

### 391: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Connection: keep-alive
Content-Disposition: attachment; filename="test-centos72_centos72-
↳netinstall.iso"
Content-Encoding: gzip
Content-Type: application/download
Date: Wed, 05 Feb 2020 21:59:15 GMT
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 05 Feb 2020 22:59:11
↳GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Transfer-Encoding: chunked
Whole-Content-sha512: sLSVQGrLCQ4hGQhv2reragQHWNi2aKMcz2c/
↪HMAH45tLcZ1LenPy0zWRcRfHUNbV4PEEK0oiTfwE2H1A+WtRIQ==
X-Server-Name: traffic_ops_golang/
```

## jobs

### GET

Retrieve *Content Invalidation Jobs*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

<sup>1</sup> When viewing *Content Invalidation Jobs*, only those *jobs* that operate on a *Delivery Service* visible to the requesting user's *Tenant* will be returned. Likewise, creating a new *Content Invalidation Jobs* requires that the target *Delivery Service* is modifiable by the requesting user's *Tenant*. However, when modifying or deleting an existing *Content Invalidation Jobs*, the operation can be completed if and only if the requesting user's *Tenant* is the same as the *Content Invalidation Job*'s *Delivery Service*'s *Tenant* or a descendant thereof, **and** if the requesting user's *Tenant* is the same as the *Tenant* of the user who initially created the *:term: 'Content Invalidation Job'* or a descendant thereof.

## Request Structure

Table 183: Request Query Parameters

Name	Re-quired	Description
assetUrl	no	Return only <i>Content Invalidation Jobs</i> that operate on URLs by matching this regular expression
cdn	no	Return only <i>Content Invalidation Jobs</i> for delivery services with this CDN name
createdBy	no	Return only <i>Content Invalidation Jobs</i> that were created by the user with this username
delivery-Service	no	Return only <i>Content Invalidation Jobs</i> that operate on the <i>Delivery Service</i> with this <i>xml_id</i>
dsId	no	Return only <i>Content Invalidation Jobs</i> pending on the <i>Delivery Service</i> identified by this integral, unique identifier
id	no	Return only the single invalidation <i>Content Invalidation Job</i> identified by this integral, unique identifier
keyword	no	Return only <i>Content Invalidation Jobs</i> that have this “keyword” - only “PURGE” should exist
maxReval-Duration-Days	no	Return only <i>Content Invalidation Jobs</i> with a <i>startTime</i> that is within the window defined by the <i>maxRevalDurationDays</i> <i>Parameter</i> in <i>The GLOBAL Profile</i>
userId	no	Return only <i>Content Invalidation Jobs</i> created by the user identified by this integral, unique identifier

### 392: Request Example

```
GET /api/3.0/jobs?id=3&dsId=1&userId=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**assetUrl** A regular expression - matching URLs will be operated upon according to keyword

**createdBy** The username of the user who initiated the *Content Invalidation Job*

**deliveryService** The *xml\_id* of the *Delivery Service* on which this *Content Invalidation Job* operates

**id** An integral, unique identifier for this *Content Invalidation Job*

**keyword** A keyword that represents the operation being performed by the *Content Invalidation Job*:

**PURGE** This *Content Invalidation Job* will prevent caching of URLs matching the `assetUrl` until it is removed (or its Time to Live expires)

**parameters** A string containing key/value pairs representing parameters associated with the *Content Invalidation Job* - currently only uses Time to Live e.g. "TTL:48h"

**startTime** The date and time at which the *Content Invalidation Job* began, in a non-standard format

### 393: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: gH41oEi2zrd3y8yo+wfohn4/
→oHU098RpyPnqBzU7HlLUDkMOPKjAZnamcYqfdy7yDCFDUcgqkvbFAvnljxyb8w==
X-Server-Name: traffic_ops_golang/
Date: Tue, 18 Jun 2019 19:47:30 GMT
Content-Length: 186

{ "response": [{
  "assetUrl": "http://origin.infra.ciab.test/*.\"",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "id": 3,
  "keyword": "PURGE",
  "parameters": "TTL:2h",
  "startTime": "2019-06-18 21:28:31+00"
}]}
```

## POST

Creates a new *Content Invalidation Job*.

**Caution:** Creating a *Content Invalidation Job* immediately triggers a CDN-wide revalidation update. In the case that the global *Parameter* `use_reval_pending` has a value of exactly "0", this will instead trigger a CDN-wide "Queue Updates". This means that *Content Invalidation Jobs* become active **immediately** at their `startTime` - unlike most other configuration changes they do not wait for a *Snapshot* or a "Queue Updates". Furthermore, if the global *Parameter* `use_reval_pending` is "0", this will cause all pending configuration changes to propagate to all *cache servers* in the CDN. Take care when using this endpoint.

**Auth. Required** Yes

**Roles Required** "operations" or "admin"<sup>Page 721, 1</sup>

**Response Type** Object

## Request Structure

**deliveryService** This should either be the integral, unique identifier of a *Delivery Service*, or a string containing an *xml\_id*

**startTime** This can be a string in the legacy YYYY-MM-DD HH:MM:SS format, or a string in **RFC 3339** format, or a string representing a date in the same non-standard format as the `last_updated` fields common in other API responses, or finally it can be a number indicating the number of milliseconds since the Unix Epoch (January 1, 1970 UTC). This date must be in the future.

**regex** A regular expression that will be used to match the path part of URIs for content stored on *cache servers* that service traffic for the *Delivery Service* identified by `deliveryService`.

**ttl** Either the number of hours for which the *Content Invalidation Job* should remain active, or a "duration" string, which is a sequence of numbers followed by units. The accepted units are:

- h gives a duration in hours
- m gives a duration in minutes
- s gives a duration in seconds
- ms gives a duration in milliseconds
- us (or μs) gives a duration in microseconds
- ns gives a duration in nanoseconds

These durations can be combined e.g. 2h45m specifies a TTL of two hours and forty-five minutes - however note that durations are always rounded up

to the nearest hour so that e.g. 121m becomes three hours. TTLs cannot ever be negative, obviously.

#### 394: Request Example

```
POST /api/3.0/jobs HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 80
Content-Type: application/json

{
  "deliveryService": "demo1",
  "startTime": 1560893311219,
  "regex": "/.*",
  "ttl": "121m"
}
```

## Response Structure

**assetUrl** A regular expression - matching URLs will be operated upon according to keyword

**createdBy** The username of the user who initiated the *Content Invalidation Job*

**deliveryService** The *xml\_id* of the *Delivery Service* on which this *Content Invalidation Job* operates

**id** An integral, unique identifier for this *Content Invalidation Job*

**keyword** A keyword that represents the operation being performed by the *Content Invalidation Job*:

**PURGE** This *Content Invalidation Job* will prevent caching of URLs matching the `assetUrl` until it is removed (or its Time to Live expires)

**parameters** A string containing key/value pairs representing parameters associated with the *Content Invalidation Job* - currently only uses Time to Live e.g. "TTL:48h"

**startTime** The date and time at which the *Content Invalidation Job* began, in a non-standard format

## 395: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Location: https://trafficops.infra.ciab.test/api/3.0/jobs?id=3
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳nB2xg2IqO56rLT8dI4+KZgx0sTe5ShctG1U8epRsY9NyyMIpx8TZYt5Mr02QikuYh+NnyoR6V0VICnGCI
X-Server-Name: traffic_ops_golang/
Date: Tue, 18 Jun 2019 19:37:06 GMT
Content-Length: 238

{
  "alerts": [
    {
      "text": "Invalidation Job creation was successful",
      "level": "success"
    }
  ],
  "response": {
    "assetUrl": "http://origin.infra.ciab.test/.*",
    "createdBy": "admin",
    "deliveryService": "demo1",
    "id": 3,
    "keyword": "PURGE",
    "parameters": "TTL:2h",
    "startTime": "2019-06-18 21:28:31+00"
  }
}
```

**PUT**

Replaces an existing *Content Invalidation Job* with a new one provided in the request. This method of editing a *Content Invalidation Job* does not prevent the requesting user from changing fields that normally only have one value. Use with care.

**Caution:** Modifying a *Content Invalidation Job* immediately triggers a CDN-wide revalidation update. In the case that the global *Parameter* `use_reval_pending` has a value of exactly `"0"`, this will instead trigger a CDN-wide “Queue Updates”. This means that *Con-*



*Content Invalidation Jobs*

*Snapshot*

*Parameter*

*cache servers*

**Auth. Required** Yes

**Roles Required** “operations” or “admin”<sup>Page 721, 1</sup>

**Response Type** Object

## Request Structure

Table 184: Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Content Invalidation Job</i> being modified

**assetUrl** A regular expression - matching URLs will be operated upon according to keyword

**Note:** Unlike in the payloads of *POST* requests to this endpoint, this must be a **full** URL regular expression, as it is **not** combined with the *Origin Server Base URL* of the *Delivery Service* identified by `deliveryService`.

**createdBy** The username of the user who initiated the *Content Invalidation Job*<sup>2</sup>

**deliveryService** The *xml\_id* of the *Delivery Service* on which this *Content Invalidation Job* operates<sup>2</sup> - unlike *POST* request payloads, this cannot be an integral, unique identifier

**id** An integral, unique identifier for this *Content Invalidation Job*<sup>2</sup>

**keyword** A keyword that represents the operation being performed by the *Content Invalidation Job*. It can have any (string) value, but the only value with any meaning to Traffic Control is:

**PURGE** This *Content Invalidation Job* will prevent caching of URLs matching the `assetUrl` until it is removed (or its Time to Live expires)

**parameters** A string containing space-separated key/value pairs - delimited by colons (:) representing parameters associated with the *Content Invalidation Job*. In practice, any string can be passed as a *Content Invalidation Job*'s

<sup>2</sup> This field must exist, but it must *not* be different than the same field of the existing *Content Invalidation Job* (i.e. as seen in a *GET* response)

parameters, but the only value with meaning is a single key/value pair indicated a TTL in hours in the format `TTL:hoursh`, and any other type of value may cause components of Traffic Control to work improperly or not at all.

**startTime** This can be a string in the legacy `YYYY-MM-DD HH:MM:SS` format, or a string in [RFC 3339](#) format, or a string representing a date in the same non-standard format as the `last_updated` fields common in other API responses, or finally it can be a number indicating the number of milliseconds since the Unix Epoch (January 1, 1970 UTC). This **must** be in the future, but only by no more than two days.

#### 396: Request Example

```
PUT /api/3.0/jobs?id=3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 188
Content-Type: application/json

{
  "assetUrl": "http://origin.infra.ciab.test/.*",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "id": 3,
  "keyword": "PURGE",
  "parameters": "TTL:360h",
  "startTime": "2019-06-20 18:33:40+00"
}
```

## Response Structure

**assetUrl** A regular expression - matching URLs will be operated upon according to keyword

**createdBy** The username of the user who initiated the *Content Invalidation Job*

**deliveryService** The *xml\_id* of the *Delivery Service* on which this *Content Invalidation Job* operates

**id** An integral, unique identifier for this *Content Invalidation Job*

**keyword** A keyword that represents the operation being performed by the *Content Invalidation Job*:

**PURGE** This *Content Invalidation Job* will prevent caching of URLs matching the `assetUrl` until it is removed (or its Time to Live expires)

**parameters** A string containing key/value pairs representing parameters associated with the *Content Invalidation Job* - currently only uses Time to Live e.g. "TTL:48h"

**startTime** The date and time at which the *Content Invalidation Job* began, in a non-standard format

#### 397: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳+P1PTav4ZBoiQcCqQnUqf+J0dCfQgVj8mzzKtUCA69mWYulya9Bjf6BUd8Aro2apmpgPBkCEA5sITJV1t
X-Server-Name: traffic_ops_golang/
Date: Wed, 19 Jun 2019 13:38:59 GMT
Content-Length: 234

{ "alerts": [{
    "text": "Content invalidation job updated",
    "level": "success"
}],
  "response": {
    "assetUrl": "http://origin.infra.ciab.test/.*",
    "createdBy": "admin",
    "deliveryService": "demo1",
    "id": 3,
    "keyword": "PURGE",
    "parameters": "TTL:360h",
    "startTime": "2019-06-20 18:33:40+00"
  }}

```

## DELETE

Deletes a *Content Invalidation Job*.

---

**Tip:** Content *Content Invalidation Jobs* that have passed their TTL are not automatically deleted - for record-keeping purposes - so use this to clean up old *jobs* that are no longer useful.

---

**Caution:** Deleting a *Content Invalidation Job* immediately triggers a CDN-wide revalidation update. In the case that the global *Parameter* `use_reval_pending` has a value of exactly "0", this will instead trigger a CDN-wide "Queue Updates". This means that *Content Invalidation Jobs* become active **immediately** at their `startTime` - unlike most other configuration changes they do not wait for a *Snapshot* or a "Queue Updates". Furthermore, if the global *Parameter* `use_reval_pending` is "0", this will cause all pending configuration changes to propagate to all *cache servers* in the CDN. Take care when using this endpoint.

**Auth. Required** Yes

**Roles Required** "operations" or "admin"<sup>Page 721, 1</sup>

**Response Type** Object

## Request Structure

Table 185: Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Content Invalidation Job</i> being modified

398: Request Example

```
DELETE /api/3.0/jobs?id=3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

**assetUrl** A regular expression - matching URLs will be operated upon according to keyword

**createdBy** The username of the user who initiated the *Content Invalidation Job*

**deliveryService** The *xml\_id* of the *Delivery Service* on which this *Content Invalidation Job* operates

**id** An integral, unique identifier for this *Content Invalidation Job*

**keyword** A keyword that represents the operation being performed by the *Content Invalidation Job*:

**PURGE** This *Content Invalidation Job* will prevent caching of URLs matching the `assetUrl` until it is removed (or its Time to Live expires)

**parameters** A string containing key/value pairs representing parameters associated with the *Content Invalidation Job* - currently only uses Time to Live e.g. "TTL:48h"

**startTime** The date and time at which the *Content Invalidation Job* began, in a non-standard format

### 399: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: FqfziXJYYwHb84Fac9+p4NEY3EsklYxe94wg/
    ↳V0mlXk4R6l4SaPSh015CChPt/yT72MsWSETnIuRD9KtoK4I+w==
X-Server-Name: traffic_ops_golang/
Date: Tue, 18 Jun 2019 22:55:15 GMT
Content-Length: 234

{ "alerts": [
  {
    "text": "Content invalidation job was deleted",
    "level": "success"
  }
],
"response": {
  "assetUrl": "http://origin.infra.ciab.test/*.\"",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "id": 3,
  "keyword": "PURGE",
  "parameters": "TTL:36h",
  "startTime": "2019-06-20 18:33:40+00"
}}
```

### letsencrypt/autorenew

#### POST

Generates an SSL certificate and private key using Let's Encrypt for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

#### Request Structure

No parameters available

#### Response Structure

400: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "alerts": [
  {
    "text": "This endpoint is deprecated, please use letsencrypt/
↪autorenew instead",
    "level": "warning"
  },
  {
    "text": "Beginning async call to renew certificates. This may.
↪take a few minutes.",
    "level": "success"
  }
]}
```

### letsencrypt/dnsrecords

#### GET

Gets DNS challenge records for Let's Encrypt DNS challenge for a specified FQDN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Array

## Request Structure

Table 186: Request Query Parameters

Name	Required	Description
fqdn	no	Return only DNS challenge records for the specified FQDN

### 401: Request Example

```
GET /api/3.0/letsencrypt/dnsrecord?fqdn=_acme-challenge.demo1.example.
→com. HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**fqdn** The FQDN for the TXT record location to complete the DNS challenge

**record** The record provided by Let's Encrypt to complete the DNS challenge

### 402: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": [
  {
    "fqdn": "_acme-challenge.demo1.example.com.",
    "record": "testRecord"
  }
]}
```

## logs

**Note:** This endpoint's responses will contain a cookie (`last_seen_log`) that is used by `logs/newcount` to determine the time of last access. Be sure your client uses cookies properly if you intend to use `logs/newcount` in concert with this endpoint!

**GET**

Fetches a list of changes that have been made to the Traffic Control system

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

**Request Structure**

Table 187: Request Query Parameters

Name	Required	Description
days	no	An integer number of days of change logs to return, by default there is no limit applied
limit	no	The number of records to which to limit the response, if there is no limit query params, it limits to 1000
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .
username	no	A name to which to limit the response too

New in version ATCv6: The `username`, `page`, `offset` query parameters were added to this in endpoint across across all API versions in ATC version 6.0.0.



#### 403: Request Example

```
GET /api/3.0/logs?days=1&limit=2&username=admin HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

- id** Integral, unique identifier for the Log entry
- lastUpdated** Date and time at which the change was made, in *Traffic Ops's Custom Date/Time Format*
- level** Log categories for each entry, e.g. 'UICHANGE', 'OPER', 'APICHANGE'
- message** Log detail about what occurred
- ticketNum** Optional field to cross reference with any bug tracking systems
- user** Name of the user who made the change

#### 404: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 15:11:38 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: last_seen_log="2018-11-15% 15:11:38"; path=/; Max-
➔Age=604800
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
➔GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
➔40dV+aZaZ3b6F30y6YHVbV3H2a3ekZrdoxICupwaxQnj62pwYfb7YCM7Qhe3OAItmB77Tbg9INy27ymaz3
Content-Length: 357

{ "response": [
  {
    "ticketNum": null,
    "level": "APICHANGE",
```

(continues on next page)

(continued from previous page)

```

        "lastUpdated": "2018-11-14 21:40:06.493975+00",
        "user": "admin",
        "id": 444,
        "message": "User [ test ] unlinked from deliveryservice [ 1 |
→demo1 ]".
    },
    {
        "ticketNum": null,
        "level": "APICHANGE",
        "lastUpdated": "2018-11-14 21:37:30.707571+00",
        "user": "admin",
        "id": 443,
        "message": "1 delivery services were assigned to test"
    }
  ],
  "summary": {
    "count": 2
  }
}

```

## Summary Fields

The summary object returned by this method of this endpoint uses only the count *standard property*.

## logs/newcount

### GET

Gets the number of new changes made to the Traffic Control system - “new” being defined as the last time the client requested either *logs*

---

**Note:** This endpoint’s functionality is implemented by the *logs* endpoint’s response setting cookies for the client to use when requesting `_this_` endpoint. Take care that your client respects cookies!

---

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

No parameters available

## Response Structure

**newLogcount** The integer number of new changes

405: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 15:17:35 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: Ugdqe8GXKSOExphwbDX/
    ↳Gli+2vBpubttbpfYMbJaCP7adox3MzmVRi2RxTDL5kwPewrcL1C088zGITskh0sc9g==
Content-Length: 30

{ "response": {
  "newLogcount": 4
}}
```

## origins

### GET

Gets all requested *Origins*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 188: Request Query Parameters

Name	Re- quired	Description
cacheGroup	no	Return only <i>Origins</i> within the <i>Cache Group</i> that has this <i>ID</i>
co- ordi- nate	no	Return only <i>Origins</i> located at the geographic coordinates identified by this integral, unique identifier
de- liv- ery- ser- vice	no	Return only <i>Origins</i> that belong to the <i>Delivery Service</i> identified by this integral, unique identifier
id	no	Return only the <i>Origin</i> that has this integral, unique identifier
name	no	Return only <i>Origins</i> by this name
profileId	no	Return only <i>Origins</i> which use the <i>Profile</i> that has this <i>ID</i>
pri- mary	no	If <code>true</code> , return only <i>Origins</i> which are the the primary <i>Origin</i> of the <i>Delivery Service</i> to which they belong - if <code>false</code> return only <i>Origins</i> which are <i>not</i> the primary <i>Origin</i> of the <i>Delivery Service</i> to which they belong
tenant	no	Return only <i>Origins</i> belonging to the tenant identified by this integral, unique identifier
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the $n^{\text{th}}$ page of results, where “ $n$ ” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

**Note:** Several fields of origin definitions which are filterable by Query Parameters are allowed to be `null`. `null` values in these fields will be filtered *out* appropriately by such Query Parameters, but do note that `null` is not a valid value accepted by any of these Query Parameters, and attempting to pass it will result in an error.

### 406: Request Example

```
GET /api/3.0/origins?name=demo1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cachegroup** A string that is the *name of the Cache Group* to which the *Origin* belongs
- cachegroupId** An integer that is the *ID of the Cache Group* to which the *Origin* belongs
- coordinate** The name of a coordinate pair that defines the origin's geographic location
- coordinateId** An integral, unique identifier for the coordinate pair that defines the *Origin's* geographic location
- deliveryService** A string that is the *xml\_id* of the *Delivery Service* to which the *Origin* belongs
- deliveryServiceId** An integral, unique identifier for the *Delivery Service* to which the *Origin* belongs
- fqdn** The FQDN of the *Origin*
- id** An integral, unique identifier for this *Origin*
- ip6Address** The IPv6 address of the *Origin*
- ipAddress** The IPv4 address of the *Origin*
- isPrimary** A boolean value which, when `true` specifies this *Origin* as the 'primary' *Origin* served by `deliveryService`
- lastUpdated** The date and time at which this *Origin* was last modified
- name** The name of the *Origin*
- port** The TCP port on which the *Origin* listens
- profile** The *Name* of the *Profile* used by this *Origin*
- profileId** The *ID* of the *Profile* used by this *Origin*
- protocol** The protocol used by this origin - will be one of 'http' or 'https'
- tenant** The name of the *Tenant* that owns this *Origin*
- tenantId** An integral, unique identifier for the *Tenant* that owns this *Origin*

### 407: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→sm8DpvdvrfdSVLtmXTdfjsZbTlbc+pI40Gy0aj00XIURTPfFXuv/
→4LgHb6A3r92iybRHvFrH6qdB2g97U2sBg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 15:43:41 GMT
Content-Length: 376
```

```
{ "response": [
  {
    "cachegroup": null,
    "cachegroupId": null,
    "coordinate": null,
    "coordinateId": null,
    "deliveryService": "demo1",
    "deliveryServiceId": 1,
    "fqdn": "origin.infra.ciab.test",
    "id": 1,
    "ip6Address": null,
    "ipAddress": null,
    "isPrimary": true,
    "lastUpdated": "2018-12-10 19:11:32+00",
    "name": "demo1",
    "port": null,
    "profile": null,
    "profileId": null,
    "protocol": "http",
    "tenant": "root",
    "tenantId": 1
  }
]}
```

## POST

Creates a new origin definition.

**Warning:** At the time of this writing it is possible to create and/or modify origin definitions assigned to STEERING and CLIENT\_STEERING *Delivery Services* - despite that an origin has no meaning in those contexts. In these cases, the API responses may give incorrect output - see [GitHub Issue #3107](#) for details and updates.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**cachegroupId** An optional, integer which, if present, should be the *Cache Group ID* that identifies a *Cache Group* to which the new *Origin* shall belong

**coordinateId** An optional, integral, unique identifier of a coordinate pair that shall define the *Origin*’s geographic location

**deliveryServiceId** The integral, unique identifier of the *Delivery Service* to which the new *Origin* shall belong

**fqdn** The FQDN of the *Origin*

**ip6Address** An optional string containing the IPv6 address of the *Origin*

**ipAddress** An optional string containing the IPv4 address of the *Origin*

**isPrimary** An optional boolean which, if `true` will set this *Origin* as the ‘primary’ *Origin* served by the *Delivery Service* identified by `deliveryServiceID`

**Note:** Though not specifying this field in this request will leave it as `null` in the output, Traffic Ops will silently coerce that to its default value: `false`.

**name** A human-friendly name of the *Origin*

**port** An optional port number on which the *Origin* listens for incoming TCP connections

**profileId** An optional *ID* of a *Profile* that shall be used by this *Origin*

**protocol** The protocol used by the origin - must be one of ‘http’ or ‘https’

**tenantId** An optional<sup>1</sup>, integral, unique identifier for the *Tenant* which shall own the new *Origin*

<sup>1</sup> The `tenantId` field is required if and only if tenancy is enabled within Traffic Ops.

#### 408: Request Example

```
POST /api/3.0/origins HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 114
Content-Type: application/json

{
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "name": "example",
  "port": 80,
  "protocol": "http",
  "tenantId": 1
}
```

### Response Structure

- cachegroup** A string that is the *name of the Cache Group* to which the *Origin* belongs
- cachegroupId** An integer that is the *ID of the Cache Group* to which the *Origin* belongs
- coordinate** The name of a coordinate pair that defines the origin's geographic location
- coordinateId** An integral, unique identifier for the coordinate pair that defines the *Origin's* geographic location
- deliveryService** The 'xml\_id' of the *Delivery Service* to which the *Origin* belongs
- deliveryServiceId** An integral, unique identifier for the *Delivery Service* to which the *Origin* belongs
- fqdn** The FQDN of the *Origin*
- id** An integral, unique identifier for this *Origin*
- ip6Address** The IPv6 address of the *Origin*
- ipAddress** The IPv4 address of the *Origin*
- isPrimary** A boolean value which, when `true` specifies this *Origin* as the 'primary' *Origin* served by *deliveryService*
- lastUpdated** The date and time at which this *Origin* was last modified
- name** The name of the *Origin*



- port** The TCP port on which the *Origin* listens
- profile** The *Name* of the *Profile* used by this *Origin*
- profileId** The *ID* the *Profile* used by this *Origin*
- protocol** The protocol used by this origin - will be one of 'http' or 'https'
- tenant** The name of the *Tenant* that owns this *Origin*
- tenantId** An integral, unique identifier for the *Tenant* that owns this *Origin*

#### 409: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳z4gp0MaqYu+gSR0RhKT2e0bVBuVDVx1rdteRaN5kRL9uJ3hNzUCi4dSKIt0rgNgOEDt6x/
    ↳iTyrMvhr/TSHYtmA==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 15:14:27 GMT
Content-Length: 418

{ "alerts": [
  {
    "text": "origin was created.",
    "level": "success"
  }
],
"response": {
  "cachegroup": null,
  "cachegroupId": null,
  "coordinate": null,
  "coordinateId": null,
  "deliveryService": null,
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "id": 2,
  "ip6Address": null,
  "ipAddress": null,
  "isPrimary": null,
  "lastUpdated": "2018-12-11 15:14:27+00",
  "name": "example",
```

(continues on next page)

(continued from previous page)

```
"port": 80,
"profile": null,
"profileId": null,
"protocol": "http",
"tenant": null,
"tenantId": 1
}}
```

## PUT

Updates an *Origin* definition.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 189: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the <i>Origin</i> definition being edited

**cachegroupId** An optional, integer which, if present, should be the *Cache Group ID* that identifies a *Cache Group* to which the new *Origin* shall belong

**coordinateId** An optional, integral, unique identifier of a coordinate pair that shall define the *Origin*’s geographic location

**deliveryServiceId** The integral, unique identifier of the *Delivery Service* to which the *Origin* shall belong

**fqdn** The FQDN of the *Origin*

**ip6Address** An optional string containing the IPv6 address of the *Origin*

**ipAddress** An optional string containing the IPv4 address of the *Origin*

**isPrimary** An optional boolean which, if `true` will set this *Origin* as the ‘primary’ origin served by the *Delivery Service* identified by `deliveryServiceID`

**name** A human-friendly name of the *Origin*

**port** An optional port number on which the *Origin* listens for incoming TCP connections

**profileId** An optional *ID* of the *Profile* that shall be used by this *Origin*

**protocol** The protocol used by the *Origin* - must be one of ‘http’ or ‘https’

**tenantId** An optional<sup>Page 743, 1</sup>, integral, unique identifier for the *Tenant* which shall own the new *Origin*

#### 410: Request Example

```
PUT /api/3.0/origins?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 135
Content-Type: application/json

{
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "isprimary": true,
  "name": "example",
  "port": 443,
  "protocol": "https",
  "tenantId": 1
}
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the *Origin* belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the *Origin* belongs

**coordinate** The name of a coordinate pair that defines the origin's geographic location

**coordinateId** An integral, unique identifier for the coordinate pair that defines the *Origin's* geographic location

**deliveryService** The 'xml\_id' of the *Delivery Service* to which the *Origin* belongs

**deliveryServiceId** An integral, unique identifier for the *Delivery Service* to which the *Origin* belongs

**fqdn** The FQDN of the *Origin*

**id** An integral, unique identifier for this *Origin*

**ip6Address** The IPv6 address of the *Origin*

**ipAddress** The IPv4 address of the *Origin*

**isPrimary** A boolean value which, when true specifies this *Origin* as the 'primary' *Origin* served by deliveryService

**lastUpdated** The date and time at which this *Origin* was last modified

**name** The name of the *Origin*

**port** The TCP port on which the *Origin* listens

**profile** The *Name* of the *Profile* used by this *Origin*

**profileId** The *ID* the *Profile* used by this *Origin*

**protocol** The protocol used by this origin - will be one of 'http' or 'https'

**tenant** The name of the *Tenant* that owns this *Origin*

**tenantId** An integral, unique identifier for the *Tenant* that owns this *Origin*

#### 411: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Zx7j0a7UAQxRtDenYodvGQSo0oPj4m0yY0AIEUpbdelmYMiNdPYtW82BCmMesFXkmP74nV4HbTUyDHVMu.
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 15:40:53 GMT
Content-Length: 420

{ "alerts": [
  {
    "text": "origin was updated.",
    "level": "success"
  }
],
"response": {
  "cachegroup": null,
  "cachegroupId": null,
  "coordinate": null,
  "coordinateId": null,
  "deliveryService": null,
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "id": 2,
  "ip6Address": null,
  "ipAddress": null,
  "isPrimary": true,
```

(continues on next page)

(continued from previous page)

```
"lastUpdated": "2018-12-11 15:40:53+00",
"name": "example",
"port": 443,
"profile": null,
"profileId": null,
"protocol": "https",
"tenant": null,
"tenantId": 1
}}
```

## DELETE

Deletes an *Origin* definition.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 190: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the <i>Origin</i> definition being deleted

### 412: Request Example

```
DELETE /api/3.0/origins?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 413: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: fLaY4/
→nh0yR38xq5weBKYg02+aQV6Z1ZroQ9UqUCHLMMrH1NMyhOHx+EphPq7JxkjmGY04WCt6VvDyjGWcgfQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 17:04:14 GMT
Content-Length: 61

{ "alerts": [
  {
    "text": "origin was deleted.",
    "level": "success"
  }
]}

```

## osversions

### See also:

*Generate ISO*

## GET

Gets all available OS (Operating System) versions for ISO generation, as well as the name of the directory where the “kickstarter” files are found.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

No parameters available.

## Response Structure

This endpoint has no constant keys in its response. Instead, each key in the response object is the name of an OS, and the value is a string that names the directory where the ISO source can be found. These directories sit under `/var/www/files/` on the Traffic Ops host machine by default, or at the location defined by the `kickstart.files.location` *Parameter* of the Traffic Ops server's *Profile*, if it is defined.

### 414: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳RxbRY2DZ+1Y0dTzzUETEZ3wtLBiD2BwXMVuaZjhe4a4cwgcZKRBWxZ6Qy5YYujFe1+UBiTG4sML/
    ↳Amn27F4AVg==
X-Server-Name: traffic_ops_golang/
Date: Fri, 30 Nov 2018 19:14:36 GMT
Content-Length: 38

{ "response": {
  "CentOS 7.2": "centos72"
}}
```

## Configuration File

The data returned from the endpoint comes directly from a configuration file. By default, the file is located at `/var/www/files/osversions.json`. The **directory** of the file can be changed by creating a specific *Parameter* named `kickstart.files.location` in configuration file `mkisofs`.

The format of the file is a JSON object as described in *Response Structure*.

### 415: Example osversions.json file

```
{
  "CentOS 7.2": "centos72"
}
```

The legacy Perl Traffic Ops used a Perl configuration file located by default at `/var/www/files/osversions.cfg`. A Perl script is provided to convert the legacy configuration file to the new JSON format. The script is located within the Traffic Control repository at

traffic\_ops/app/bin/osversions-convert.pl.

416: Example usage of conversion script

```
./osversions-convert.pl < /var/www/files/osversions.cfg > /var/www/  
→files/osversions.json
```

### parameterprofile

#### POST

Create one or more *Parameter/Profile* assignments.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

**paramId** The *ID* of the *Parameter* to be assigned to the *Profiles* identified within the profileIds array

**profileIds** An array of *Profile IDs* to which the *Parameter* identified by paramId shall be assigned

**replace** An optional boolean (default: false) which, if true, will cause any conflicting *Profile/Parameter* assignments to be overridden.

417: Request Example

```
POST /api/3.0/parameterprofile HTTP/1.1  
Host: trafficops.infra.ciab.test  
User-Agent: curl/7.47.0  
Accept: */*  
Cookie: mojolicious=...  
Content-Length: 38  
Content-Type: application/json  
  
{  
  "paramId": 4,  
  "profileIds": [18]  
}
```



## Response Structure

**paramId** The *ID* of the *Parameter* which has been assigned to the *Profiles* identified within the `profileIds` array

**profileIds** An array of *Profile IDs* to which the *Parameter* identified by `paramId` has been assigned

**replace** An optional boolean (default: false) which, if true, caused any conflicting *Profile/Parameter* assignments to be overridden.

### 418: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳N2ahnhEnfZ0UqnjylN6Vu3HaOZk340YuiuyiqkhTbk0pENp+kwBPYu4Z/
    ↳sqBAloCfXSQaWlJzaeXw4uOD5heWw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:18:23 GMT
Content-Length: 147

{ "alerts": [
  {
    "text": "2 parameters were assigned to the 18 profile",
    "level": "success"
  }
],
"response": {
  "profileId": 18,
  "paramIds": [
    2,
    3
  ],
  "replace": false
}}
```

## parameters

### GET

Gets all *Parameters* configured in Traffic Ops

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 191: Request Query Parameters

Name	Re- quired	Description
con- fig- File	no	Filter <i>Parameters</i> by <i>Config File</i>
id	no	Filters <i>Parameters</i> by <i>ID</i>
name	no	Filter <i>Parameters</i> by <i>Name</i>
value	no	Filter <i>Parameters</i> by <i>Value</i>
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

New in version ATCv6: The value query parameter was added to all API versions in ATC version 6.0.

### 419: Request Example

```
GET /api/3.0/parameters?configFile=records.config&name=location HTTP/1.
↪1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name of the Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 420: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
→jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
→DbBwhMAOJoeqDImoDlrLnrvjQGO4AooA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 18:23:39 GMT
Content-Length: 212

{ "response": [
  {
    "configFile": "records.config",
    "id": 29,
    "lastUpdated": "2018-12-05 17:51:02+00",
    "name": "location",
    "profiles": [
      "ATS_EDGE_TIER_CACHE",
      "ATS_MID_TIER_CACHE"
    ],
    "secure": false,
    "value": "/etc/trafficserver/"
  }
]}
```

## POST

Creates one or more new *Parameters*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Array

## Request Structure

The request body may be in one of two formats, a single *Parameter* object or an array of *Parameter* objects. Each *Parameter* object shall have the following keys:

**configFile** The *Parameter*’s *Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter*’s *Value*

### 421: Request Example - Single Object Format

```
POST /api/3.0/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 84
Content-Type: application/json

{
  "name": "test",
  "value": "quest",
  "configFile": "records.config",
  "secure": false
}
```

### 422: Request Example - Array Format

```
POST /api/3.0/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 180
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
[{
  "name": "test",
  "value": "quest",
  "configFile": "records.config",
  "secure": false
},
{
  "name": "foo",
  "value": "bar",
  "configFile": "records.config",
  "secure": false
}]
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name of the Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 423: Response Example - Single Object Format

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→eQrl48zWids0kDpfCYmntYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
→Cfh2X9i3G+MQ36eH95gukJqOBOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:18:21 GMT
Content-Length: 212
```

(continues on next page)

(continued from previous page)

```
{ "alerts": [
  {
    "text": "param was created.",
    "level": "success"
  }
],
"response": {
  "configFile": "records.config",
  "id": 124,
  "lastUpdated": "2018-12-05 19:18:21+00",
  "name": "test",
  "profiles": null,
  "secure": false,
  "value": "quest"
}}
```

parameters/{{ID}}

PUT

Replaces a *Parameter*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 192: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Parameter</i> which will be deleted

**configFile** The *Parameter*’s *Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter*’s *Value*

424: Request Example

```
PUT /api/3.0/parameters/124 HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)

(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 81
Content-Type: application/json

{
  "name": "foo",
  "value": "bar",
  "configFile": "records.config",
  "secure": false
}
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name of the Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 425: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: DMxS2gKceFVKRtezON/
→vsnrC+zI8onASSHaGv5i3wwvUvyt9KEe72gxQd6ZgVcSq3K8ZpkH6g3UI/WtEfdp5vA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 20:21:07 GMT
Content-Length: 209

{ "alerts": [
```

(continues on next page)

(continued from previous page)

```
{
  "text": "param was updated.",
  "level": "success"
},
],
"response": {
  "configFile": "records.config",
  "id": 125,
  "lastUpdated": "2018-12-05 20:21:07+00",
  "name": "foo",
  "profiles": null,
  "secure": false,
  "value": "bar"
}}
```

## DELETE

Deletes the specified *Parameter*. If, however, the *Parameter* is associated with one or more *Profiles*, deletion will fail.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response TType** undefined

## Request Structure

Table 193: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Parameter</i> which will be deleted



#### 426: Request Example

```
DELETE /api/3.0/parameters/124 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 427: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: hJjQq2Seg7sqWt+jKgp6gwRxUtoVU34PFoc9wEaweXdaIBTn/
    ↳BscoUuyw2/n+V8GZPqpeQcihZE50/0oQhdtHw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:20:30 GMT
Content-Length: 60

{ "alerts": [
  {
    "text": "param was deleted.",
    "level": "success"
  }
]}
```

## phys\_locations

### GET

Retrieves *Physical Locations*

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 194: Request Query Parameters

Name	Description
id	Filter by integral, unique identifier
re-gion	Filter by integral, unique identifier of containing <i>Region</i>
name	Filter by name
or-derby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

428: Request Example

```
GET /api/3.0/phys_locations?name=CDN_in_a_Box HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- address** The physical location’s street address
- city** The name of the city in which the physical location lies
- comments** Any and all human-readable comments
- email** The email address of the physical location’s poc
- id** An integral, unique identifier for the physical location
- lastUpdated** The date and time at which the physical location was last updated, in *Traffic Ops’s Custom Date/Time Format*
- name** The name of the physical location
- phone** A phone number where the the physical location’s poc might be reached
- poc** The name of a “point of contact” for the physical location
- region** The name of the region within which the physical location lies

**regionId** An integral, unique identifier for the region within which the physical location lies

**shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

#### 429: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳0g4b3W1AwXytCnBo8TReQQij2v9oHA17MG9KuwMig5V4sFcMM5qP8dgPsFTunFr00DPI20c7BpUbZsvJts
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 22:19:52 GMT
Content-Length: 275

{ "response": [
  {
    "address": "1600 Pennsylvania Avenue NW",
    "city": "Washington",
    "comments": "",
    "email": "",
    "id": 2,
    "lastUpdated": "2018-12-05 17:50:58+00",
    "name": "CDN_in_a_Box",
    "phone": "",
    "poc": "",
    "regionId": 1,
    "region": "Washington, D.C",
    "shortName": "ciab",
    "state": "DC",
    "zip": "20500"
  }
]}
```

### POST

Creates a new physical location

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

**address** The physical location’s street address

**city** The name of the city in which the physical location lies

**comments** An optional string for containing any and all human-readable comments

**email** An optional string containing email address of the physical location’s poc

**name** An optional name of the physical location

**phone** An optional string containing the phone number where the the physical location’s poc might be reached

**poc** The name of a “point of contact” for the physical location

**region** An optional string naming the region that contains this physical location<sup>1</sup>

**regionId** An integral, unique identifier for the region within which the physical location lies<sup>1</sup>

**shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

#### 430: Request Example

```
POST /api/3.0/phys_locations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 326
Content-Type: application/json
```

(continues on next page)

---

<sup>1</sup> The only “region” key that actually matters in the request body is **regionId**; **region** is not validated and has no effect - particularly not the effect of re-naming the region - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the region. Note that if **region** is not present in the request body it will be **null** in the response, but again further requests will show the true region name.

(continued from previous page)

```
{
  "address": "Buckingham Palace",
  "city": "London",
  "comments": "Buckingham Palace",
  "email": "steve.kingstone@royal.gsx.gov.uk",
  "name": "Great_Britain",
  "phone": "0-843-816-6276",
  "poc": "Her Majesty The Queen Elizabeth Alexandra Mary Windsor II",
  "regionId": 3,
  "shortName": "uk",
  "state": "Westminster",
  "zip": "SW1A 1AA"
}
```

## Response Structure

- address** The physical location's street address
- city** The name of the city in which the physical location lies
- comments** Any and all human-readable comments
- email** The email address of the physical location's poc
- id** An integral, unique identifier for the physical location
- lastUpdated** The date and time at which the physical location was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of the physical location
- phone** A phone number where the the physical location's poc might be reached
- poc** The name of a "point of contact" for the physical location
- region** The name of the region within which the physical location lies
- regionId** An integral, unique identifier for the region within which the physical location lies
- shortName** An abbreviation of the name
- state** An abbreviation of the name of the state or province within which this physical location lies
- zip** The zip code of the physical location

### 431: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: GZ/
    ↳BC+AgGpOQNfd9oiZy19jtsD8MP0deyi7PVdz+9YSiLYP44gmn5K+Xi1yS0l59yjHf70+C1loVQPS1IeP9
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 00:14:47 GMT
Content-Length: 443

{ "alerts": [
  {
    "text": "physLocation was created.",
    "level": "success"
  }
],
"response": {
  "address": "Buckingham Palace",
  "city": "London",
  "comments": "Buckingham Palace",
  "email": "steve.kingstone@royal.gsx.gov.uk",
  "id": 3,
  "lastUpdated": "2018-12-06 00:14:47+00",
  "name": "Great_Britain",
  "phone": "0-843-816-6276",
  "poc": "Her Majesty The Queen Elizabeth Alexandra Mary Windsor II",
  "regionId": 3,
  "region": null,
  "shortName": "uk",
  "state": "Westminster",
  "zip": "SW1A 1AA"
}}
```

`phys_locations/{ID}`

**PUT**

Updates a *Physical Location*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Array

## Request Structure

Table 195: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Physical Location</i> being modified

**address** The physical location’s street address

**city** The name of the city in which the physical location lies

**comments** An optional string for containing any and all human-readable comments

**email** An optional string containing email address of the physical location’s poc

**name** An optional name of the physical location

**phone** An optional string containing the phone number where the the physical location’s poc might be reached

**poc** The name of a “point of contact” for the physical location

**region** An optional string naming the region that contains this physical location<sup>1</sup>

**regionId** An integral, unique identifier for the region within which the physical location lies<sup>1</sup>

**shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

### 432: Request Structure

```
PUT /api/3.0/phys_locations/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 268
Content-Type: application/json

{
  "address": "1600 Pennsylvania Avenue NW",
  "city": "Washington",
  "comments": "The White House",
```

(continues on next page)

<sup>1</sup> The only “region” key that actually matters in the request body is regionId; region is not validated and has no effect - particularly not the effect of re-naming the region - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the region. Note that if region is not present in the request body it will be null in the response, but again further requests will show the true region name.

(continued from previous page)

```
"email": "the@white.house",
"name": "CDN_in_a_Box",
"phone": "1-202-456-1414",
"poc": "Donald J. Trump",
"regionId": 2,
"shortName": "ciab",
"state": "DC",
"zip": "20500"
}
```

## Response Structure

- address** The physical location's street address
- city** The name of the city in which the physical location lies
- comments** Any and all human-readable comments
- email** The email address of the physical location's poc
- id** An integral, unique identifier for the physical location
- lastUpdated** The date and time at which the physical location was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of the physical location
- phone** A phone number where the the physical location's poc might be reached
- poc** The name of a "point of contact" for the physical location
- region** The name of the region within which the physical location lies
- regionId** An integral, unique identifier for the region within which the physical location lies
- shortName** An abbreviation of the name
- state** An abbreviation of the name of the state or province within which this physical location lies
- zip** The zip code of the physical location

### 433: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)



(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→qnMe60qxjSU8H1njlh00HWNr20YnVlOCufqCTdMBcdC1322jk2ICFQsQQ3Xu0OR0WSb7h70HCfXqDC1/
→jA1xjA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 23:39:17 GMT
Content-Length: 385

{ "alerts": [
  {
    "text": "physLocation was updated.",
    "level": "success"
  }
],
"response": {
  "address": "1600 Pennsylvania Avenue NW",
  "city": "Washington",
  "comments": "The White House",
  "email": "the@white.house",
  "id": 2,
  "lastUpdated": "2018-12-05 23:39:17+00",
  "name": "CDN_in_a_Box",
  "phone": "1-202-456-1414",
  "poc": "Donald J. Trump",
  "regionId": 2,
  "region": null,
  "shortName": "ciab",
  "state": "DC",
  "zip": "20500"
}}
```

## DELETE

Deletes a *Physical Location*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 196: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Physical Location</i> being deleted

### 434: Request Example

```
DELETE /api/3.0/phys_locations/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 435: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: KeW/tEmICwpCGC8F0YMTqHdeR9J6W6Z3w/
→U+H0SbeCGyaEheCIhIsWlngT3dyfH1tiu8UyzaPB6QrJyXdybBkw==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 00:28:48 GMT
Content-Length: 67

{ "alerts": [
  {
    "text": "physLocation was deleted.",
    "level": "success"
  }
]}
```

## ping

Checks whether Traffic Ops is online.

### GET

**Auth. Required** No

**Response Type** undefined

### Request Structure

No parameters available.

436: Request Example

```
GET /api/3.0/ping HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

**ping** Returns an object containing only the "ping" property, which always has the value "pong".

437: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Whole-Content-Sha512: 0HqcLcYHCB4AFYGFzcAsP2h+PCmLYxk/
↳TqMajcy3fWCzY730Tv32k5trUaJLeSBbRx2FUi7z/sTAuzikdg0E4g==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 20:52:01 GMT
Content-Length: 40
Content-Type: application/x-gzip

{
```

(continues on next page)

(continued from previous page)

```
"ping": "pong"
}
```

### plugins

#### See also:

*Go Plugins*

### GET

Retrieves the list of configured enabled Traffic Ops plugins.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

438: Request Example

```
GET /api/3.0/plugins HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

**description** A short description of the plugin.

**name** The name of the plugin.

**version** A (hopefully) semantic version number describing the version of the plugin.

439: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Tue, 11 Dec 2018 20:51:48 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↪GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: n73jg9XR4V5Cwqq56Rf3wuIi99k3mM5u2NAjcZ/
    ↪gQBu8jvAFymDlnZqKeJ+wTll1vjIsHpXC0VXV7+5UGakLgA==
Transfer-Encoding: chunked

{ "response": [
  {
    "name": "helloworld",
    "version": "1.0.0",
    "description": "configuration plugin to run at startup"
  }
]}

```

## profileparameter

See also:

*profileparameters*.

## POST

Create one or more *Profile/Parameter* assignments.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**profileId** The *ID* of the *Profile* to which the *Parameters* identified within the *parameterIds* array will be assigned

**paramIds** An array of *Parameter IDs* which shall be assigned to the *Profile* identified by *profileId*

**replace** An optional boolean (default: false) which, if true, will cause any conflicting *Profile/Parameter* assignments to be overridden.

#### 440: Request Example

```
POST /api/3.0/profileparameter HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 38
Content-Type: application/json

{
  "profileId": 18,
  "paramIds": [2, 3]
}
```

### Response Structure

- profileId** The *ID* of the *Profile* to which the *Parameters* identified within the *parameterIds* array are assigned
- paramIds** An array of *Parameter IDs* which have been assigned to the *Profile* identified by *profileId*
- replace** An optional boolean (default: false) which, if true, indicates that any conflicting *Profile/Parameter* assignments have been overridden.

#### 441: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→N2ahnhEnfZ0UqnjylN6Vu3HaOZk340YuiuyiqkhTbk0pENp+kwBPYu4Z/
→sqBAloCfXSQaWlJzaeXw4uOD5heWw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:18:23 GMT
Content-Length: 147

{ "alerts": [
  {
    "text": "2 parameters were assigned to the 18 profile",
```

(continues on next page)

(continued from previous page)

```

        "level": "success"
    }
],
"response": {
    "profileId": 18,
    "paramIds": [
        2,
        3
    ],
    "replace": false
}}

```

## profileparameters

### GET

Retrieves all *Parameter/Profile* assignments.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 197: Request Query Parameters

Name	Re- quired	Description
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

## Response Structure

**lastUpdated** The date and time at which this *Profile/Parameter* association was last modified, in *Traffic Ops's Custom Date/Time Format*

**parameter** The *ID* of a *Parameter* assigned to profile

**profile** The *Name* of the *Profile* to which the *Parameter* identified by parameter is assigned

### 442: Response Structure

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳+bnMkRgdx4bJoGGlr3mZl539obj3aQAP8e65FAXgywdRAUfXZCFM6VNDn7wScXBmvF2SFXo9F+MhuSwrtI
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:09:13 GMT
Transfer-Encoding: chunked

{ "response": [
  {
    "lastUpdated": "2018-12-05 17:50:49+00",
    "profile": "GLOBAL",
    "parameter": 4
  },
  {
    "lastUpdated": "2018-12-05 17:50:49+00",
    "profile": "GLOBAL",
    "parameter": 5
  }
]}
```

---

**Note:** The response example for this endpoint has been truncated to only the first two elements of the resulting array, as the output was hundreds of lines long.

---



## POST

Associate a *Parameter* to a *Profile*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

This endpoint accepts two formats for the request payload:

**Single Object Format** For assigning a single *Parameter* to a single *Profile*

**Array Format** For making multiple assignments of *Parameters* to *Profiles* simultaneously

## Single Object Format

**parameterId** The *ID* of a *Parameter* to assign to some *Profile*

**profileId** The *ID* of the *Profile* to which the *Parameter* identified by parameterId will be assigned

#### 443: Request Example - Single Object Format

```
POST /api/3.0/profileparameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 36
Content-Type: application/json

{
  "profileId": 18,
  "parameterId": 1
}
```

### Array Format

**parameterId** The *ID* of a *Parameter* to assign to some *Profile*

**profileId** The *ID* of the *Profile* to which the *Parameter* identified by parameterId will be assigned

#### 444: Request Example - Array Format

```
POST /api/3.0/profileparameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 88
Content-Type: application/json

[
  {
    "profileId": 18,
    "parameterId": 2
  },
  {
    "profileId": 18,
    "parameterId": 3
  }
]
```

## Response Structure

**lastUpdated** The date and time at which the *Profile/Parameter* assignment was last modified, in *Traffic Ops's Custom Date/Time Format*

**parameter** *Name* of the *Parameter* which is assigned to profile

**parameterId** The *ID* of the assigned *Parameter*

**profile** *Name* of the *Profile* to which the *Parameter* is assigned

**profileId** The *ID* of the *Profile* to which the *Parameter* identified by parameterId is assigned

### 445: Response Example - Single Object Format

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳eDmIwlzX44fZdxLRPHMNa8aoGAK5fQv9Y70A2eeQHfEkliU4evwcsQ4WeHcH013/
    ↳wPTGlpYC0gwLo8LQqpUxWQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 13:50:11 GMT
Content-Length: 166

{ "alerts": [
  {
    "text": "profileParameter was created.",
    "level": "success"
  }
],
"response": {
  "lastUpdated": null,
  "profile": null,
  "profileId": 18,
  "parameter": null,
  "parameterId": 1
}}
```

profileparameters/{profileID}/{parameterID}

## DELETE

Deletes a *Profile/Parameter* association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 198: Request Path Parameters

Name	Description
profileID	The <i>ID</i> of the <i>Profile</i> from which a <i>Parameter</i> shall be removed
parameterID	The <i>ID</i> of the <i>Parameter</i> which shall be removed from the <i>Profile</i> identified by profileID

446: Request Example

```
DELETE /api/3.0/profileparameters/18/129 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

447: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳JQuBqHyT9MnNw09NSIDVQhkRtXdeAJc95W1pF2dwQeoBFmf0Y8knXm3/0/
    ↳rbJDEoUC7DhUQN1aoYIsqqmz4qQQ==
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

(continued from previous page)

Date: Mon, 10 Dec 2018 15:00:15 GMT

Content-Length: 71

```
{ "alerts": [
  {
    "text": "profileParameter was deleted.",
    "level": "success"
  }
]}
```

## profiles

GET

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 199: Request Query Parameters

Name	Re- quired	Description
cdn	no	Used to filter <i>Profiles</i> by the integral, unique identifier of the CDN to which they belong
id	no	Filters <i>Profiles</i> by <i>ID</i>
name	no	Filters <i>Profiles</i> by <i>Name</i>
param	no	Used to filter <i>Profiles</i> by the <i>ID</i> of a <i>Parameter</i> associated with them

#### 448: Request Example

```
GET /api/3.0/profiles?name=ATS_EDGE_TIER_CACHE HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

**cdn** The integral, unique identifier of the *CDN* to which this *Profile* belongs

**cdnName** The name of the *CDN* to which this *Profile* belongs

**description** The *Profile*'s *Description*

**id** The *Profile*'s *ID*

**lastUpdated** The date and time at which this *Profile* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** The *Profile*'s *Name*

**routingDisabled** The *Profile*'s *Routing Disabled* setting

**type** The *Profile*'s *Type*

#### 449: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: QEpKM/
    ↳DwHBRvue9K7XKrpwKFKhW6yCMQ2vSQgxE7dWFGJaqC4KOU092bsJU/
    ↳5fjI9qlB+1uMT2kz6mFb1Wzp/w==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 20:40:31 GMT
Content-Length: 220

{ "response": [
  {
    "id": 9,
    "lastUpdated": "2018-12-05 17:51:00+00",
```

(continues on next page)

(continued from previous page)

```

    "name": "ATS_EDGE_TIER_CACHE",
    "description": "Edge Cache - Apache Traffic Server",
    "cdnName": "CDN-in-a-Box",
    "cdn": 2,
    "routingDisabled": false,
    "type": "ATS_PROFILE"
  }
}]

```

## POST

Creates a new *Profile*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**cdn** The integral, unique identifier of the *CDN* to which this *Profile* shall belong

**description** The *Profile*’s *Description*

**name** The *Profile*’s *Name*

**routingDisabled** The *Profile*’s *Routing Disabled* setting

**type** The *Profile*’s *Type*

### 450: Request Example

```

POST /api/3.0/profiles HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 125
Content-Type: application/json

{
  "name": "test",
  "description": "A test profile for API examples",
  "cdn": 2,
  "type": "UNK_PROFILE",
  "routingDisabled": true
}

```

## Response Structure

- cdn** The integral, unique identifier of the *CDN* to which this *Profile* belongs
- cdnName** The name of the *CDN* to which this *Profile* belongs
- description** The *Profile's Description*
- id** The *Profile's ID*
- lastUpdated** The date and time at which this *Profile* was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The *Profile's Name*
- routingDisabled** The *Profile's Routing Disabled* setting
- type** The *Profile's Type*

### 451: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→UGV3PCnYBY0J3siICR0f9VVRNdUK1+9zsDDP6T9yt6t+AoHckHe6bvz0li9to/
→fGhC2zz5l9Nc1ro4taJUDD8g==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 21:24:49 GMT
Content-Length: 251

{ "alerts": [
  {
    "text": "profile was created.",
    "level": "success"
  }
],
"response": {
  "id": 16,
  "lastUpdated": "2018-12-07 21:24:49+00",
  "name": "test",
  "description": "A test profile for API examples",
  "cdnName": null,
  "cdn": 2,
  "routingDisabled": true,
```

(continues on next page)



(continued from previous page)

```
"type": "UNK_PROFILE"
}}
```

`profiles/{{ID}}`

**PUT**

Replaces the specified *Profile* with the one in the request payload

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 200: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> being modified

**cdn** The integral, unique identifier of the *CDN* to which this *Profile* will belong

**description** The *Profile*’s new *Description*

**name** The *Profile*’s new *Name*

**routingDisabled** The *Profile*’s new *Routing Disabled* setting

**type** The *Profile*’s new *Type*

**Warning:** Changing this will likely break something, be **VERY** careful when modifying this value

## 452: Request Example

```
PUT /api/3.0/profiles/16 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 125
Content-Type: application/json

{
```

(continues on next page)

(continued from previous page)

```
"name": "test",
"description": "A test profile for API examples",
"cdn": 2,
"type": "UNK_PROFILE",
"routingDisabled": true
}
```

## Response Structure

- cdn** The integral, unique identifier of the *CDN* to which this *Profile* belongs
- cdnName** The name of the *CDN* to which this *Profile* belongs
- description** The *Profile*'s *Description*
- id** The *Profile*'s *ID*
- lastUpdated** The date and time at which this *Profile* was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The *Profile*'s *Name*
- routingDisabled** The *Profile*'s *Routing Disabled* setting
- type** The *Profile*'s *Type*

### 453: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Pnf+G9G3/
→+edt4b8PVsyGZHsNzaFEgphaGSminjRlRmMpWtuLAA20WZDUo3nX0Q081c2GCuFuEh9uMF2Vjeppg==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 21:45:06 GMT
Content-Length: 251

{ "alerts": [
  {
    "text": "profile was updated.",
    "level": "success"
  }
],
```

(continues on next page)

(continued from previous page)

```
"response": {
  "id": 16,
  "lastUpdated": "2018-12-07 21:45:06+00",
  "name": "test",
  "description": "A test profile for API examples",
  "cdnName": null,
  "cdn": 2,
  "routingDisabled": true,
  "type": "UNK_PROFILE"
}}
```

## DELETE

Allows user to delete a *Profile*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 201: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> being deleted

454: Request Example

```
DELETE /api/3.0/profiles/16 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

455: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: HNmJkZaNW9yil08/
→3TnqZ5FllH6Rp+jgp3KI46FZdojLYcu+8jEhDL1lokoirdrHyU4R1c3hjCI0urN7PVvWDA==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 21:55:33 GMT
Content-Length: 62

{ "alerts": [
  {
    "text": "profile was deleted.",
    "level": "success"
  }
]}
```

`profiles/{{ID}}/export`

GET

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 202: Request Path Parameters

Parameter	Description
id	The <i>ID</i> of the <i>Profile</i> to be exported

456: Request Example

```
GET /api/3.0/profiles/3/export HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- profile** The exported *Profile*
  - cdn** The name of the *CDN* to which this *Profile* belongs
  - description** The *Profile's Description*
  - name** The *Profile's Name*
  - type** The *Profile's Type*
- parameters** An array of *Parameters* in use by this *Profile*
  - config\_file** The *Parameter's Config File*
  - name** *Name* of the *Parameter*
  - value** The *Parameter's Value*

### 457: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Disposition: attachment; filename="GLOBAL.json"
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳mzP7DVxFAGhICxqagwDyBDRea7oBZPMAx7NCDeOBVCRqlcCFFe7XL3JP58b80aaVOW/
    ↳2ZGfg/jpYF70cdDfzQA==
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 Sep 2019 20:14:42 GMT
Transfer-Encoding: gzip

{ "profile": {
  "name": "GLOBAL",
  "description": "Global Traffic Ops profile",
  "cdn": "ALL",
  "type": "UNK_PROFILE"
},
  "parameters": [
    {
      "config_file": "global",
      "name": "tm.instance_name",
      "value": "Traffic Ops CDN"
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "config_file": "global",
      "name": "tm.toolname",
      "value": "Traffic Ops"
    }
  ]
}
```

## profiles/{{ID}}/parameters

### GET

Retrieves all *Parameters* assigned to the *Profile*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 203: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> for which <i>Parameters</i> will be listed

### 458: Request Example

```

GET /api/3.0/parameters/profile/GLOBAL HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name* of the *Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

#### 459: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: NudgZXUNyKNpmSFf856KEjyy+Pin/
    ↳bFhG9NoRBDaxYbRKt2T5fF5Ze7sUNZfFI5n/ZZsgbx6Tsgtfd7oM6j+eg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 21:08:56 GMT
Content-Length: 542
```

```
{ "response": [
  {
    "configFile": "global",
    "id": 4,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.instance_name",
    "secure": false,
    "value": "Traffic Ops CDN"
  },
  {
    "configFile": "global",
    "id": 5,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.toolname",
    "secure": false,
    "value": "Traffic Ops"
  },
  {
    "configFile": "global",
    "id": 6,
    "lastUpdated": "2018-12-05 17:50:51+00",
    "name": "use_tenancy",
    "secure": false,
    "value": "1"
  },
  {
```

(continues on next page)

(continued from previous page)

```

    "configFile": "regex_revalidate.config",
    "id": 7,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "maxRevalDurationDays",
    "secure": false,
    "value": "90"
  }
}]

```

## POST

Associates *Parameters* to a *Profile*. If the *Parameter* does not exist, creates it and associates it to the *Profile*. If the *Parameter* already exists, associates it to the *Profile*. If the *Parameter* is already associated with the *Profile*, keep the association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 204: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> to which <i>Parameters</i> will be assigned

This endpoint accepts two formats for the request payload:

**Single Object Format** For assigning a single *Parameter* to a single *Profile*

**Parameter Array Format** For making multiple assignments of *Parameters* to *Profiles* simultaneously

**Warning:** Most API endpoints dealing with *Parameters* treat *Secure* as a boolean value, whereas this endpoint takes the legacy approach of treating it as an integer. Be careful when passing data back and forth, as boolean values will **not** be accepted by this endpoint!



## Single Parameter Format

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 460: Response Example - Single Parameter Format

```
POST /api/3.0/profiles/18/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 99
Content-Type: application/json

{
  "name": "test",
  "configFile": "quest",
  "value": "A test parameter for API examples",
  "secure": 0
}
```

## Parameter Array Format

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 461: Request Example - Parameter Array Format

```
POST /api/3.0/profiles/18/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 212
Content-Type: application/json

[ {
  "name": "test",
  "configFile": "quest",
```

(continues on next page)

(continued from previous page)

```

    "value": "A test parameter for API examples",
    "secure": 0
  },
  {
    "name": "foo",
    "configFile": "bar",
    "value": "Another test parameter for API examples",
    "secure": 0
  }
]
```

## Response Structure

**parameters** An array of objects representing the *Parameters* which have been assigned

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

**profileId** The *ID* of the *Profile* to which the *Parameter(s)* have been assigned

**profileName** *Name* of the *Profile* to which the *Parameter(s)* have been assigned

### 462: Response Example - Single Parameter Format

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: R2QUyCaNvKvVv/PNVNmEd/ma5h/iP1fMJlqhv+x2jE/
→zNpHJ1KVXt6s3btB8nnHv6IDF/gt5kIzQ0mbW5U8bpg==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 14:45:28 GMT
Content-Length: 253

{ "alerts": [
  {
    "text": "Assign parameters successfully to profile test",
```

(continues on next page)

(continued from previous page)

```

        "level": "success"
    }
],
"response": {
    "parameters": [
        {
            "configFile": "quest",
            "name": "test",
            "secure": 0,
            "value": "A test parameter for API examples",
            "id": 126
        }
    ],
    "profileId": 18,
    "profileName": "test"
}}

```

---

**Note:** The format of the request does not affect the format of the response. `parameters` will be an array either way.

---

## profiles/import

### POST

Imports a *Profile* that was exported via *profiles/{ID}/export*

---

**Note:** On import of the *Profile Parameters* if a *Parameter* already exists with the same *Name*, *Config File* and *Value* it will link that to the *Profile* instead of creating it.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

- profile** The exported *Profile*
  - cdn** The name of the *CDN* to which this *Profile* belongs
  - description** The *Profile's Description*
  - name** The *Profile's Name*
  - type** The *Profile's Type*
- parameters** An array of *Parameters* in use by this *Profile*
  - config\_file** The *Parameter's Config File*
  - name** *Name* of the *Parameter*
  - value** The *Parameter's Value*

### 463: Request Example

```
POST /api/3.0/profiles/import HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Type: application/json

{ "profile": {
  "name": "GLOBAL",
  "description": "Global Traffic Ops profile",
  "cdn": "ALL",
  "type": "UNK_PROFILE"
},
"parameters": [
  {
    "config_file": "global",
    "name": "tm.instance_name",
    "value": "Traffic Ops CDN"
  },
  {
    "config_file": "global",
    "name": "tm.toolname",
    "value": "Traffic Ops"
  }
]
}
```

## Response Structure

**cdn** The name of the *CDN* to which this *Profile* belongs

**description** The *Profile's Description*

**name** The *Profile's Name*

**type** The *Profile's Type*

**id** The *Profile's ID*

### 464: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳mzP7DVxFAGhICxqagwDyBDRea7oBZPMAx7NCDeOBVCRqlcCFFe7XL3JP58b80aaVOW/
    ↳2ZGfg/jpYF70cdDfzQA==
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 Sep 2019 20:14:42 GMT
Transfer-Encoding: gzip

{ "alerts": [
  {
    "level": "success",
    "text": "Profile imported [ Global ] with 2 new and 0 existing
    ↳parameters"
  }
],
"response": {
  "cdn": "ALL",
  "name": "Global",
  "id": 18,
  "type": "UNK_PROFILE",
  "description": "Global Traffic Ops profile"
}}
```

`profiles/name/{{name}}/copy/{{copy}}`

POST

Copy *Profile* to a new *Profile*. The new *Profile*'s *Name* must not already exist.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 205: Request Path Parameters

Name	Description
name	The <i>Name</i> of the new <i>Profile</i>
copy	The <i>Name</i> of <i>Profile</i> from which the copy will be made

465: Request Example

```
POST /api/3.0/profiles/name/GLOBAL_copy/copy/GLOBAL HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**description** The new *Profile*'s *Description*

**id** The *ID* of the new *Profile*

**idCopyFrom** The *ID* of the *Profile* from which the copy was made

**name** The *Name* of the new *Profile*

**profileCopyFrom** The *Name* of the *Profile* from which the copy was made

466: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Fri, 07 Dec 2018 22:03:54 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:↵
→r6V9viEZui1WCns0AUGEx1MtxjjXiU8SZV0tSQjeq7ZJDL15s8fMmjJdR/
→HRWduHn7Ax6GzYhoKwnIjMyc7ZWg==
Content-Length: 252

{ "alerts": [
  {
    "level": "success",
    "text": "Created new profile [ GLOBAL_copy ] from existing↵
→profile [ GLOBAL ]"
  }
],
"response": {
  "idCopyFrom": 1,
  "name": "GLOBAL_copy",
  "profileCopyFrom": "GLOBAL",
  "id": 17,
  "description": "Global Traffic Ops profile, DO NOT DELETE"
}}
```

`profiles/name/{{name}}/parameters`

**GET**

Retrieves all *Parameters* associated with a given *Profile*

**Auth. Required** Yes

**Roles Required** None

**Response Type** None

## Request Structure

Table 206: Request Path Parameters

Name	Description
name	The <i>Name</i> of the <i>Profile</i> for which <i>Parameters</i> will be listed

467: Request Example

```
GET /api/3.0/profiles/name/GLOBAL/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name* of the *Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

468: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: NudgZXUNyKNpmSFf856KEjyy+Pin/
→bFhG9NoRBDaxYbRKt2T5fF5Ze7sUNZfFI5n/ZZsgbx6Tsgtfd7oM6j+eg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 21:52:08 GMT
Content-Length: 542
```

(continues on next page)



(continued from previous page)

```
{ "response": [
  {
    "configFile": "global",
    "id": 4,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.instance_name",
    "secure": false,
    "value": "Traffic Ops CDN"
  },
  {
    "configFile": "global",
    "id": 5,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.toolname",
    "secure": false,
    "value": "Traffic Ops"
  },
  {
    "configFile": "global",
    "id": 6,
    "lastUpdated": "2018-12-05 17:50:51+00",
    "name": "use_tenancy",
    "secure": false,
    "value": "1"
  },
  {
    "configFile": "regex_revalidate.config",
    "id": 7,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "maxRevalDurationDays",
    "secure": false,
    "value": "90"
  }
]}
}
```

## POST

Associates *Parameters* to a *Profile*. If the *Parameter* does not exist, creates it and associates it to the *Profile*. If the *Parameter* already exists, associates it to the *Profile*. If the *Parameter* is already associated with the *Profile*, keep the association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 207: Request Path Parameters

Name	Description
name	The <i>Name</i> of the <i>Profile</i> to which <i>Parameters</i> will be assigned

This endpoint accepts two formats for the request payload:

**Single Parameter Format** Specify a single parameter to assign to the specified profile

**Parameter Array Format** Specify multiple parameters to assign to the specified profile

**Warning:** Most API endpoints dealing with parameters treat `secure` as a boolean value, whereas this endpoint takes the legacy approach of treating it as an integer. Be careful when passing data back and forth, as boolean values will **not** be accepted by this endpoint!

## Single Parameter Format

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

469: Request Example - Single Parameter Format

```
POST /api/3.0/profiles/name/test/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 99
Content-Type: application/json

{
  "name": "test",
  "configFile": "quest",
  "value": "A test parameter for API examples",
  "secure": 0
}
```

## Parameter Array Format

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

470: Request Example - Parameter Array Format

```
POST /api/3.0/profiles/name/test/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 212
Content-Type: application/json

[{"name": "test",
  "configFile": "quest",
  "value": "A test parameter for API examples",
  "secure": 0},
 {"name": "foo",
  "configFile": "bar",
  "value": "Another test parameter for API examples",
  "secure": 0}]
```

## Response Structure

**parameters** An array of objects representing the *Parameters* which have been assigned

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

**profileId** The *ID* of the *Profile* to which the *Parameter(s)* have been assigned

**profileName** *Name* of the *Profile* to which the *Parameter(s)* have been assigned

#### 471: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: HQWq0koYHjvcxheWPrH0b0oZnUC+qLG1L040jtsLLnZYVUIu/
    ↳qgJrzvziPnKq3FEHUWaZrnDCZM/iZD8AXOKBw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 14:20:22 GMT
Content-Length: 357

{ "alerts": [
  {
    "text": "Assign parameters successfully to profile test",
    "level": "success"
  }
],
"response": {
  "parameters": [
    {
      "configFile": "quest",
      "name": "test",
      "secure": 0,
      "value": "A test parameter for API examples",
      "id": 126
    },
    {
      "configFile": "bar",
      "name": "foo",
      "secure": 0,
      "value": "Another test parameter for API examples",
      "id": 129
    }
  ],
  "profileId": 18,
  "profileName": "test"
}}
```

**Note:** The format of the request does not affect the format of the response. parameters will

be an array either way.

## regions

### GET

Retrieves information about *Regions*

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

Table 208: Request Query Parameters

Name	Required	Description
division	no	Filter <i>Regions</i> by the integral, unique identifier of the <i>Division</i> which contains them
id	no	Filter <i>Regions</i> by integral, unique identifier
name	no	Filter <i>Regions</i> by name
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 472: Request Example

```
GET /api/3.0/regions?division=1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- divisionName** The name of the division which contains this region
- divisionId** The integral, unique identifier of the division which contains this region
- id** An integral, unique identifier for this region
- lastUpdated** The date and time at which this region was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The region name

### 473: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳nSYbR+fRXaxhYl7dWgf0Udo2AsiXEnwvED1CPbk7ZNVK03I3TOhtmCQx9ABnJJ6xKYnl6EKMeopVTK0n
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 01:58:38 GMT
Content-Length: 117

{ "response": [
  {
    "divisionName": "Quebec",
    "division": 1,
    "id": 2,
    "lastUpdated": "2018-12-05 17:50:58+00",
    "name": "Montreal"
  }
]}
```

## POST

Creates a new region

- Auth. Required** Yes
- Roles Required** “admin” or “operations”
- Response Type** Object

## Request Structure

**division** The integral, unique identifier of the division which shall contain the new region<sup>1</sup>

**divisionName** The name of the division which shall contain the new region<sup>1</sup>

**name** The name of the region

### 474: Request Example

```
POST /api/3.0/regions HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 65
Content-Type: application/json

{
  "name": "Manchester",
  "division": "4",
  "divisionName": "England"
}
```

## Response Structure

**divisionName** The name of the division which contains this region

**divisionId** The integral, unique identifier of the division which contains this region

**id** An integral, unique identifier for this region

**lastUpdated** The date and time at which this region was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** The region name

### 475: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST, GET, OPTIONS, PUT, DELETE
```

(continues on next page)

<sup>1</sup> The only “division” key that actually matters in the request body is `division`; `divisionName` is not validated and has no effect - particularly not the effect of re-naming the division - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the division. Note that if `divisionName` is not present in the request body it will be `null` in the response, but again further requests will show the true division name (provided it has been assigned to a division).

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→ezxk+iP7o7KE7zpWmGc0j8nz5k+1wAzY0HiNiA2xswTQrt+N+6CgQqUV2r9G1HAsPNr0HF2PhYs/
→Xr7DrYOW0A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 02:14:45 GMT
Content-Length: 178

{ "alerts": [
  {
    "text": "region was created.",
    "level": "success"
  }
],
"response": {
  "divisionName": "England",
  "division": 3,
  "id": 5,
  "lastUpdated": "2018-12-06 02:14:45+00",
  "name": "Manchester"
}}
```

## DELETE

Deletes a region. If no query parameter is specified, a 400 Bad Request response is returned.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 209: Request Query Parameters

Name	Required	Description
id	no	Delete <i>Region</i> by integral, unique identifier
name	no	Delete <i>Region</i> by name



#### 476: Request Example

```
DELETE /api/3.0/regions?name=Manchester HTTP/1.1
User-Agent: curl/7.29.0
Host: trafficops.infra.ciab.test
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 477: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 07 Feb 2020 13:56:24
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳yNqXKcoiohEbJrEkH8LD1zifh87dIusuvUqgQnYueyKqCXkfd5bQvQ00hQ2AAdSZa/
    ↳oe2SAqMjojGsUlxHtIQw==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Feb 2020 12:56:24 GMT
Content-Length: 62

{
  "alerts": [
    {
      "text": "region was deleted.",
      "level": "success"
    }
  ]
}
```

`regions/{{ID}}`

**PUT**

Updates a *Region*.

**Auth. Required** Yes

**Role(s) Required** “admin” or “operator”

**Response Type** Object

## Request Structure

Table 210: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the region to update

**division** The new integral, unique identifier of the division which shall contain the region<sup>1</sup>

**divisionName** The new name of the division which shall contain the region<sup>1</sup>

**name** The new name of the region

478: Request Example

```
PUT /api/3.0/regions/5 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 60
Content-Type: application/json

{
  "name": "Leeds",
  "division": 3,
  "divisionName": "England"
}
```

---

<sup>1</sup> The only “division” key that actually matters in the request body is `division`; `divisionName` is not validated and has no effect - particularly not the effect of re-naming the division - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the division. Note that if `divisionName` is not present in the request body it will be `null` in the response, but again further requests will show the true division name (provided it has been assigned to a division).

## Response Structure

- divisionName** The name of the division which contains this region
- divisionId** The integral, unique identifier of the division which contains this region
- id** An integral, unique identifier for this region
- lastUpdated** The date and time at which this region was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The region name

### 479: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳7SVj4q7dtSTNQEJlBApEwlad28WBVFnpdHaatoIpNfeLltfcpcdVTcOKB4JXQv7rlSD2p/
    ↳TxBQC6EXpxwYTnKQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 02:23:40 GMT
Content-Length: 173

{ "alerts": [
  {
    "text": "region was updated.",
    "level": "success"
  }
],
"response": {
  "divisionName": "England",
  "division": 3,
  "id": 5,
  "lastUpdated": "2018-12-06 02:23:40+00",
  "name": "Leeds"
}}
```

## roles

### GET

Retrieves all user *Roles*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

Table 211: Request Query Parameters

Name	Re- quired	Description
id	no	Return only the <i>Role</i> identified by this integral, unique identifier
name	no	Return only the <i>Role</i> with this name
privLevel	no	Return only those <i>Roles</i> that have this privilege level
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <b>response</b> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <b>limit</b>
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <b>limit</b> long and the first page is 1. If <b>offset</b> was defined, this query parameter has no effect. <b>limit</b> must be defined to make use of <b>page</b> .

#### 480: Request Example

```
GET /api/3.0/roles?name=admin HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

**capabilities** An array of the names of the Capabilities given to this *Role*

**description** A description of the *Role*

**id** The integral, unique identifier for this *Role*

**name** The name of the *Role*

**privLevel** An integer that allows for comparison between *Roles*

#### 481: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳TEDXlQqWMSnJbL10JtFdbw0nqciNpjc4bd6m7iAB8aymakWeF+ghs1k5LayjdzHcjeDE8UNF/
    ↳HXSxOFvoLFEuA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 04 Sep 2019 17:15:36 GMT
Content-Length: 120

{ "response": [
  {
    "id": 4,
    "name": "admin",
    "description": "super-user",
    "privLevel": 30,
    "capabilities": [
      "all-write",
      "all-read"
    ]
  }
]
```

(continues on next page)

(continued from previous page)

```
}
[]}
```

## POST

Creates a new *Role*.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

## Request Structure

**capabilities** An optional array of capability names that will be granted to the new *Role*

**description** A helpful description of the *Role*’s purpose.

**name** The name of the new *Role*

**privLevel** The privilege level of the new *Role*<sup>1</sup>

### 482: Request Example

```
POST /api/3.0/roles HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "name": "test",
  "description": "quest",
  "privLevel": 30
}
```

<sup>1</sup> **privLevel** cannot exceed the privilege level of the requesting user. Which, of course, must be the privilege level of “admin”. Basically, this means that there can never exist a *Role* with a higher privilege level than “admin”.

## Response Structure

**capabilities** An array of the names of the Capabilities given to this *Role*

---

**Tip:** This can be null *or* empty, depending on whether it was present in the request body, or merely empty. Obviously, it can also be a populated array.

---

**description** A description of the *Role*

**id** The integral, unique identifier for this *Role*

**name** The name of the *Role*

**privLevel** An integer that allows for comparison between *Roles*

### 483: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: gzfc7m/
    ↳in5vVsVP+Y9h6JJfDhgpXKn9VAzoiPENhKbQfP8Q6jug08Rt2AK/
    ↳3Nz1cx5zZ8P9IjVxDdIg7mlC8bw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 04 Sep 2019 17:44:42 GMT
Content-Length: 150

{ "alerts": [{
  "text": "role was created.",
  "level": "success"
}],
"response": {
  "id": 5,
  "name": "test",
  "description": "quest",
  "privLevel": 30,
  "capabilities": null
}}
```

## PUT

Replaces an existing *Role* with one provided by the request.

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type**

## Request Structure

Table 212: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the <i>Role</i> to be replaced

**capabilities** An optional array of capability names that will be granted to the new *Role*

**Warning:** When not present, the affected *Role*’s Capabilities will be unchanged - *not* removed, unlike when the array is empty.

**description** A helpful description of the *Role*’s purpose.

**name** The new name of the *Role*

**privLevel** The new privilege level of the new *Role*<sup>Page 810, 1</sup>

### 484: Request Example

```
PUT /api/3.0/roles?id=5 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "name": "test",
  "privLevel": 29,
  "description": "quest"
}
```



## Response Structure

**capabilities** An array of the names of the Capabilities given to this *Role*

**Tip:** This can be null *or* empty, depending on whether it was present in the request body, or merely empty. Obviously, it can also be a populated array.

**Warning:** If no **capabilities** array was given in the request, this will *always* be null, even if the *Role* has Capabilities that would have gone unchanged.

**description** A description of the *Role*

**id** The integral, unique identifier for this *Role*

**name** The name of the *Role*

**privLevel** An integer that allows for comparison between *Roles*

### 485: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳mlHQenE1Q3gjrIK21C2hfueQOaTCpdYefboN0A9vYPUIwTiaF5ZaAMPQBdfGyiAhgHRxowITs3bR7s1L+
X-Server-Name: traffic_ops_golang/
Date: Thu, 05 Sep 2019 12:56:46 GMT
Content-Length: 150

{
  "alerts": [
    {
      "text": "role was updated.",
      "level": "success"
    }
  ],
  "response": {
    "id": 5,
    "name": "test",
```

(continues on next page)

(continued from previous page)

```

    "description": "quest",
    "privLevel": 29,
    "capabilities": null
  }
}

```

## DELETE

Deletes a *Role*

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** undefined

## Request Structure

Table 213: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the <i>Role</i> to be replaced

### 486: Request Example

```

DELETE /api/3.0/roles?id=5 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...

```

## Response Structure

### 487: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly

```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: 10jeFZihtbvAus/XyHAW8rhgS9JBD+X/
→ezCp1iExYkEcHxN4gjr1L6x8zDFXORueBSlFldgtbWKT7QsmwCHUWA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 05 Sep 2019 13:02:06 GMT
Content-Length: 59

{ "alerts": [{
  "text": "role was deleted.",
  "level": "success"
}]}
```

## server\_capabilities

GET

Retrieves *Server Capabilities*.

**Auth. Required** Yes

**Roles Required** “read-only”

**Response Type** Array

## Request Structure

Table 214: Request Query Parameters

Name	Required	Description
name	no	Return the <i>Server Capability</i> with this name

#### 488: Request Structure

```
GET /api/3.0/server_capabilities?name=RAM HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

**name** The name of this *Server Capability*

**lastUpdated** The date and time at which this *Server Capability* was last updated, in ISO-like format

#### 489: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpGT3YRyKJ/
    ↳p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 21:36:13 GMT
Content-Length: 68

{
  "response": [
    {
      "name": "RAM",
      "lastUpdated": "2019-10-07 20:38:24+00"
    }
  ]
}
```

## POST

Create a new *Server Capability*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**name** The name of the *Server Capability*

490: Request Example

```
POST /api/3.0/server_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 15
Content-Type: application/json

{
  "name": "RAM"
}
```

## Response Structure

**name** The name of this *Server Capability*

**lastUpdated** The date and time at which this *Server Capability* was last updated, in ISO-like format

491: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ysdopC//
→JQI79BRUa61s6M2HzHxYHpo5Rdcuau0oqCYxiV0oUhNZfOVydVkv8zDN2qA374XKnym4kWj3VzQIXg==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:10:00 GMT
Content-Length: 137

{
  "alerts": [
    {
      "text": "server capability was created.",
      "level": "success"
    }
  ],
  "response": {
    "name": "RAM",
    "lastUpdated": "2019-10-07 22:10:00+00"
  }
}
```

## DELETE

Deletes a specific *Server Capability*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 215: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Server Capability</i> to be deleted

#### 492: Request Example

```
DELETE /api/3.0/server_capabilities?name=RAM HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

#### 493: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 8zCAATbCzcqiqigGVBy7WF1duDuXu1Wg2DBe9yfqTw/
    ↳c+yhE2eUk73hFTA/Oqt0kocaN7+1GkbFdPkQPvbnRaA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 20:44:40 GMT
Content-Length: 72

{
  "alerts": [
    {
      "text": "server capability was deleted.",
      "level": "success"
    }
  ]
}
```

### server\_server\_capabilities

#### GET

Gets all associations of *Server Capabilities* to *cache servers*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 216: Request Query Parameters

Name	Re- quired	Description
serverId	no	Filter <i>Server Capability</i> associations by the integral, unique identifier of the server to which they are assigned
server-Host-Name	no	Filter <i>Server Capability</i> associations by the host name of the server to which they are assigned
server-Ca-pabil-ity	no	Filter <i>Server Capability</i> associations by <i>Server Capability</i> name
or-derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 494: Request Example

```
GET /api/3.0/server_server_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**serverHostName** The server’s host name

**serverId** The server’s integral, unique identifier

**lastUpdated** The date and time at which this association between the server and the *Server Capability* was last updated, in *Traffic Ops’s Custom Date/Time Format*

**serverCapability** The *Server Capability*’s name



#### 495: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
    ↳jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
    ↳DbBwhMAOJoeqDIImoDlrLnrVjQG04AooA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 150

{
  "response": [
    {
      "lastUpdated": "2019-10-07 22:05:31+00",
      "serverHostName": "atlanta-org-1",
      "serverId": 260,
      "serverCapability": "ram"
    },
    {
      "lastUpdated": "2019-10-07 22:05:31+00",
      "serverHostName": "atlanta-org-2",
      "serverId": 261,
      "serverCapability": "disk"
    }
  ]
}
```

## POST

Associates a *Server Capability* to a server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**serverId** The integral, unique identifier of a server to be associated with a *Server Capability*

**serverCapability** The *Server Capability*'s name to associate

---

**Note:** The server referenced must be either an *Edge-tier* or *Mid-tier cache server*.

---

### 496: Request Example

```
POST /api/3.0/server_server_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 84
Content-Type: application/json

{
  "serverId": 1,
  "serverCapability": "disk"
}
```

## Response Structure

**serverId** The integral, unique identifier of the newly associated server

**lastUpdated** The date and time at which this association between the server and the *Server Capability* was last updated, in *Traffic Ops's Custom Date/Time Format*

**serverCapability** The *Server Capability*'s name

### 497: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
  ↳Cfh2X9i3G+MQ36eH95gukJqOBOGbQ==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 157

{
  "alerts": [
    {
      "text": "server server_capability was created.",
      "level": "success"
    }
  ],
  "response": {
    "lastUpdated": "2019-10-07 22:15:11+00",
    "serverId": 1,
    "serverCapability": "disk"
  }
}
```

## DELETE

Disassociate a server from a *Server Capability*.

---

**Note:** If the serverCapability is a *Server Capability* required by a *Delivery Service* that to which the server is assigned the DELETE will be blocked until either the server is unassigned from the *Delivery Service* or the *Server Capability* is no longer required by the *Delivery Service*.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 217: Request Query Parameters

Name	Re-quired	Description
serverId	yes	The integral, unique identifier of the server to disassociate
serverCapabil-ity	yes	term: <i>Server Capability</i> name to disassociate from given server

#### 498: Request Example

```
DELETE /api/3.0/server_server_capabilities?serverId=1&
→serverCapability=disk HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 499: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
→jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
→DbBwhMAOJoeqDImoDlrLnrVjQG04AooA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 96

{
  "alerts": [
    {
      "text": "server server_capability was deleted.",
      "level": "success"
    }
  ]
}
```

## servercheck

### See also:

*Check Extensions*

### GET

Fetches identifying and meta information as well as “check” values regarding all servers that have a *Type* with a name beginning with “EDGE” or “MID” (ostensibly this is equivalent to all *cache servers*).

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

### Request Structure

Table 218: Request Query Parameters

Name	Required	Description
id	no	Return only <i>cache servers</i> with this integral, unique identifier (id)
hostName	no	Return only <i>cache servers</i> with this host_name

500: Request Example with hostName query param

```
GET /api/4.0/servercheck?hostName=edge HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### 501: Request Example with id query param

```
GET /api/4.0/servercheck?id=12 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**adminState** The name of the server’s *Status* - called “adminState” for legacy reasons

**cacheGroup** The name of the *Cache Group* to which the server belongs

**checks** An optionally present map of the names of “checks” to their values. Only numeric and boolean checks are represented, and boolean checks are represented as integers with 0 meaning “false” and 1 meaning “true”. Will not appear if the server in question has no valued “checks”.

**hostName** The (short) hostname of the server

**id** The server’s integral, unique identifier

**profile** The name of the *Profile* used by the server

**revalPending** A boolean that indicates whether or not the server has pending revalidations

**type** The name of the server’s *Type*

**updPending** A boolean that indicates whether or not the server has pending updates

### 502: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Thu, 18 Feb 2021 20:00:19
    ↳GMT; Max-Age=3600; HttpOnly
X-Server-Name: traffic_ops_golang/
Date: Thu, 18 Feb 2021 19:00:19 GMT
Content-Length: 352
```

(continues on next page)

(continued from previous page)

```
{ "response": [
  {
    "adminState": "REPORTED",
    "cacheGroup": "CDN_in_a_Box_Edge",
    "id": 12,
    "hostname": "edge",
    "revalPending": false,
    "profile": "ATS_EDGE_TIER_CACHE",
    "type": "EDGE",
    "updPending": false
  }
]}
```

## POST

Post a server check result to the “serverchecks” table. Updates the resulting value from running a given check extension on a server.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Object

## Request Structure

The request only requires to have either `host_name` or `id` defined.

**host\_name** The hostname of the server to which this “servercheck” refers.

**id** The id of the server to which this “servercheck” refers.

**servercheck\_short\_name** The short name of the “servercheck”.

**value** The value of the “servercheck”

### 503: Request Example

```
POST /api/3.0/servercheck HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 113
Content-Type: application/json
```

(continues on next page)

<sup>1</sup> No roles are required to use this endpoint, however access is controlled by username. Only the reserved user extension is permitted the use of this endpoint.

(continued from previous page)

```
{
  "id": 1,
  "host_name": "edge",
  "servercheck_short_name": "test",
  "value": 1
}
```

## Response Structure

504: Response Example

```
{ "alerts": [
  {
    "level": "success",
    "text": "Server Check was successfully updated."
  }
]}
```

### servercheck/extensions

#### See also:

*Managing Traffic Ops Extensions*

#### GET

Retrieves the list of Traffic Ops extensions.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array



## Request Structure

Table 219: Request Query Parameters

Name	Re- quired	Description
id	no	Filter TO Extensions by the integral, unique identifier of an Extension
name	no	Filter TO Extensions by the name of an Extension
script_file	no	Filter TO Extensions by the base filename of the script that runs for the Extension
is-active	no	Boolean used to return either only active (1) or inactive(0) TO Extensions
type	no	Filter TO Extensions by the type of Extension
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 505: Request Example

```
GET /api/3.0/servercheck/extensions HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**additional\_config\_json** A string containing a JSON-encoded object with extra configuration options... inside a JSON object...

**description** A short description of the extension

---

**Note:** This is, unfortunately, null for all default extensions

---

**id** An integral, unique identifier for this extension definition

**info\_url** A URL where info about this extension may be found

**isactive** An integer describing the boolean notion of whether or not the extension is active; one of:

**0** disabled

1 enabled

**name** The name of the extension

**script\_file** The base filename of the script that runs for the extension

**servercheck\_shortname** The name of the column in the table at ‘Monitor’ -> ‘Cache Checks’ in Traffic Portal, where “Check Extension” output is displayed

**type** The Check *Type* of the extension. This will always be a CHECK\_EXTENSION type with the naming convention of CHECK\_EXTENSION\_\*.

**version** A (hopefully) semantic version number describing the version of the plugin

### 506: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Tue, 11 Dec 2018 20:51:48 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: n73jg9XR4V5Cwqq56Rf3wuIi99k3mM5u2NAjcZ/
→gQB8jvAFymDlnZqKeJ+wTll1vjIsHpXCOVXV7+5UGakLgA==
Transfer-Encoding: chunked

{ "response": [
  {
    "script_file": "ToPingCheck.pl",
    "version": "1.0.0",
    "name": "ILO_PING",
    "description": null,
    "info_url": "-",
    "additional_config_json": "{ check_name: \"ILO\", \"base_url\
→\": \"https://localhost\", \"select\": \"ilo_ip_address\", \"cron\": \
→\"9 * * * *\" }",
    "isactive": 1,
    "type": "CHECK_EXTENSION_BOOL",
    "id": 1,
    "servercheck_short_name": "ILO"
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "script_file": "ToPingCheck.pl",
      "version": "1.0.0",
      "name": "10G_PING",
      "description": null,
      "info_url": "-",
      "additional_config_json": "{ check_name: \"10G\", \"base_url\
→\": \"https://localhost\", \"select\": \"ip_address\", \"cron\": \"18_
→* * * *\" }",
      "isactive": 1,
      "type": "CHECK_EXTENSION_BOOL",
      "id": 2,
      "servercheck_short_name": "10G"
    }
  ]}

```

## POST

Creates a new Traffic Ops check extension.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** undefined

## Request Structure

**additional\_config\_json** An optional string containing a JSON-encoded object with extra configuration options... inside a JSON object...

**description** A short description of the extension

**info\_url** A URL where info about this extension may be found

**isactive** An integer describing the boolean notion of whether or not the extension is active; one of:

**0** disabled

**1** enabled

**name** The name of the extension

**script\_file** The base filename of the script that runs for the extension

<sup>1</sup> No roles are required to use this endpoint, however access is controlled by username. Only the reserved user extension is permitted the use of this endpoint.

**See also:**

*Managing Traffic Ops Extensions* for details on where the script should be located on the Traffic Ops server

**servercheck\_shortcode** The name of the column in the table at ‘Monitor’ -> ‘Cache Checks’ in Traffic Portal, where “Check Extension” output is displayed

**type** The *Type* of extension. Must be CHECK\_EXTENSION type with the naming convention of CHECK\_EXTENSION\_\*.

**version** A (hopefully) semantic version number describing the version of the plugin

507: Request Example

```
POST /api/3.0/servercheck/extensions HTTP/1.1
Host: cache1.example.com:6443
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 208
Content-Type: application/json

{
  "name": "test",
  "version": "0.0.1-1",
  "info_url": "",
  "script_file": "",
  "isactive": 0,
  "description": "A test extension for API examples",
  "servercheck_shortcode": "test",
  "type": "CHECK_EXTENSION_NUM"
}
```

## Response Structure

508: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Date: Wed, 12 Dec 2018 16:37:44 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↪GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 7M67PYnli6WzGQFS3g8Gh1SOyq6VENZMqm/
    ↪kUffOTLLFfuWSEuSLA65R5R+VyJiNjdqOG5Bp78mk+JYcqhtVGw==
Content-Length: 89

{ "supplemental":
  {
    "id": 5
  },
  "alerts": [{
    "level": "success",
    "text": "Check Extension Loaded."
  }]
}
```

`servercheck/extensions/{ID}`

## DELETE

Deletes a Traffic Ops server check extension definition. This does **not** delete the actual extension file.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** undefined

## Request Structure

Table 220: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the extension definition to be deleted

## 509: Request Example

```
DELETE /api/3.0/servercheck/extensions/16 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

<sup>1</sup> No roles are required to use this endpoint, however access is controlled by username. Only the reserved user extension is permitted the use of this endpoint.

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 510: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 12 Dec 2018 16:33:52 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: EB0Nu85azbGzaehDTAODP3NPqWbByIza1XQhgwtS W2WTXyK/
    ↳dxQtncp0YiJXy00tH9H+n+6BBfojB0b5h0dFPA==
Content-Length: 60

{ "alerts": [
  {
    "level": "success",
    "text": "Extension deleted."
  }
]}
```

## servers

### GET

Retrieves properties of all servers across all CDNs.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 221: Request Query Parameters

Name	Re- quired	Description
cache-group	no	Return only those servers within the <i>Cache Group</i> that has this <i>ID</i>
cache-group-Name	no	Return only those servers within the <i>Cache Group</i> that has this <i>Name</i>
dsId	no	Return only those servers assigned to the <i>Delivery Service</i> identified by this integral, unique identifier. If the Delivery Service has a <i>Topology</i> assigned to it, the <i>servers</i> endpoint will return each server whose <i>Cache Group</i> is associated with a <i>Topology Node</i> of that Topology and has the <i>Server Capabilities</i> that are <i>required by the Delivery Service</i> but excluding <i>Origin Servers</i> that are not assigned to the Delivery Service. For more information, see <i>Configure Multi-Site Origin</i> .
host-Name	no	Return only those servers that have this (short) hostname
id	no	Return only the server with this integral, unique identifier
profileId	no	Return only those servers that are using the <i>Profile</i> that has this <i>ID</i>
status	no	Return only those servers with this status - see <i>Health Protocol</i>
type	no	Return only servers of this <i>Type</i>
topology	no	Return only servers who belong to cachegroups assigned to the <i>Topology</i> identified by this name
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n<sup>th</sup></i> page of results, where “n” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

## 511: Request Example

```
GET /api/3.0/servers?hostName=mid HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** The integral, unique identifier of the CDN to which the server belongs

**cdnName** Name of the CDN to which the server belongs

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be `null`

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service<sup>1</sup>

**iloIpGateway** The IPv4 gateway address of the server's ILO service<sup>Page 837, 1</sup>

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service<sup>Page 837, 1</sup>

**iloPassword** The password of the of the server's ILO service user<sup>Page 837, 1</sup> - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' *Role(s)*

**iloUsername** The user name for the server's ILO service<sup>Page 837, 1</sup>

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

---

<sup>1</sup> For more information see the [Wikipedia page on Lights-Out management](#).



**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the physical location where the server resides

**physLocationId** An integral, unique identifier for the physical location where the server resides

**profile** The *Name* of the *Profile* used by this server

**profileDesc** A *Description* of the *Profile* used by this server

**profileId** The *ID* the *Profile* used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**rack** A string indicating “server rack” location

**routerHostName** The human-readable name of the router responsible for reaching this server

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server

**status** The *Status* of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the *Type* of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (*t3c*, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 512: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:06:25 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:06:25 GMT
Content-Length: 538

{ "response": [{
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "mid",
  "httpsPort": 443,
  "id": 12,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "lastUpdated": "2020-05-19 14:49:39+00",
  "mgmtIpAddress": "",
```

(continues on next page)

(continued from previous page)

```
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocation": "Apachecon North America 2018",
"physLocationId": 1,
"profile": "ATS_MID_TIER_CACHE",
"profileDesc": "Mid Cache - Apache Traffic Server",
"profileId": 10,
"rack": "",
"revalPending": false,
"routerHostName": "",
"routerPortName": "",
"status": "REPORTED",
"statusId": 3,
"tcpPort": 80,
"type": "MID",
"typeId": 12,
"updPending": false,
"xmppId": "",
"xmppPasswd": "",
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "172.26.0.4/16",
        "gateway": "172.26.0.1",
        "serviceAddress": true
      }
    ],
    "maxBandwidth": null,
    "monitor": false,
    "mtu": 1500,
    "name": "eth0"
  }
],
}],
"summary": {
  "count": 13
}}
```

## Summary Fields

The summary object returned by this method of this endpoint uses only the count *standard property*.

## POST

Allows a user to create a new server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server shall belong

**cdnId** The integral, unique identifier of the CDN to which the server shall belong

**domainName** The domain part of the server’s FQDN

**hostName** The (short) hostname of the server

**httpsPort** An optional port number on which the server listens for incoming HTTPS connections/requests

**iloIpAddress** An optional IPv4 address of the server’s ILO service<sup>Page 837, 1</sup>

**iloIpGateway** An optional IPv4 gateway address of the server’s ILO service<sup>Page 837, 1</sup>

**iloIpNetmask** An optional IPv4 subnet mask of the server’s ILO service<sup>Page 837, 1</sup>

**iloPassword** An optional string containing the password of the of the server’s ILO service user<sup>Page 837, 1</sup> - displays as simply `*****` if the currently logged-in user does not have the ‘admin’ or ‘operations’ *Role(s)*

**iloUsername** An optional string containing the user name for the server’s ILO service<sup>Page 837, 1</sup>

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be necessary, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be necessary, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface. This unsigned integer may not be less than 1280.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**physLocationId** An integral, unique identifier for the physical location where the server resides

**profileId** The *ID* the *Profile* that shall be used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**rack** An optional string indicating “server rack” location

**routerHostName** An optional string containing the human-readable name of the router responsible for reaching this server

**routerPortName** An optional string containing the human-readable name of the port used by the router responsible for reaching this server

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** An optional port number on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (T3C, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** An optional password used in XMPP communications with the server

513: Request Example

```
POST /api/3.0/servers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 599
Content-Type: application/json

{
  "cachegroupId": 6,
  "cdnId": 2,
  "domainName": "infra.ciab.test",
  "hostName": "test",
  "httpsPort": 443,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "interfaces": [
    {
      "ipAddresses": [
        {
          "address": "::1",
          "gateway": "::2",
          "serviceAddress": true
        },
        {
          "address": "0.0.0.1/24",
          "gateway": "0.0.0.2",
          "serviceAddress": true
        }
      ],
      "maxBandwidth": null,
      "monitor": true,
      "mtu": 1500,
      "name": "eth0"
    }
  ],
  "interfaceMtu": 1500,
  "interfaceName": "eth0",
  "ip6Address": "::1",
  "ip6Gateway": "::2",
  "ipAddress": "0.0.0.1",
```

(continues on next page)



(continued from previous page)

```

"ipGateway": "0.0.0.2",
"ipNetmask": "255.255.255.0",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocationId": 1,
"profileId": 10,
"routerHostName": "",
"routerPortName": "",
"statusId": 3,
"tcpPort": 80,
"typeId": 12,
"updPending": false
}

```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** The integral, unique identifier of the CDN to which the server belongs

**cdnName** Name of the CDN to which the server belongs

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be null

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service [Page 837, 1](#)

**iloIpGateway** The IPv4 gateway address of the server's ILO service [Page 837, 1](#)

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service [Page 837, 1](#)

**iloPassword** The password of the of the server's ILO service user<sup>Page 837, 1</sup> - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' `ROLE(s) <ROLE>`

**iloUsername** The user name for the server's ILO service<sup>Page 837, 1</sup>

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the *Physical Location* where the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* where the server resides

**profile** The *Name* of the *Profile* used by this server

**profileDesc** A *Description* of the *Profile* used by this server

**profileId** The *ID* the *Profile* used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**rack** A string indicating "server rack" location

**routerHostName** The human-readable name of the router responsible for reaching this server

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server

**status** The status of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the ‘type’ of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (T3C, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 514: Response Example

```
HTTP/1.1 201 Created
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:34:40 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:34:40 GMT
Content-Length: 562

{ "alerts": [
  {
    "text": "Server created",
    "level": "success"
  }
],
"response": {
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
```

(continues on next page)

(continued from previous page)

```

"cdnId": 2,
"cdnName": "CDN-in-a-Box",
"domainName": "infra.ciab.test",
"guid": null,
"hostName": "test",
"httpsPort": 443,
"id": 14,
"iloIpAddress": "",
"iloIpGateway": "",
"iloIpNetmask": "",
"iloPassword": "",
"iloUsername": "",
"lastUpdated": "2020-05-19 16:34:40+00",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocation": "Apachecon North America 2018",
"physLocationId": 1,
"profile": "ATS_MID_TIER_CACHE",
"profileDesc": "Mid Cache - Apache Traffic Server",
"profileId": 10,
"rack": null,
"revalPending": false,
"routerHostName": "",
"routerPortName": "",
"status": "REPORTED",
"statusId": 3,
"tcpPort": 80,
"type": "MID",
"typeId": 12,
"updPending": false,
"xmppId": null,
"xmppPasswd": null,
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "::1",
        "gateway": "::2",
        "serviceAddress": true
      },
      {
        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": true
      }
    ]
  }
]

```

(continues on next page)

(continued from previous page)

```

        }
      ],
      "maxBandwidth": null,
      "monitor": true,
      "mtu": 1500,
      "name": "eth0"
    }
  ]
}

```

## servers/details

Retrieves details of *Servers*.

Deprecated since version 3.1: This endpoint has been removed from the latest version of the API, and clients are advised to use *servers* instead.

### GET

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

**Note:** On top of the response including the response key that is of type array it will also include the keys `limit`, `orderBy`, and `size`.

## Request Structure

Table 222: Request Query Parameters

Name	Required	Description
host-Name	Required if no physLocationID provided	Return only the servers with this (short) hostname. Capitalization of “hostName” is important.
phys-LocationID	Required if no hostName provided	Return only servers with this integral, unique identifier for the physical location where the server resides. Capitalization of “physLocationID” is important.

### 515: Request Example

```
GET /api/3.0/servers/details?hostName=edge HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**limit** The maximum size of the `response` array, also indicative of the number of results per page using the pagination requested by the query parameters (if any) - this should be the same as the `limit` query parameter (if given)

**orderby** A string that names the field by which the elements of the `response` array are ordered - should be the same as the `orderby` request query parameter (if given)

**response** An array of objects, each of which represents the details of a given *Server*.

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cdnName** Name of the CDN to which the server belongs

**deliveryservices** An array of integral, unique identifiers for *Delivery Services* to which this server belongs

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be `null`

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service<sup>1</sup>

**iloIpGateway** The IPv4 gateway address of the server's ILO service<sup>1</sup>

---

<sup>1</sup> For more information see the [Wikipedia page on Lights-Out management](#).

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service<sup>1</sup>

**iloPassword** The password of the of the server's ILO service user<sup>Page 855, 1</sup> - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' *Role(s)*

**iloUsername** The user name for the server's ILO service<sup>Page 855, 1</sup>

**interfaces** An array of interface and IP address information

**max\_bandwidth** The maximum allowed bandwidth for this interface to be considered "healthy" by Traffic Monitor. This has no effect if *monitor* is not true. Values are in kb/s. The value *0* means "no limit".

**monitor** A boolean indicating if Traffic Monitor should monitor this interface

**mtu** The MTU to configure for *interfaceName*

**See also:**

[The Wikipedia article on Maximum Transmission Unit](#)

**name** The network interface name used by the server.

**ipAddresses** An array of the IP address information for the interface

**address** The IPv4 or IPv6 address and subnet mask of the server - applicable for the interface name

**gateway** The IPv4 or IPv6 gateway address of the server - applicable for the interface name

**service\_address** A boolean determining if content will be routed to the IP address

**mgmtIpAddress** The IPv4 address of the server's management port



**mgmtIpGateway** The IPv4 gateway of the server's management port

**mgmtIpNetmask** The IPv4 subnet mask of the server's management port

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the physical location where the server resides

**profile** The *Name* of the *Profile* used by this server

**profileDesc** A *Description* of the *Profile* used by this server

**rack** A string indicating "server rack" location

**routerHostName** The human-readable name of the router responsible for reaching this server

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server

**status** The status of the server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with "HTTP port", as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the 'type' of this server

**xmppId** A system-generated UUID used to generate a server hashId for use in Traffic Router's consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

**size** The page number - if pagination was requested in the query parameters, else 0 to indicate no pagination - of the results represented by the response array. This is named "size" for legacy reasons

## 516: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 01:27:36
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳HW2F3CEpohNAvNlEDhUfXmtwpEka4dwUWfVUSSjW98aXiv10vI6ysRIcC2P9huabCz5fdHqY3tp0LR4ekv
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 00:27:36 GMT
Content-Length: 493

{
  "limit": 1000,
  "orderby": "hostName",
  "response": [
    {
      "cachegroup": "CDN_in_a_Box_Edge",
      "cdnName": "CDN-in-a-Box",
      "deliveryservices": [
        1
      ],
      "domainName": "infra.ciab.test",
      "guid": null,
      "hardwareInfo": null,
      "hostName": "edge",
      "httpsPort": 443,
      "id": 5,
      "iloIpAddress": "",
      "iloIpGateway": "",
      "iloIpNetmask": "",
      "iloPassword": "",
      "iloUsername": "",
      "mgmtIpAddress": "",
      "mgmtIpGateway": "",
      "mgmtIpNetmask": "",
      "offlineReason": "",
      "physLocation": "Apachecon North America 2018",
      "profile": "ATS_EDGE_TIER_CACHE",
      "profileDesc": "Edge Cache - Apache Traffic Server",
      "rack": "",
    }
  ]
}

```

(continues on next page)

(continued from previous page)

```

    "routerHostName": "",
    "routerPortName": "",
    "status": "REPORTED",
    "tcpPort": 80,
    "type": "EDGE",
    "xmppId": "edge",
    "xmppPasswd": "",
    "interfaces": [
      { "ipAddresses": [
          {
            "address": "172.16.239.100",
            "gateway": "172.16.239.1",
            "service_address": true
          },
          {
            "address": "fc01:9400:1000:8::100",
            "gateway": "fc01:9400:1000:8::1",
            "service_address": true
          }
        ]
      },
      {
        "max_bandwidth": 0,
        "monitor": true,
        "mtu": 1500,
        "name": "eth0"
      }
    ]
  },
  "size": 1
}

```

`servers/{{HostName-Or-ID}}/update`

**POST**

*Queue* or dequeue updates and revalidation updates for a specific server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 223: Request Path Parameters

Name	Description
HostName-OR-ID	The hostName or integral, unique identifier of the server on which updates are being queued or dequeued

Table 224: Request Query Parameters

Name	Re-quired	Description
updated (Depre-cated)	no	The value to set for the queue update flag on this server. May be ‘true’ or ‘false’.
reval_updated (Dep-recated)	no	The value to set for the queue update flag on this server. May be ‘true’ or ‘false’.

Deprecated since version 3.1: The boolean values above have resulted in an unintended race condition. These fields are removed in APIv4 and replaced.

### 517: Request Example

```
POST /api/3.0/servers/my-edge/update?updated=true&reval_updated=false.
→HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 518: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Mon, 10 Dec 2018 18:20:04 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: 9Mmo9hIFZyF5gAvfdJD//
→VH9eNgiHVLinXt88H0GlJSHhwND8gMxaFyC+f9XZfiNAoGd1MKi1934ZJGmaIR6qQ==
Content-Length: 49

{
  "alerts" :
  [
    {
      "text" : "successfully set server 'my-edge' ↵
→updated=true reval_updated=false",
      "level" : "success"
    }
  ]
}
```

## servers/{hostname}/update\_status

**Note:** This endpoint only truly has meaning for *cache servers*, though it will return a valid response for any server configured in Traffic Ops.

### GET

Retrieves information regarding pending updates and *Content Invalidation Jobs* for a given server

**Auth. Required** Yes

**Roles Required** None

**Response Type** undefined - this endpoint will return a top-level array containing the response, as opposed to within a response object

### Request Structure

Table 225: Request Path Parameters

Name	Description
hostname	The (short) hostname of the server being inspected

### 519: Request Example

```
GET /api/3.0/servers/edge/update_status HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

Each object in the returned array<sup>1</sup> will contain the following fields:

- host\_id** The integral, unique identifier for the server for which the other fields in this object represent the pending updates and revalidation status
- host\_name** The (short) hostname of the server for which the other fields in this object represent the pending updates and revalidation status
- parent\_pending** A boolean telling whether or not any *Topology* ancestor or *parent* of this server has pending updates
- parent\_reval\_pending** A boolean telling whether or not any *Topology* ancestor or *parent* of this server has pending *Content Invalidation Jobs*
- reval\_pending** true if the server has pending *Content Invalidation Jobs*, false otherwise
- status** The name of the status of this server

#### See also:

*Health Protocol* gives more information on how these statuses are used, and the GET method of the *statuses* endpoint can be used to retrieve information about all server statuses configured in Traffic Ops.

- upd\_pending** true if the server has pending updates, false otherwise
- use\_reval\_pending** A boolean which tells *ORT* whether or not this version of Traffic Ops should use pending *Content Invalidation Jobs*

---

**Note:** This field was introduced to give *ORT* the ability to work with Traffic Control versions 1.x and 2.x seamlessly - as of Traffic Control v3.0 there is no reason for this field to ever be false.

---

### 520: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
```

(continues on next page)

---

<sup>1</sup> The returned object is an array, and there is no guarantee that one server exists for a given hostname. However, for each server in the array, that server's update status will be accurate for the server with that particular server ID.

(continued from previous page)

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳R6BjNVrceCHGn3eGDqQ1yDiBnEDGQe7QtOMIsRwlpck9SZR8chRQznrkTF3YdROAZ1l8BxR3fXTIvKHIz
    ↳dA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 04 Feb 2019 16:24:01 GMT
Content-Length: 174

[{"host_name": "edge",
  "upd_pending": false,
  "reval_pending": false,
  "use_reval_pending": true,
  "host_id": 10,
  "status": "REPORTED",
  "parent_pending": false,
  "parent_reval_pending": false
}]
```

`servers/{ID}`

**PUT**

Allow user to edit a server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 226: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a server

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server shall belong

**cdnId** The integral, unique identifier of the CDN to which the server shall belong

**domainName** The domain part of the server's FQDN

**hostName** The (short) hostname of the server

**httpsPort** An optional port number on which the server listens for incoming HTTPS connections/requests

**iloIpAddress** An optional IPv4 address of the server's ILO service<sup>1</sup>

**iloIpGateway** An optional IPv4 gateway address of the server's ILO service<sup>Page 862, 1</sup>

**iloIpNetmask** An optional IPv4 subnet mask of the server's ILO service<sup>Page 862, 1</sup>

**iloPassword** An optional string containing the password of the of the server's ILO service user<sup>Page 862, 1</sup> - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' `ROLE(s) <ROLE>`

**iloUsername** An optional string containing the user name for the server's ILO service<sup>Page 862, 1</sup>

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be necessary, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be necessary, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface -

---

<sup>1</sup> For more information see the [Wikipedia page on Lights-Out management](#).



which is therefore sometimes referred to as the “service interface” for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning “no limit”. It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface. This unsigned integer must not be less than 1280.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface’s device name on the server, e.g. `eth0`.

**mgmtIpAddress** The IPv4 address of some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**physLocationId** An integral, unique identifier for the physical location where the server resides

**profileId** The *ID* the *Profile* that shall be used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**rack** An optional string indicating “server rack” location

**routerHostName** An optional string containing the human-readable name of the router responsible for reaching this server

**routerPortName** An optional string containing the human-readable name of the port used by the router responsible for reaching this server

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** An optional port number on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (*t3c*, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** An optional password used in XMPP communications with the server

### 521: Request Example

```
PUT /api/3.0/servers/14 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 599
Content-Type: application/json

{
  "cachegroupId": 6,
  "cdnId": 2,
  "domainName": "infra.ciab.test",
  "hostName": "quest",
  "httpsPort": 443,
  "iloIpAddress": "",
```

(continues on next page)

(continued from previous page)

```

"iloIpGateway": "",
"iloIpNetmask": "",
"iloPassword": "",
"iloUsername": "",
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "::1",
        "gateway": "::2",
        "serviceAddress": true
      },
      {
        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": false
      }
    ],
    "maxBandwidth": null,
    "monitor": true,
    "mtu": 1500,
    "name": "bond0"
  }
],
"interfaceMtu": 1500,
"interfaceName": "eth0",
"ip6Address": "::1",
"ip6Gateway": "::2",
"ipAddress": "0.0.0.1",
"ipGateway": "0.0.0.2",
"ipNetmask": "255.255.255.0",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocationId": 1,
"profileId": 10,
"routerHostName": "",
"routerPortName": "",
"statusId": 3,
"tcpPort": 80,
"typeId": 12,
"updPending": false
}

```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** The integral, unique identifier of the CDN to which the server belongs

**cdnName** Name of the CDN to which the server belongs

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be `null`

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service Page 862, 1

**iloIpGateway** The IPv4 gateway address of the server's ILO service Page 862, 1

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service Page 862, 1

**iloPassword** The password of the of the server's ILO service user Page 862, 1 - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' `ROLE(S) <ROLE>`

**iloUsername** The user name for the server's ILO service Page 862, 1

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces`

property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the *Physical Location* where the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* where the server resides

**profile** The *Name* of the *Profile* used by this server

**profileDesc** A *Description* of the *Profile* used by this server

**profileId** The *ID* the *Profile* used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**rack** A string indicating “server rack” location

**routerHostName** The human-readable name of the router responsible for reaching this server

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server

**status** The status of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the ‘type’ of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (T3C, formerly ORT)

**xmppId** A system-generated UUID used to generate a server hashId for use in Traffic Router's consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 522: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:46:33.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:46:33 GMT
Content-Length: 566

{ "alerts": [
  {
    "text": "Server updated",
    "level": "success"
  }
],
"response": {
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "quest",
  "httpsPort": 443,
  "id": 14,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "lastUpdated": "2020-05-19 16:46:33+00",
```

(continues on next page)

(continued from previous page)

```
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocation": "Apachecon North America 2018",
"physLocationId": 1,
"profile": "ATS_MID_TIER_CACHE",
"profileDesc": "Mid Cache - Apache Traffic Server",
"profileId": 10,
"rack": null,
"revalPending": false,
"routerHostName": "",
"routerPortName": "",
"status": "REPORTED",
"statusId": 3,
"tcpPort": 80,
"type": "MID",
"typeId": 12,
"updPending": false,
"xmppId": null,
"xmppPasswd": null,
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "::1",
        "gateway": "::2",
        "serviceAddress": true
      },
      {
        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": false
      }
    ],
    "maxBandwidth": null,
    "monitor": true,
    "mtu": 1500,
    "name": "bond0"
  }
]
}}
```



## DELETE

Allow user to delete server through api.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

Changed in version 3.0: In older versions of the API, this endpoint did not return a response object. It now returns a representation of the deleted server.

## Request Structure

Table 227: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a server

### 523: Request Example

```
DELETE /api/3.0/servers/14 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belonged

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belonged

**cdnId** The integral, unique identifier of the CDN to which the server belonged

**cdnName** Name of the CDN to which the server belonged

**domainName** The domain part of the server’s FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be null

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listened for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service [Page 862, 1](#)

**iloIpGateway** The IPv4 gateway address of the server's ILO service [Page 862, 1](#)

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service [Page 862, 1](#)

**iloPassword** The password of the of the server's ILO service user [Page 862, 1](#) - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' *Role(s)*

**iloUsername** The user name for the server's ILO service [Page 862, 1](#)

**interfaces** A set of the network interfaces that were in use by the server

**ipAddresses** A set of objects representing IP Addresses that were assigned to this network interface

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address".

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors would have considered the entire server unhealthy - which includes *all* configured network interfaces. If this was `null`, it has the meaning "no limit". It had no effect if `monitor` was not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should have been monitored by Traffic Monitor for statistics and health consideration

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information was either not available or not applicable for this interface.

**name** The name of the interface. It is the same as the network interface's device name on the server, e.g. `eth0`.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server that was used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server that was used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server that was used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server was in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the physical location where the server resided

**physLocationId** An integral, unique identifier for the physical location where the server resided

**profile** The *Name* of the *Profile* which was used by this server

**profileDesc** A *Description* of the *Profile* which was used by this server

**profileId** The *ID* the *Profile* which was used by this server

**revalPending** A boolean value which, if `true` indicates that this server had pending content invalidation/revalidation

**rack** A string indicating “server rack” location

**routerHostName** The human-readable name of the router responsible for reaching this server

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server

**status** The *Status* of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listened for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the *Type* of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server had updates of some kind pending, typically to be acted upon by Traffic Ops *ORT*

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 524: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:50:13 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:50:13 GMT
Content-Length: 568

{ "alerts": [
  {
    "text": "Server deleted",
    "level": "success"
  }
],
"response": {
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "quest",
  "httpsPort": 443,
  "id": 14,
  "iloIpAddress": "",
  "iloIpGateway": "",
```

(continues on next page)

(continued from previous page)

```

"iloIpNetmask": "",
"iloPassword": "",
"iloUsername": "",
"lastUpdated": "2020-05-19 16:46:33+00",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocation": "Apachecon North America 2018",
"physLocationId": 1,
"profile": "ATS_MID_TIER_CACHE",
"profileDesc": "Mid Cache - Apache Traffic Server",
"profileId": 10,
"rack": null,
"revalPending": false,
"routerHostName": "",
"routerPortName": "",
"status": "REPORTED",
"statusId": 3,
"tcpPort": 80,
"type": "MID",
"typeId": 12,
"updPending": false,
"xmppId": null,
"xmppPasswd": null,
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": false
      },
      {
        "address": "::1",
        "gateway": "::2",
        "serviceAddress": true
      }
    ],
    "maxBandwidth": null,
    "monitor": true,
    "mtu": 1500,
    "name": "bond0"
  }
]
}}
```

`servers/{{ID}}/deliveryservices`

GET

Retrieves all *Delivery Services* assigned to a specific server.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

## Request Structure

Table 228: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the server for which assigned <i>Delivery Services</i> shall be listed

Table 229: Request Query Parameters

Name	Required	Description
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <sup>n</sup> th page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## 525: Request Example

```
GET /api/3.0/servers/9/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

<sup>1</sup> Only the *Delivery Services* visible to the requesting user’s *Tenant* will appear, regardless of their *Role* or the *Delivery Services*’ actual ‘server assignment’ status.

## Response Structure

- active** A boolean that defines *Active*.
- anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*
- cacheurl** A *Cache URL Expression*  
 Deprecated since version ATCv3.0: This field has been deprecated in Traffic Control 3.x and is subject to removal in Traffic Control 4.x or later
- ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons
- cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs
- cdnName** Name of the *CDN* to which the *Delivery Service* belongs
- checkPath** A *Check Path*
- consistentHashRegex** A *Consistent Hashing Regular Expression*
- consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*
- deepCachingType** The *Deep Caching* setting for this *Delivery Service*
- displayName** The *Display Name*
- dnsBypassCname** A *DNS Bypass CNAME*
- dnsBypassIp** A *DNS Bypass IP*
- dnsBypassIp6** A *DNS Bypass IPv6*
- dnsBypassTtl** The *DNS Bypass TTL*
- dscp** A *DSCP* to be used within the *Delivery Service*
- ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*
- edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*
- exampleURLs** An array of *Example URLs*
- firstHeaderRewrite** A set of *First Header Rewrite Rules*
- fqPacingRate** The *Fair-Queuing Pacing Rate Bps*
- geoLimit** An integer that defines the *Geo Limit*
- geoLimitCountries** A string containing a comma-separated list defining the *Geo Limit Countries*
- geoLimitRedirectUrl** A *Geo Limit Redirect URL*
- geoProvider** The *Geolocation Provider*
- globalMaxMbps** The *Global Max Mbps*
- globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in *Traffic Ops's Custom Date/Time Format*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**longDesc1** The 2nd long description of this *Delivery Service*

**longDesc2** the 3rd long description of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
- setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
- type** The type of match performed using pattern.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated



- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service's xml\_id*

#### 526: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: CFmtW41aoDezCYxtAXnS54dffOD6jdxDJ2/
→LMpbBqnn5kac7JQhdFAWF109s195XVSUV85JHFzXZTw/mJabQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Jun 2019 17:01:30 GMT
Content-Length: 1500
```

```
{ "response": [ {
  "active": true,
  "anonymousBlockingEnabled": false,
  "cacheurl": null,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "checkPath": null,
  "displayName": "Demo 1",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "edgeHeaderRewrite": null,
  "firstHeaderRewrite": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectURL": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "id": 1,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": true,
  "lastHeaderRewrite": null,
  "lastUpdated": "2019-06-10 15:14:29+00",
  "logsEnabled": true,
  "longDesc": "Apachecon North America 2018",
  "longDesc1": null,
  "longDesc2": null,
  "matchList": [
    {
      "type": "HOST_REGEX",
      "setNumber": 0,
```

(continues on next page)

(continued from previous page)

```

        "pattern": ".*\\.demo1\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
    "http://video.demo1.mycdn.ciab.test",
    "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"trResponseHeaders": null,
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [
    "abc",
    "pdq",
    "xxx",
    "zyx"
],
"maxOriginConnections": 0,
"ecsEnabled": false,
"rangeSliceBlockSize": null,

```

(continues on next page)

(continued from previous page)

```
"topology": null
}}}
```

## POST

Assign an arbitrary number of *Delivery Services* to a single server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 230: Request Path Parameters

Name	Re- quired	Description
ID	Yes	The integral, unique identifier of the server that you want to assign delivery services to.

Table 231: Request Query Parameters

Name	Re- quired	Description
re- place	Yes	Whether the list of <i>Delivery Services</i> you provide should replace the existing list or be merged with the existing list. Must be a 1, or true, or 0, or false.

The request body is an array of IDs of *Delivery Services* that you want to assign to the server. The array can be empty, but it must be provided.

### 527: Request Example

```
POST /api/3.0/servers/6/deliveryservices?replace=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 3

[
  1
]
```

## Response Structure

**dsIds** An array of integral, unique identifiers for *Delivery Services* which the request added to server. If `:replace:` is false, *Delivery Services* that are already assigned will remain, though they are not listed by `:dsIds:`.

**replace** The `:replace:` value you provided in the body of the request, or null if none was provided.

**serverId** The server's integral, unique identifier

### 528: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 09:08:32
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳iV+JzAZSsmlxRZsNtIRg3oA9470hAwRMpq5xhcYVi0Y831Trx2YRlsyhYp0PqHg5+QPoXHGF0nx8uso0f
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 08:08:32 GMT
Content-Length: 129

{
  "alerts": [
    {
      "text": "successfully assigned dses to server",
      "level": "success"
    }
  ],
  "response": {
    "serverId": 6,
    "dsIds": [
      1
    ],
    "replace": true
  }
}
```

## servers/{{ID}}/queue\_update

**Caution:** In the vast majority of cases, it is advisable that the PUT method of the *servers/{{ID}}* endpoint be used instead.

### POST

*Queue* or dequeue updates for a specific server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

Table 232: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the server on which updates are being queued or dequeued

**action** A string describing what action to take regarding server updates; one of:

**queue** *Queue Updates* for the server, propagating configuration changes to the actual server

**dequeue** Cancels any pending updates on the server

#### 529: Request Example

```
POST /api/3.0/servers/13/queue_update HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 22
Content-Type: application/json

{
  "action": "dequeue"
}
```

## Response Structure

**action** The action processed, one of:

**queue** *Queue Updates* was performed on the server, propagating configuration changes to the actual server

**dequeue** Canceled any pending updates on the server

**serverId** The integral, unique identifier of the server on which action was taken

### 530: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Mon, 10 Dec 2018 18:20:04 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 9Mmo9hIFZyF5gAvfdJD//
→VH9eNgiHVLinXt88H0GlJSHhwND8gMxaFyC+f9XZfiNAoGd1MKi1934ZJGmaIR6qQ==
Content-Length: 49

{
  "response": {
    "serverId": "13",
    "action": "dequeue"
  }
}
```

## servers/{{ID}}/status

### PUT

Updates server status and queues updates on all descendant *Topology* nodes or child caches if server type is EDGE or MID. Also, captures offline reason if status is set to ADMIN\_DOWN or OFFLINE and prepends offline reason with the user that initiated the status change.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 233: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the server whose status is being changed

**offlineReason** A string containing the reason for the status change

**status** The name or integral, unique identifier of the server's new status

531: Request Example

```
PUT /api/3.0/servers/13/status HTTP/1.1
```

```
Host: trafficops.infra.ciab.test
```

```
User-Agent: curl/7.47.0
```

```
Accept: */*
```

```
Cookie: mojolicious=...
```

```
Content-Length: 56
```

```
Content-Type: application/json
```

```
{
  "status": "ADMIN_DOWN",
  "offlineReason": "Bad drives"
}
```

## Response Structure

532: Response Example

```
HTTP/1.1 200 OK
```

```
Access-Control-Allow-Credentials: true
```

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
```

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

```
Access-Control-Allow-Origin: *
```

```
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

```
Content-Type: application/json
```

```
Date: Mon, 10 Dec 2018 18:08:44 GMT
```

```
X-Server-Name: traffic_ops_golang/
```

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
```

```
Vary: Accept-Encoding
```

```
Whole-Content-Sha512:
→LS1jCo5eMVKxmeYDol0I2LgLYazocSggR5hynNoLcPmMov9u2s3ulksPdQtG1N3aS+VM9tdMsCrahFPra
```

```
Content-Length: 158
```

(continues on next page)



(continued from previous page)

```
{ "alerts": [
  {
    "level": "success",
    "text": "Updated status [ ADMIN_DOWN ] for quest.infra.ciab.
→test [ admin: Bad drives ] and queued updates on all child caches"
  }
]}
```

## service\_categories

### GET

Get all requested *Service Categories*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 234: Request Query Parameters

Name	Description
name	Filter for <i>Service Categories</i> with this name
or-derby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 533: Request Example

```
GET /api/3.0/service_categories?name=SERVICE_CATEGORY_NAME HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**name** This *Service Category*’s name

**lastUpdated** The date and time at which this *Service Category* was last modified, in *Traffic Ops*’s *Custom Date/Time Format*

### 534: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Yzr6TfhxgpZ3pbrrr4TRG4wC3PlnHDDzgs2igtz/
→1ppLSy2MzugqaGW4y5yzwzl5T3+7q6HWej7GQZt1XIVeZQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 11 Mar 2020 20:02:47 GMT
Content-Length: 102

{
  "response": [
    {
      "lastUpdated": "2020-03-04 15:46:20-07",
      "name": "SERVICE_CATEGORY_NAME"
    }
  ]
}
```

## POST

Create a new service category.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**name** This *Service Category*'s name

535: Request Example

```
POST /api/3.0/service_categories HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 48
Content-Type: application/json

{
  "name": "SERVICE_CATEGORY_NAME",
}
```

## Response Structure

**name** This *Service Category*'s name

**lastUpdated** The date and time at which this *Service Category* was last modified,  
in *Traffic Ops's Custom Date/Time Format*

536: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +pJm4c30+JTaSXNt+LP+u240Ba/
→SsvSSDOQ4rDc6hcyZ0FIL+iY/WWrMHhpLulRGKGY88bM4YPCMaxGn3FZ9yQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 11 Mar 2020 20:12:20 GMT
Content-Length: 154

{
  "alerts": [
    {
      "text": "serviceCategory was created.",
      "level": "success"
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```

    }
  ],
  "response": {
    "lastUpdated": "2020-03-11 14:12:20-06",
    "name": "SERVICE_CATEGORY_NAME"
  }
}

```

## DELETE

Deletes a specific *Service Category*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

537: Request Example

```

DELETE /api/3.0/service_categories/my-service-category HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0

```

## Response Structure

538: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 17 Aug 2020 16:13:31
→GMT; Max-Age=3600; HttpOnly

```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: yErJobzG9IA0khvqZQK+Yi7X4pFVvOqxn6PjrdzN5DnKVm/
→K8Kka3REul1XmKJnMXVRY8RayoEVDm16mBFfe4Q==
X-Server-Name: traffic_ops_golang/
Date: Mon, 17 Aug 2020 15:13:31 GMT
Content-Length: 93

{
  "alerts": [
    {
      "text": "serviceCategory was deleted.",
      "level": "success"
    }
  ]
}
```

## snapshot

### PUT

Performs a CDN *Snapshot*. Effectively, this propagates the new *configuration* of the CDN to its *operating state*, which replaces the output of the `cdns/{name}/snapshot` endpoint with the output of the `cdns/{name}/snapshot/new` endpoint. This also changes the output of the `cdns/{name}/configs/monitoring` endpoint since that endpoint returns the latest monitoring information from the *operating state*.

**Note:** By default, snapshotting the CDN also deletes all HTTPS certificates for every *Delivery Service* which has been deleted since the last *Snapshot*. In order to disable this behavior, set `disable_auto_cert_deletion` in `cdn.conf` to `true`.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 235: Request Query Parameters

Name	Description
cdn	The name of the CDN for which a <i>Snapshot</i> shall be taken
cdnID	The id of the CDN for which a <i>Snapshot</i> shall be taken

---

**Note:** At least one query parameter must be given.

---

### 539: Request Example

```
PUT /api/3.0/snapshot?cdn=CDN-in-a-Box HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 540: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳gmaWI0tKgNFPY00zMrLCGDosBJkPbeIvW4BH9tEh96VjBqyWqzjgPySoMa3ViM1BQXA6VAUOGmc76VyhB
X-Server-Name: traffic_ops_golang/
Date: Wed, 18 Mar 2020 15:51:48 GMT
Content-Length: 47

{
  "response": "SUCCESS"
}
```

## staticdnsentries

### GET

Retrieve all static DNS entries configured within Traffic Control

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 236: Request Query Parameters

Name	Re- quired	Description
ad- dress	no	Return only static DNS entries that operate on this address/CNAME
cachegroup	no	Return only static DNS entries assigned to the <i>Cache Group</i> that has this <i>Name</i>
cachegroupId	no	Return only static DNS entries assigned to the <i>Cache Group</i> that has this <i>ID</i>
deliv- ery- ser- vice	no	Return only static DNS entries that apply within the domain of the <i>Delivery Service</i> with this <i>xml_id</i>
deliv- ery- servi- ceId	no	Return only static DNS entries that apply within the domain of the <i>Delivery Service</i> identified by this integral, unique identifier
host	no	Return only static DNS entries that resolve this FQDN
id	no	Return only the static DNS entry with this integral, unique identifier
ttd	no	Return only static DNS entries with this TTL
type	no	Return only static DNS entries of this type
typeId	no	Return only static DNS entries of the type identified by this integral, unique identifier
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n<sup>th</sup></i> page of results, where “ <i>n</i> ” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

### 541: Request Example

```
GET /api/3.0/staticdnsentries?address=foo.bar HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**address** If `typeId` identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which `host` shall be resolved

**cachegroup** An optional string containing the *Name of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryservice** The name of a *Delivery Service* under the domain of which this static DNS entry shall be active

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If `typeId` identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the Fully Qualified Domain Name (FQDN) which shall resolve to `address`

**id** An integral, unique identifier for this static DNS entry

**ttl** The TTL of this static DNS entry in seconds

**type** The name of the type of this static DNS entry

**typeId** The integral, unique identifier of the *Type* of this static DNS entry



## 542: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Px1zTH3ihg+hfmDADGcap0Juuud39fGsX5Y3CzqaFNmRwFu1ZLMz0sy0EN2pb7vp0tpI6/
    ↳zeIUYAC3dbSbw0mA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:04:33 GMT
Content-Length: 226

{ "response": [
  {
    "address": "foo.bar.",
    "cachegroup": null,
    "cachegroupId": null,
    "deliveryservice": "demo1",
    "deliveryserviceId": 1,
    "host": "test",
    "id": 2,
    "lastUpdated": "2018-12-10 19:59:56+00",
    "ttl": 300,
    "type": "CNAME_RECORD",
    "typeId": 41
  }
]}
```

## POST

Creates a new, static DNS entry.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**address** If `typeId` identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which `host` shall be resolved

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If `typeId` identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the FQDN which shall resolve to `address`

**ttl** The TTL of this static DNS entry in seconds

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

### 543: Request Example

```
POST /api/3.0/staticdnsentries HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 92
Content-Type: application/json

{
  "address": "test.quest.",
  "deliveryserviceId": 1,
  "host": "test",
  "ttl": 300,
  "typeId": 41
}
```

## Response Structure

**address** If `typeId` identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which `host` shall be resolved

**cachegroup** An optional string containing the *Name of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryservice** The name of a *Delivery Service* under the domain of which this static DNS entry shall be active

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If `typeId` identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the Fully Qualified Domain Name (FQDN) which shall resolve to `address`

**id** An integral, unique identifier for this static DNS entry

**ttl** The TTL of this static DNS entry in seconds

**type** The name of the *Type* of this static DNS entry

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

### 544: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 8dcJyjw2NJZx0L90z16P7g/
→7j5A1jlpYiY6Y+rRVQ2wGcwYI3yiGPrz6ur0qKzgqEBBsh8aPF44WTHAR9jUJdg==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 19:54:19 GMT
Content-Length: 282

{ "alerts": [
  {
    "text": "staticDNSEntry was created.",
    "level": "success"
  }
],
"response": {
  "address": "test.quest.",
  "cachegroup": null,
  "cachegroupId": null,
  "deliveryservice": null,
  "deliveryserviceId": 1,
  "host": "test",
  "id": 2,
  "lastUpdated": "2018-12-10 19:54:19+00",
  "ttl": 300,
  "type": "CNAME_RECORD",
  "typeId": 41
}}
```

## PUT

Updates a static DNS entry.

**Auth. Required** Yes

**Role(s) Required** “admin” or “operator”

**Response Type** Object

## Request Structure

Table 237: Request Query Parameters

Name	Description
id	The integral, unique identifier of the static DNS entry to modify

**address** If typeId identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which host shall be resolved

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If typeId identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the Fully Qualified Domain Name (FQDN) which shall resolve to address

**ttl** The TTL of this static DNS entry in seconds

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

#### 545: Request Example

```
PUT /api/3.0/staticdnsentries?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 89
Content-Type: application/json

{
  "address": "foo.bar.",
  "deliveryserviceId": 1,
  "host": "test",
  "ttl": 300,
  "typeId": 41
}
```

## Response Structure

**address** If typeId identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which host shall be resolved

**cachegroup** An optional string containing the *Name of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

- deliveryservice** The name of a *Delivery Service* under the domain of which this static DNS entry shall be active
- deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active
- host** If typeId identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the FQDN which shall resolve to address
- id** An integral, unique identifier for this static DNS entry
- ttl** The TTL of this static DNS entry in seconds
- type** The name of the *Type* of this static DNS entry
- typeId** The integral, unique identifier of the *Type* of this static DNS entry

#### 546: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳+FaYmpnlIIzVSBq0nosw29NZcV9xFhlVgWuUqXUyiDihVUSzX4jrdAlORDgzDvKsYQB8LSkPdGHwt1zjg
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 19:59:56 GMT
Content-Length: 279

{ "alerts": [
  {
    "text": "staticDNSEntry was updated.",
    "level": "success"
  }
],
"response": {
  "address": "foo.bar.",
  "cachegroup": null,
  "cachegroupId": null,
  "deliveryservice": null,
  "deliveryserviceId": 1,
```

(continues on next page)

(continued from previous page)

```
"host": "test",
"id": 2,
"lastUpdated": "2018-12-10 19:59:56+00",
"ttl": 300,
"type": "CNAME_RECORD",
"typeId": 41
}}
```

## DELETE

Delete staticdnsentries.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 238: Request Query Parameters

Name	Description
id	The integral, unique identifier of the static DNS entry to delete

### 547: Request Example

```
DELETE /api/3.0/staticdnsentries?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 548: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→g6uqHPU44LuTtqU2ahtazrVCpcpNWVc9kWJQ0YRuiVLDnsm39KOB/xt3XM6j0/
→X3WYiIawnNspkxRC85LJHwFA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:05:52 GMT
Content-Length: 69

{ "alerts": [
  {
    "text": "staticDNSEntry was deleted.",
    "level": "success"
  }
]}

```

## stats\_summary

### GET

Either retrieve a list of summary stats or the timestamp of the latest recorded stats summary.

What is returned is driven by the query parameter `lastSummaryDate`.

If the parameter is set it will return an object with the latest timestamp, else an array of summary stats will be returned.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array or Object



## Request Structure

### Summary Stats

Table 239: Request Query Parameters

Name	Required	Description
deliveryServiceName	no	Return only summary stats that were reported for <i>Delivery Service</i> with the given display name
cdnName	no	Return only summary stats that were reported for CDN with the given name
statName	no	Return only summary stats that were reported for given stat name
orderBy	no	Choose the ordering of the results - can only be one of deliveryServiceName, statName or cdnName
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 549: Request Example

```
GET /api/3.0/stats_summary HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Last Updated Summary Stat

Table 240: Request Query Parameters

Name	Required	Description
statName	no	Get latest updated date for the given stat
lastSummaryDate	yes	Tells route to get only latest updated timestamp

## 550: Request Example

```
GET /api/3.0/stats_summary?lastSummaryDate=true HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### Summary Stats

**cdnName** The CDN name for which the summary stat was taken for

---

**Note:** If the `cdn` is equal to `all` it represents `summary_stats` across all delivery services across all CDNs

---

**deliveryServiceName** The *Delivery Service* display name for which the summary stat was taken for

---

**Note:** If the `deliveryServiceName` is equal to `all` it represents `summary_stats` across all delivery services within the given CDN

---

**statName** Stat name summary stat represents

**statValue** Summary stat value

**summaryTime** Timestamp of summary, in *Traffic Ops's Custom Date/Time Format*

**statDate** Date stat was taken, in YYYY-MM-DD format

## 551: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGS1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
→3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": [
  {
    "cdnName": "CDN-in-a-Box",
    "deliveryServiceName": "all",
    "statName": "daily_maxgbps",
    "statValue": 5,
    "summaryTime": "2019-11-19 00:04:06+00",
    "statDate": "2019-11-19"
  },
  {
    "cdnName": "CDN-in-a-Box",
    "deliveryServiceName": "all",
    "statName": "daily_maxgbps",
    "statValue": 3,
    "summaryTime": "2019-11-18 00:04:06+00",
    "statDate": "2019-11-18"
  },
  {
    "cdnName": "CDN-in-a-Box",
    "deliveryServiceName": "all",
    "statName": "daily_bytesserved",
    "statValue": 1000,
    "summaryTime": "2019-11-19 00:04:06+00",
    "statDate": "2019-11-19"
  }
]}

```

## Last Updated Summary Stat

**summaryTime** Timestamp of the last updated summary, in *Traffic Ops's Custom Date/Time Format*

### 552: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json

```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGS1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
→3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": {
  "summaryTime": "2019-11-19 00:04:06+00"
}}
```

## POST

Post a stats summary for a given stat.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

**cdnName** The CDN name for which the summary stat was taken for

---

**Note:** If the `cdn` is equal to `all` it represents `summary_stats` across all delivery services across all CDNs

---

**deliveryServiceName** The *Delivery Service* display name for which the summary stat was taken for

---

**Note:** If the `deliveryServiceName` is equal to `all` it represents `summary_stats` across all delivery services within the given CDN

---

**statName** Stat name summary stat represents

**statValue** Summary stat value

**summaryTime** Timestamp of summary, in *Traffic Ops's Custom Date/Time Format*

**statDate** Date stat was taken, in YYYY-MM-DD format

**Note:** statName, statValue and summaryTime are required. If cdnName and deliveryServiceName are not given they will default to all.

#### 553: Request Example

```
POST /api/3.0/stats_summary HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 113
Content-Type: application/json

{
  "cdnName": "CDN-in-a-Box",
  "deliveryServiceName": "all",
  "statName": "daily_maxgbps",
  "statValue": 10,
  "summaryTime": "2019-12-05 00:03:57+00",
  "statDate": "2019-12-05"
}
```

## Response Structure

#### 554: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳ezxk+iP7o7KE7zpWmGc0j8nz5k+1wAzY0HiNiA2xswTQrt+N+6CgQqUV2r9G1HAsPNr0HF2PhYs/
  ↳Xr7DrY0w0A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 02:14:45 GMT
Content-Length: 97

{ "alerts": [
  {
```

(continues on next page)

(continued from previous page)

```

    "text": "Stats Summary was successfully created",
    "level": "success"
  }
}
```

## statuses

### GET

Retrieves a list of all server *Statuses*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 241: Request Query Parameters

Name	Re- quired	Description
de- scrip- tion	no	Return only <i>Statuses</i> with this <i>exact</i> description
id	no	Return only the <i>Status</i> with this integral, unique identifier
name	no	Return only <i>Statuses</i> with this name
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
off- set	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n</i> <sup>th</sup> page of results, where “ <i>n</i> ” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

### 555: Request Example

```

GET /api/3.0/statuses?name=REPORTED HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- description** A short description of the status
- id** The integral, unique identifier of this status
- lastUpdated** The date and time at which this status was last modified, in *Traffic Ops's Custom Date/Time Format*
- name** The name of the status

### 556: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGS1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
    ↳3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": [
  {
    "description": "Server is online and reported in the health
    ↳protocol.",
    "id": 3,
    "lastUpdated": "2018-12-10 19:11:17+00",
    "name": "REPORTED"
  }
]}
```

## steering

### GET

Gets a list of all *Steering Targets* in the Traffic Ops database.

**Auth. Required** Yes

**Roles Required** “Portal”, “Steering”, “Federation”, “operations” or “admin”

**Response Type** Array

### Request Structure

No parameters available.

#### 557: Request Example

```
GET /api/3.0/steering HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**clientSteering** Whether this is a *client steering* Delivery Service.

**targets** The delivery services that the *Steering Delivery Service* targets.

**order** If this is a *STEERING\_ORDER* target, this is the value of the order. Otherwise, 0.

**weight** If this is a *STEERING\_WEIGHT* target, this is the value of the weight. Otherwise, 0.

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**filters** Filters of type *STEERING\_REGEXP* that exist on either of the targets.

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**pattern** A regular expression - the use of this pattern is dependent on the *type* field (backslashes are escaped)



558: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 18:56:57
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳hcJa4xVLDx7bxBmoSjYo5YUwdSBWQr9GqlqRYrc6ZU7LeenjiV3go22YlIHt/
    ↳GtjLcHQjJ5DulKRhdsvFMq7Fng==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 17:56:57 GMT
Content-Length: 167
```

```
{
  "response": [
    {
      "deliveryService": "steering1",
      "clientSteering": true,
      "targets": [
        {
          "order": 0,
          "weight": 1,
          "deliveryService": "demo1"
        },
        {
          "order": 0,
          "weight": 2,
          "deliveryService": "demo2"
        }
      ],
      "filters": [
        {
          "deliveryService": "demo1",
          "pattern": ".*\\.demo1\\.\\.*)"
        },
        {
          "deliveryService": "demo2",
          "pattern": ".*\\.demo2*\\.\\.*)"
        }
      ]
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
]
}
```

## steering/{ID}/targets

### GET

Get all targets for a steering *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 242: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i> for which targets shall be listed

Table 243: Request Query Parameters

Name	Description
target	Return only the target mappings that target the <i>Delivery Service</i> identified by this integral, unique identifier
orderBy	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

## 559: Request Structure

```
GET /api/3.0/steering/2/targets?target=1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*
- deliveryServiceId** An integral, unique identifier for the steering *Delivery Service*
- target** A string that is the *xml\_id* of this target *Delivery Service*
- targetId** An integral, unique identifier for this target *Delivery Service*
- type** The steering type of this target *Delivery Service*. This should be one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT
- typeId** An integral, unique identifier for the *steering type* of this target *Delivery Service*
- value** The 'weight', 'order', 'geo\_order' or 'geo\_weight' attributed to this steering target as an integer

### 560: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳utlJK4oYS2l6Ff7NzAqRuQeMEtaZyn3rM3Nlux2XgTLxvSyslHy0mJrwDExSU05gVMdrgYCLZrZEvPH1EL
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 14:09:23 GMT
Content-Length: 130

{ "response": [
  {
    "deliveryService": "test",
    "deliveryServiceId": 2,
    "target": "demo1",
    "targetId": 1,
    "type": "STEERING_ORDER",
    "typeId": 44,
```

(continues on next page)

(continued from previous page)

```

    "value": 100
  }
}]

```

## POST

Create a steering target.

**Auth. Required** Yes

**Roles Required** Portal, Steering, Federation, “operations” or “admin”

**Response Type** Object

## Request Structure

Table 244: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i> to which a target shall be added

**targetId** The integral, unique identifier of a *Delivery Service* which shall be a new steering target for the *Delivery Service* identified by the ID path parameter

**typeId** The integral, unique identifier of the steering type of the new target *Delivery Service*. This should be corresponding to one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT

**value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ which shall be attributed to the new target *Delivery Service*

## 561: Request Example

```

POST /api/3.0/steering/2/targets HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 43
Content-Type: application/json

{
  "targetId": 1,
  "value": 100,
  "typeId": 43
}

```

## Response Structure

- deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*
- deliveryServiceId** An integral, unique identifier for the steering *Delivery Service*
- target** A string that is the *xml\_id* of this target *Delivery Service*
- targetId** An integral, unique identifier for this target *Delivery Service*
- type** The steering type of this target *Delivery Service*. This should be one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT
- typeId** An integral, unique identifier for the *steering type* of this target *Delivery Service*
- value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ attributed to this steering target as an integer

### 562: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳+dTvfzrnOhdWAOMmY28r0+gFV5z+3aABI2FfAMziTYcU+pZrDanrJzMXpKWIL5Q/
    ↳oCUBZpJDRt9hRCFkT4oGYw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 21:22:17 GMT
Content-Length: 196

{ "alerts": [
  {
    "text": "steeringtarget was created.",
    "level": "success"
  }
],
"response": {
  "deliveryService": "test",
  "deliveryServiceId": 2,
  "target": "demo1",
  "targetId": 1,
  "type": "HTTP",
```

(continues on next page)

(continued from previous page)

```
"typeId": 1,
"value": 100
}}
```

`steering/{ID}/targets/{targetID}`

**PUT**

Updates a steering target.

**Auth. Required** Yes

**Roles Required** Portal, Steering, Federation, “operations” or “admin”

**Response Type** Object

## Request Structure

Table 245: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i>
targetID	The integral, unique identifier of a <i>Delivery Service</i> which is a target of the <i>Delivery Service</i> identified by ID

**typeId** The integral, unique identifier of the *steering type* of the target *Delivery Service*. This should be corresponding to one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT

**value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ which shall be attributed to the target *Delivery Service*

## 563: Request Example

```
PUT /api/3.0/steering/2/targets/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 26
Content-Type: application/json

{
  "value": 1,
  "typeId": 43
}
```

## Response Structure

- deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*
- deliveryServiceId** An integral, unique identifier for the steering *Delivery Service*
- target** A string that is the *xml\_id* of this target *Delivery Service*
- targetId** An integral, unique identifier for this target *Delivery Service*
- type** The steering type of this target *Delivery Service*
- typeId** An integral, unique identifier for the *steering type* of this target *Delivery Service*
- value** The ‘weight’ attributed to this steering target as an integer

### 564: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→AfXsIRzdtU3HZYkr93qBMVTZRJ5oTF2u5sKYnd+DSqxZ+RQxY6vXtCupnnXCf9dxMt5QXRW1EFOW/
→FBG6lFrTg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 14:34:22 GMT
Content-Length: 194

{ "alerts": [
  {
    "text": "steeringtarget was updated.",
    "level": "success"
  }
],
"response": {
  "deliveryService": "test",
  "deliveryServiceId": 2,
  "target": "demo1",
  "targetId": 1,
  "type": "HTTP",
  "typeId": 1,
  "value": 1
}}
```

## DELETE

Removes a specific target mapping from a specific *Delivery Service*

**Auth. Required** Yes

**Roles Required** Portal, Steering, Federation, “operations” or “admin”

**Response Type** undefined

## Request Structure

Table 246: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i> - a target of which shall be deleted
targetID	The integral, unique identifier of a <i>Delivery Service</i> which is a target to be removed of the <i>Delivery Service</i> identified by ID

### 565: Request Example

```
DELETE /api/3.0/steering/2/targets/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 566: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳N6h8Kl7uQveqpTc3fmKXFDY2yYe5smApNcaTow4ab0DHGFdJfqQh89I4nvvaXvmVNHxVAqX3UE/
    ↳6blb08/9Xqg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 14:42:54 GMT
```

(continues on next page)



(continued from previous page)

Content-Length: 69

```
{ "alerts": [
  {
    "text": "steeringtarget was deleted.",
    "level": "success"
  }
]}
```

## system/info

GET

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

No parameters available.

## Response Structure

**parameters** An object containing information about the Traffic Ops server

---

**Note:** These are all the *Parameters* in *The GLOBAL Profile*, so the keys below are merely those present by default required for Traffic Control to operate

---

**default\_geo\_miss\_latitude** The default latitude used when geographic lookup of an IP address fails

**default\_geo\_miss\_longitude** The default longitude used when geographic lookup of an IP address fails

**tm.instance\_name** The name of the Traffic Ops instance; typically used when multiple instances are active

**tm.toolname** The name of the Traffic Ops tool (usually “Traffic Ops”) - used in several API endpoints and written in comment headers on most Apache Traffic Server (ATS) configuration files generated by Traffic Ops

**tm.url** The URL for this Traffic Ops instance ) - used in several API endpoints and written in comment headers on most Apache Traffic Server (ATS) configuration files generated by Traffic Ops

**use\_reval\_pending** A string containing an integer which represents a boolean value (hold your applause); one of:

“0” Do not use pending revalidations - this effectively prohibits the use of “Content Invalidation Jobs”

“1” Use pending revalidations - this effectively enables the use of “Content Invalidation Jobs”

**use\_tenancy** A string containing an integer which represents a boolean value; one of:

“0” Do not use tenancy - this effectively disables all *\*tenant\** endpoints and removes tenancy restrictions on origins and *Delivery Services*

“1” Use tenancy - this effectively enables all *\*tenant\** endpoints and enforces tenancy restrictions on origins and *Delivery Services*

#### 567: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ObxOXk1jrC1/
→JtrqElUICceyx9iJKJxZydEIHvAU7khTTQwt0QGvS04ELdkdrbu3ctFo3pf3NAMAmm9tAkNokg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 19:06:01 GMT
Content-Length: 285

{ "response": {
  "parameters": {
    "default_geo_miss_latitude": "0",
    "default_geo_miss_longitude": "-1",
    "tm.instance_name": "CDN-In-A-Box",
    "tm.logourl": "/images/tc_logo.png",
    "tm.toolname": "Traffic Ops",
    "tm.url": "https://trafficops.infra.ciab.test:443/",
    "use_reval_pending": "0",
```

(continues on next page)

(continued from previous page)

```

    "use_tenancy": "1"
  }
}
```

## tenants

### GET

Get all requested *Tenants*.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 247: Request Query Parameters

Name	Description
ac- tive	If true, return only active <i>Tenants</i> ; if false return only inactive <i>Tenants</i>
id	Return only <i>Tenants</i> with this integral, unique identifier
name	Return only <i>Tenants</i> with this name
or- derby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## 568: Request Example

```

GET /api/3.0/tenants?name=root HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**active** A boolean which indicates whether or not the *Tenant* is active

**id** The integral, unique identifier of this *Tenant*

**name** This *Tenant*'s name

**parentId** The integral, unique identifier of this *Tenant*'s parent

**parentName** The name of the parent of this *Tenant*

### 569: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Yzr6TfhxgpZ3pbrrr4TRG4wC3PlnHDDzgs2igtz/
    ↳1ppLSy2MzugqaGW4y5yzwzl5T3+7q6HWej7GQZt1XIVeZQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 19:57:58 GMT
Content-Length: 106

{ "response": [
  {
    "id": 1,
    "name": "root",
    "active": true,
    "lastUpdated": "2018-12-10 19:11:17+00",
    "parentId": null
  }
]}
```

## POST

Create a new tenant.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**active** An optional boolean - default: false - which indicates whether or not the tenant shall be immediately active

**name** The name of the tenant

**parentId** The integral, unique identifier of the parent of this tenant

### 570: Request Example

```
POST /api/3.0/tenants HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 48
Content-Type: application/json

{
  "active": true,
  "name": "test",
  "parentId": 1
}
```

## Response Structure

**active** A boolean which indicates whether or not the tenant is active

**id** The integral, unique identifier of this tenant

**name** This tenant's name

**parentId** The integral, unique identifier of this tenant's parent

### 571: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ysdopC//
  ↳JQI79BRUa61s6M2HzHxYHpo5Rdcuau0oqCYxiV0oUhNZfOVydVkv8zDN2qA374XKnym4kWj3VzQIXg==
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

(continued from previous page)

```
Date: Tue, 11 Dec 2018 19:37:16 GMT
Content-Length: 162

{ "alerts": [
  {
    "text": "tenant was created.",
    "level": "success"
  }
],
"response": {
  "id": 9,
  "name": "test",
  "active": true,
  "lastUpdated": "2018-12-11 19:37:16+00",
  "parentId": 1
}}
```

`tenants/{ID}`

**PUT**

Updates a specific tenant.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 248: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the tenant being modified

**active** An optional boolean - default: `false` - which indicates whether or not the tenant shall be immediately active

**name** The name of the tenant

**parentId** The integral, unique identifier of the parent of this tenant

### 572: Request Example

```
PUT /api/3.0/tenants/9 HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)

(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 59
Content-Type: application/json

{
  "active": true,
  "name": "quest",
  "parentId": 3
}
```

## Response Structure

**active** A boolean which indicates whether or not the tenant is active

**id** The integral, unique identifier of this tenant

**name** This tenant's name

**parentId** The integral, unique identifier of this tenant's parent

### 573: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 5soYQFrG2x5ZJ1e5UZIOLUv/
  ↳928qyd2Bfgw21Wv85rqjLpyeT3djkrVD1/xpKConulNrZs2czJKrrwZA7X61w==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 20:30:54 GMT
Content-Length: 163

{ "alerts": [
  {
    "text": "tenant was updated.",
    "level": "success"
  }
],
"response": {
  "id": 9,
```

(continues on next page)

(continued from previous page)

```
"name": "quest",
"active": true,
"lastUpdated": "2018-12-11 20:30:54+00",
"parentId": 3
}}
```

## DELETE

Deletes a specific tenant.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 249: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the tenant being deleted

### 574: Request Example

```
DELETE /api/3.0/tenants/9 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 575: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
```

(continues on next page)



(continued from previous page)

```
Whole-Content-Sha512: KU0XIbFoD0Cy06kzH2Gl59pBqie/
→TEFJgh33mssGNwXJZlRkTLaSTHT8Df4X+pOs7UauZH10akGvaA0UTiN/vg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 20:40:31 GMT
Content-Length: 61

{ "alerts": [
  {
    "text": "tenant was deleted.",
    "level": "success"
  }
]}
```

## topologies

### GET

Retrieves *Topologies*.

**Auth. Required** Yes

**Roles Required** “read-only”

**Response Type** Array

## Request Structure

Table 250: Request Query Parameters

Name	Required	Description
name	no	Return the <i>Topology</i> with this name

### 576: Request Example

```
GET /api/3.0/topologies HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- description** A short sentence that describes the *Topology*.
- lastUpdated** The date and time at which this *Topology* was last updated, in ISO-like format
- name** The name of the *Topology*. This can only be letters, numbers, and dashes.
- nodes** An array of nodes in the *Topology*
  - cachegroup** The name of a *Cache Group*
  - parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

### 577: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:22:32
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳1F4MCJCinuQWz0f1LAAZBrzbuPVsHrNn2BtTozRZojEjGpm76IsXBQK5Q0wSwBoHac+D0C1Z3p7M8kdjc
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:22:32 GMT
Content-Length: 205

{
  "response": [
    {
      "description": "This is my topology",
      "name": "my-topology",
```

(continues on next page)

(continued from previous page)

```

"nodes": [
  {
    "cachegroup": "edge1",
    "parents": [
      7
    ]
  },
  {
    "cachegroup": "edge2",
    "parents": [
      7,
      8
    ]
  },
  {
    "cachegroup": "edge3",
    "parents": [
      8,
      9
    ]
  },
  {
    "cachegroup": "edge4",
    "parents": [
      9
    ]
  },
  {
    "cachegroup": "mid1",
    "parents": []
  },
  {
    "cachegroup": "mid2",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid3",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid4",
    "parents": [

```

(continues on next page)

(continued from previous page)

```

        5
      ]
    },
    {
      "cachegroup": "mid5",
      "parents": [
        5,
        6
      ]
    },
    {
      "cachegroup": "mid6",
      "parents": [
        6
      ]
    }
  ],
  "lastUpdated": "2020-04-13 17:12:34+00"
}
]
}

```

## POST

Create a new *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

**description** A short sentence that describes the topology.

**name** The name of the topology. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group* with at least 1 server in it

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

578: Request Example

```
POST /api/3.0/topologies HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 924
Content-Type: application/json

{
  "name": "my-topology",
  "description": "This is my topology",
  "nodes": [
    {
      "cachegroup": "edge1",
      "parents": [
        7
      ]
    },
    {
      "cachegroup": "edge2",
      "parents": [
        7,
        8
      ]
    },
    {
      "cachegroup": "edge3",
      "parents": [
        8,
        9
      ]
    },
    {
      "cachegroup": "edge4",
      "parents": [
        9
      ]
    },
    {
      "cachegroup": "mid1",
      "parents": []
    },
    {
      "cachegroup": "mid2",
```

(continues on next page)

(continued from previous page)

```

        "parents": [
            4
        ]
    },
    {
        "cachegroup": "mid3",
        "parents": [
            4
        ]
    },
    {
        "cachegroup": "mid4",
        "parents": [
            5
        ]
    },
    {
        "cachegroup": "mid5",
        "parents": [
            5,
            6
        ]
    },
    {
        "cachegroup": "mid6",
        "parents": [
            6
        ]
    }
]
}

```

## Response Structure

**description** A short sentence that describes the topology.

**lastUpdated** The date and time at which this *Topology* was last updated, in ISO-like format

**name** The name of the topology. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group*

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

579: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:12:34
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ftNcDRjYCDMkQM+o/
    ↳szayKZriQZHGpcT0vNY0HpKgy88i0pXeEEeLGbUPh6LXtK7TvL76EgGECTzvCkcm+2LVA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:12:34 GMT
Content-Length: 239

{
  "alerts": [
    {
      "text": "topology was created.",
      "level": "success"
    }
  ],
  "response": {
    "description": "This is my topology",
    "name": "my-topology",
    "nodes": [
      {
        "cachegroup": "edge1",
        "parents": [
          7
        ]
      },
      {
        "cachegroup": "edge2",
        "parents": [
          7,
          8
        ]
      },
      {
        "cachegroup": "edge3",
        "parents": [
          8,
          9
        ]
      }
    ]
  }
}

```

(continues on next page)

(continued from previous page)

```

    ]
  },
  {
    "cachegroup": "edge4",
    "parents": [
      9
    ]
  },
  {
    "cachegroup": "mid1",
    "parents": []
  },
  {
    "cachegroup": "mid2",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid3",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid4",
    "parents": [
      5
    ]
  },
  {
    "cachegroup": "mid5",
    "parents": [
      5,
      6
    ]
  },
  {
    "cachegroup": "mid6",
    "parents": [
      6
    ]
  }
],
"lastUpdated": "2020-04-13 17:12:34+00"
}

```

(continues on next page)



(continued from previous page)

```
}

```

## PUT

Updates a specific *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 251: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Topology</i> to be updated

**description** A short sentence that describes the *Topology*.

**name** The name of the *Topology*. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group* with at least 1 server in it

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

### 580: Request Example

```
PUT /api/3.0/topologies?name=my-topology HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 853
Content-Type: application/json

{
  "name": "my-topology",
  "description": "The description is updated, too",
  "nodes": [
    {
      "cachegroup": "edge1",
```

(continues on next page)

(continued from previous page)

```

        "parents": [
            6
        ]
    },
    {
        "cachegroup": "edge2",
        "parents": [
            6,
            7
        ]
    },
    {
        "cachegroup": "edge3",
        "parents": [
            7,
            8
        ]
    },
    {
        "cachegroup": "edge4",
        "parents": [
            8
        ]
    },
    {
        "cachegroup": "mid2",
        "parents": []
    },
    {
        "cachegroup": "mid3",
        "parents": []
    },
    {
        "cachegroup": "mid4",
        "parents": [
            4
        ]
    },
    {
        "cachegroup": "mid5",
        "parents": [
            4,
            5
        ]
    },
    {

```

(continues on next page)

(continued from previous page)

```

        "cachegroup": "mid6",
        "parents": [
            5
        ]
    }
]
}

```

## Response Structure

**description** A short sentence that describes the *Topology*.

**lastUpdated** The date and time at which this *Topology* was last updated, in ISO-like format

**name** The name of the *Topology*. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group*

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

### 581: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:33:13.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: WV0tsoOVrEWcVjWM2TmT5DXy/
→a5Q0ygTZEQRhbKHHUmz9dgVLK1F5Joc9jtKA8gZu8/eM5+Tqqguh3mzrhAy/Q==
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:33:13 GMT
Content-Length: 237

{
  "alerts": [
    {
      "text": "topology was updated.",
      "level": "success"
    }
  ]
}

```

(continues on next page)

(continued from previous page)

```

],
"response": {
  "description": "The description is updated, too",
  "name": "my-topology",
  "nodes": [
    {
      "cachegroup": "edge1",
      "parents": [
        6
      ]
    },
    {
      "cachegroup": "edge2",
      "parents": [
        6,
        7
      ]
    },
    {
      "cachegroup": "edge3",
      "parents": [
        7,
        8
      ]
    },
    {
      "cachegroup": "edge4",
      "parents": [
        8
      ]
    },
    {
      "cachegroup": "mid2",
      "parents": []
    },
    {
      "cachegroup": "mid3",
      "parents": []
    },
    {
      "cachegroup": "mid4",
      "parents": [
        4
      ]
    }
  ]
}

```

(continues on next page)

(continued from previous page)

```

        "cachegroup": "mid5",
        "parents": [
            4,
            5
        ]
    },
    {
        "cachegroup": "mid6",
        "parents": [
            5
        ]
    }
],
"lastUpdated": "2020-04-13 17:33:13+00"
}

```

## DELETE

Deletes a specific *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

## Request Structure

Table 252: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Topology</i> to be deleted

## 582: Request Example

```

DELETE /api/3.0/topologies?name=my-topology HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0

```

## Response Structure

### 583: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:35:32
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: yErJobzG9IA0khvqZQK+Yi7X4pFVvOqxn6PjrdzN5DnKVm/
    ↳K8Kka3REul1XmKJnMXVRY8RayoEVGDm16mBFfe4Q==
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:35:32 GMT
Content-Length: 87

{
  "alerts": [
    {
      "text": "topology was deleted.",
      "level": "success"
    }
  ]
}
```

`topologies/{{name}}/queue_update`

**POST**

*Queue* or “dequeue” updates for all servers assigned to the *Cache Groups* in a specific *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 253: Request Path Parameters

Name	Description
name	The name of the <i>Topology</i> on which to queue or dequeue updates.

**action** One of “queue” or “dequeue” as appropriate

**cdnId** The integral, unique identifier for the CDN on which to (de)queue updates

### 584: Request Example

```
POST /api/3.0/topologies/demo1-top/queue_update HTTP/1.1
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 28

{
  "action": "queue",
  "cdnId": 1
}
```

## Response Structure

**action** The action processed, either "queue" or "dequeue"

**cdnId** The CDN ID on which *Queue Updates* was performed or cleared

**topology** The name of the *Topology* on which *Queue Updates* was performed or cleared

### 585: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 08 Sep 2020 17:35:42.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512:↵
↵nmu3TMVml1cHeEstLuiqPsEypNV2jcs5PyrqsqJKkexxb8N7qk84AWzTJWUpsfdVWrj/
↵JzRiCPGJS4hw0phQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 08 Sep 2020 16:35:42 GMT
Content-Length: 79

{
  "response": {
    "action": "queue",
    "cdnId": 1,
    "topology": "demo1-top"
  }
}
```

## types

### GET

Retrieves all of the *Types* of things configured in Traffic Ops. Yes, that is as specific as a description of a ‘type’ can be.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 254: Request Query Parameters

Name	Re-quired	Description
id	no	Return only the type that is identified by this integral, unique identifier
name	no	Return only types with this name
useInTable	no	Return only types that are used to identify the type of the object stored in the Traffic Ops database table that has this name

### 586: Request Structure

```
GET /api/3.0/types?name=TC_LOC HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)



(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- description** A short description of this type
- id** An integral, unique identifier for this type
- lastUpdated** The date and time at which this type was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of this type
- useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type

### 587: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpgT3YRyKJ/
→p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 12 Dec 2018 22:59:22 GMT
Content-Length: 168

{ "response": [
  {
    "id": 48,
    "lastUpdated": "2018-12-12 16:26:41+00",
    "name": "TC_LOC",
    "description": "Location for Traffic Control Component Servers
→",
    "useInTable": "cachegroup"
  }
]}
```

### POST

Creates a type

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

### Request Structure

**description** A short description of this type

**name** The name of this type

**useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type.

---

**Note:** The only useInTable value that is allowed to be created dynamically is ‘server’

---

#### 588: Request Structure

```
POST /api/3.0/type HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 67
Content-Type: application/json

{
  "name": "Example01",
  "description": "Example",
  "useInTable": "server"
}
```

### Response Structure

**description** A short description of this type

**id** An integral, unique identifier for this type

**lastUpdated** The date and time at which this type was last updated, in *Traffic Ops’s Custom Date/Time Format*

**name** The name of this type

**useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type

#### 589: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpgT3YRyKJ/
→p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 Feb 2020 18:58:41 GMT
Content-Length: 171

{
  "alerts": [
    {
      "text": "type was created.",
      "level": "success"
    }
  ],
  "response": [
    {
      "id": 3004,
      "lastUpdated": "2020-02-26 18:58:41+00",
      "name": "Example01",
      "description": "Example"
      "useInTable": "server"
    }
  ]
}
```

types/{ID}

PUT

Updates a type

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 255: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the type being updated

**description** A short description of this type

**name** The name of this type

**useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type.

---

**Note:** Only types with useInTable set to ‘server’ are allowed to be updated.

---

### 590: Request Structure

```
PUT /api/3.0/type/3004 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 68
Content-Type: application/json

{
  "name": "Example01",
  "description": "Example2",
  "useInTable": "server"
}
```

## Response Structure

**description** A short description of this type

**id** An integral, unique identifier for this type

**lastUpdated** The date and time at which this type was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** The name of this type

**useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type

## 591: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo80rCu79Tz9xpgT3YRyKJ/
    ↳p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 Feb 2020 18:58:41 GMT
Content-Length: 172

{
  "alerts": [
    {
      "text": "type was updated.",
      "level": "success"
    }
  ],
  "response": [
    {
      "id": 3004,
      "lastUpdated": "2020-02-26 18:58:41+00",
      "name": "Example02",
      "description": "Example"
      "useInTable": "server"
    }
  ]
}
```

## DELETE

Deletes a type

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 256: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the type being deleted

**Note:** Only types with useInTable set to “server” are allowed to be deleted.

### 592: Request Structure

```
DELETE /api/3.0/type/3004 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

### 593: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpgT3YRyKJ/
    ↳p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 Feb 2020 18:58:41 GMT
Content-Length: 84

{
  "alerts": [
    {
      "text": "type was deleted.",
      "level": "success"
    }
  ],
}
```

`user/current`

GET

**Caution:** As a username is needed to log in, any administrator or application must necessarily know the current username at any given time. Thus it's generally better to use the username query parameter of a GET request to `users` instead.

Retrieves the details of the authenticated user.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

No parameters available.

## Response Structure

**addressLine1** The user's address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides

**email** The user's email address

**fullName** The user's full name, e.g. "John Quincy Adams"

**gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user

**id** An integral, unique identifier for this user

**lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*

**newUser** A meta field with no apparent purpose that is usually null unless explicitly set during creation or modification of a user via some API endpoint

**phoneNumber** The user's phone number

**postalCode** The postal code of the area in which the user resides

**publicSshKey** The user's public key used for the SSH protocol

- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be null
- role** The integral, unique identifier of the highest-privilege *Role* assigned to this user
- rolename** The name of the highest-privilege *Role* assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the *Tenant* to which this user belongs
- tenantId** The integral, unique identifier of the *Tenant* to which this user belongs
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user
- username** The user's username

#### 594: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳HQwu9FxFyinxSVFK5+wpEhSxU60KbqXuokFbMZ30oerOoM5ZpWpglsHz7mRch8VAw0dzwsJzpPJivj07R
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 15:14:45 GMT
Content-Length: 382

{ "response": {
  "username": "admin",
  "localUser": true,
  "addressLine1": null,
  "addressLine2": null,
  "city": null,
  "company": null,
  "country": null,
  "email": null,
  "fullName": "admin",
  "gid": null,
  "id": 2,
  "newUser": false,
  "phoneNumber": null,
```

(continues on next page)



(continued from previous page)

```
"postalCode": null,
"publicSshKey": null,
"role": 1,
"rolename": "admin",
"stateOrProvince": null,
"tenant": "root",
"tenantId": 1,
"uid": null,
"lastUpdated": "2018-12-12 16:26:32+00"
}}
```

## PUT

**Warning:** Assuming the current user’s integral, unique identifier is known, it’s generally better to use the PUT method of the *users* instead.

**Warning:** Users that login via LDAP pass-back cannot be modified

Updates the date for the authenticated user.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Object

## Request Structure

**user** The entire request must be inside a top-level “user” key for legacy reasons

**addressLine1** The user’s address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**confirmLocalPasswd** An optional ‘confirm’ field in a new user’s password specification. This has no known effect and in fact *doesn’t even need to match* localPasswd

**country** The name of the country wherein the user resides

**email** The user's email address - cannot be an empty string<sup>1</sup>. The given email is validated (circuitously) by [GitHub user asaskevich's regular expression](#). Note that it can't actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user's full name, e.g. "John Quincy Adams"

**gid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - please don't use this

**id** The user's integral, unique, identifier - this cannot be changed<sup>Page 862, 1</sup>

**localPasswd** Optionally, the user's password. This should never be given if it will not be changed. An empty string or null can be used to explicitly specify no change.

**phoneNumber** The user's phone number

**postalCode** The user's postal code

**publicSshKey** The user's public encryption key used for the SSH protocol

**role** The integral, unique identifier of the highest permission [Role](#) which will be permitted to the user - this cannot be altered from the user's current [Role](#)<sup>Page 862, 1</sup>

**stateOrProvince** The state or province in which the user resides

**tenantId** The integral, unique identifier of the [Tenant](#) to which the new user shall belong<sup>2Page 862, 1</sup>

**uid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - please don't use this

**username** The user's new username<sup>Page 862, 1</sup>

### 595: Request Example

```
PUT /api/3.0/user/current HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 465
Content-Type: application/json
```

(continues on next page)

---

<sup>1</sup> This field cannot be null.

<sup>2</sup> This endpoint respects tenancy; a user cannot assign itself to a [Tenant](#) that is not the same [Tenant](#) to which it was previously assigned or a descendant thereof.

(continued from previous page)

```
{ "user": {
  "addressLine1": null,
  "addressLine2": null,
  "city": null,
  "company": null,
  "country": null,
  "email": "admin@infra.trafficops.ciab.test",
  "fullName": "admin",
  "gid": null,
  "id": 2,
  "phoneNumber": null,
  "postalCode": null,
  "publicSshKey": null,
  "role": 1,
  "stateOrProvince": null,
  "tenantId": 1,
  "uid": null,
  "username": "admin"
}}
```

## Response Structure

**addressLine1** The user’s address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides

**email** The user’s email address validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user’s full name, e.g. “John Quincy Adams”

**gid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user

**id** An integral, unique identifier for this user

**lastUpdated** The date and time at which the user was last modified, in *Traffic Ops’s Custom Date/Time Format*

**newUser** A meta field with no apparent purpose

**phoneNumber** The user’s phone number

- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be null
- role** The integral, unique identifier of the highest-privilege *Role* assigned to this user
- rolename** The name of the highest-privilege *Role* assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the *Tenant* to which this user belongs
- tenantId** The integral, unique identifier of the *Tenant* to which this user belongs
- uid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user
- username** The user's username

#### 596: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Date: Thu, 13 Dec 2018 21:05:49 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: sHFqZQ4Cv7IIWaIejoAvM2Fr/HSupcX3D16KU/
    ↳etjw+4jcK9EME3Bq5ohLC+eQ52BDCKW2Ra+AC3TfFtworJww==
Content-Length: 478

{ "alerts": [
  {
    "text": "User profile was successfully updated",
    "level": "success"
  }
],
"response": {
  "addressLine1": null,
  "addressLine2": null,
  "city": null,

```

(continues on next page)

(continued from previous page)

```

"company": null,
"country": null,
"email": "admin@infra.trafficops.ciab.test",
"fullName": null,
"gid": null,
"id": 2,
"lastUpdated": "2019-10-08 20:14:25+00",
"newUser": false,
"phoneNumber": null,
"postalCode": null,
"publicSshKey": null,
"registrationSent": null,
"role": 1,
"roleName": "admin",
"stateOrProvince": null,
"tenant": "root",
"tenantId": 1,
"uid": null,
"username": "admin"
}}
```

## user/login

### POST

Authentication of a user using username and password. Traffic Ops will send back a session cookie.

**Auth. Required** No

**Roles Required** None

**Response Type** undefined

### Request Structure

**p** Password

**u** Username

#### 597: Request Example

```

POST /api/3.0/user/login HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

(continues on next page)

(continued from previous page)

```
Cookie: mojolicious=...
Content-Length: 26
Content-Type: application/json

{
  "u": "admin",
  "p": "twelve"
}
```

## Response Structure

### 598: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Ud06T3tMNctnVusDXzRjVwwYOnD7jmnBzPEB9Pv0t2bHajTv3SKTPiIZjDzvhU6EX4p+JoG4fA5wlhgxp
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 15:21:33 GMT
Content-Length: 65

{ "alerts": [
  {
    "text": "Successfully logged in.",
    "level": "success"
  }
]}
```

## user/login/oauth

### POST

Authentication of a user by exchanging a code for an encrypted JSON Web Token from an OAuth service. Traffic Ops will POST to the `authCodeTokenUrl` to exchange the code for an encrypted JSON Web Token. It will then decode and validate the token, validate the key set domain, and send back a session cookie.

**Auth. Required** No

**Roles Required** None

**Response Type** undefined

## Request Structure

**authCodeTokenUrl** URL for code-to-token conversion

**code** Code

**clientId** Client Id

**redirectUri** Redirect URI

599: Request Example

```
POST /api/3.0/user/login/oauth HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 26
Content-Type: application/json

{
  "authCodeTokenUrl": "https://url-to-convert-code-to-token.example.
  ↪com",
  "code": "AbCd123",
  "clientId": "oauthClientId",
  "redirectUri": "https://traffic-portal.example.com/sso"
}
```

## Response Structure

600: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, ↪
  ↪Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54 ↪
  ↪GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ↪
  ↪Ud06T3tMNctnVusDXzRjVwwYOnD7jmnBzPEB9PvOt2bHajTv3SKTPiIZidzyhU6EX4p ↪
  ↪loG4fA5wlhgxp (continues on next page)
```

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 15:21:33 GMT
Content-Length: 65

{ "alerts": [
  {
    "text": "Successfully logged in.",
    "level": "success"
  }
]}
```

## user/login/token

### POST

Authentication of a user using a token. Normally, the token is obtained via a call to either *user/reset\_password* or *users/register*.

**Auth. Required** No

**Roles Required** None

**Response Type** undefined

### Request Structure

**t** A UUID (Universal Unique Identifier) generated for the user.

---

#### Implementation Detail

Though not strictly necessary for authentication provided direct database access, the tokens generated for use with this endpoint are compliant with [RFC 4122](#).

---

#### 601: Request Example

```
POST /api/3.0/user/login/token HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 44
Content-Type: application/json

{
```

(continues on next page)



(continued from previous page)

```
"t": "18EE200C-FF24-11E8-BF01-870C776752A3"
}
```

## Response Structure

### 602: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: FuS3TkVosxHtpxRGMJ2on+WnFdYTNSPjxz/Gh1iT4UCJ2/
    ↳P0twUbAGQ3tTx9EfGiAzg9CNQiVUFGnYjJZ6NCpg==
X-Server-Name: traffic_ops_golang/
Date: Fri, 20 Sep 2019 15:02:43 GMT
Content-Length: 66

{ "alerts": [
  {
    "text": "Successfully logged in.",
    "level": "success"
  }
]}
```

## user/logout

### POST

User logout. Invalidates the session cookie of the currently logged-in user.

**Auth. Required** Yes

**Roles Required** None

**Response Type** undefined

### Request Structure

No parameters available

### Response Structure

#### 603: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 21:25:36 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 6KEdr1ZC512zk0l03KwvQE0L7qrJ/
    ↳+ek6ztymkYy9p8gdPUyYyzGEAJ/Ldb8GY0UBFYmgqeZ3yWHvTcEs0QMiw==
Content-Length: 61

{ "alerts": [
  {
    "level": "success",
    "text": "You are logged out."
  }
]}
```

### user/reset\_password

#### POST

Sends an email to reset a user's password.

**Auth. Required** No

**Roles Required** None

**Response Type** undefined

## Request Structure

**email** The email address of the user to initiate password reset

### 604: Request Example

```
POST /api/3.0/user/reset_password HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 35
Content-Type: application/json

{
  "email": "test@example.com"
}
```

## Response Structure

### 605: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 22:11:53 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→lKWwVYbgKklk7bljnQJZwWV5bljIk+GkooD6SAC3CSexVKvfbL9dgL5iBc/
→BNNRk2pIU5I/1GglcdLrXsF1ZA==
Content-Length: 109

{ "alerts": [
  {
    "level": "success",
    "text": "Successfully sent password reset to email
→'test@example.com'"
  }
]
```

(continues on next page)

(continued from previous page)

```
]]}
```

## users

### GET

Retrieves all requested users.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Response Type** Array

## Request Structure

Table 257: Request Query Parameters

Name	Re-quired	Description
id	no	Return only the user identified by this integral, unique identifier
tenant	no	Return only users belonging to the <i>Tenant</i> identified by tenant name
role	no	Return only users belonging to the <i>Role</i> identified by role name
username	no	Return only the user with this username
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <sup>n</sup> th page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## 606: Request Example

```
GET /api/3.0/users?username=admin HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

<sup>1</sup> While no roles are required, this endpoint does respect tenancy. A user will only be able to see, create, delete or modify other users belonging to the same tenant, or its descendants.

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- addressLine1** The user's address - including street name and number
- addressLine2** An additional address field for e.g. apartment number
- city** The name of the city wherein the user resides
- company** The name of the company for which the user works
- country** The name of the country wherein the user resides
- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- id** An integral, unique identifier for this user
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The integral, unique identifier of the highest-privilege role assigned to this user
- rolename** The name of the highest-privilege role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`
- username** The user's username

## 607: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳YBJLN8NbOxOvECe1RGtcwCzIPDhyhLpW56nTJHQM5WI2WUDe2mAKREpaEE72nzrfBliq1GABwJlxsxq20dl
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 01:03:53 GMT
Content-Length: 391

{ "response": [
  {
    "username": "admin",
    "registrationSent": null,
    "addressLine1": null,
    "addressLine2": null,
    "city": null,
    "company": null,
    "country": null,
    "email": null,
    "fullName": null,
    "gid": null,
    "id": 2,
    "newUser": false,
    "phoneNumber": null,
    "postalCode": null,
    "publicSshKey": null,
    "role": 1,
    "rolename": "admin",
    "stateOrProvince": null,
    "tenant": "root",
    "tenantId": 1,
    "uid": null,
    "lastUpdated": "2018-12-12 16:26:32+00"
  }
]}

```

## POST

Creates a new user.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 964, 1</sup>

**Response Type** Object

## Request Structure

**addressLine1** An optional field which should contain the user’s address - including street name and number

**addressLine2** An optional field which should contain an additional address field for e.g. apartment number

**city** An optional field which should contain the name of the city wherein the user resides

**company** An optional field which should contain the name of the company for which the user works

**confirmLocalPasswd** The ‘confirm’ field in a new user’s password specification - must match `localPasswd`

**country** An optional field which should contain the name of the country wherein the user resides

**email** The user’s email address The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user’s full name, e.g. “John Quincy Adams”

**localPasswd** The user’s password

**newUser** An optional meta field with no apparent purpose - don’t use this

**phoneNumber** An optional field which should contain the user’s phone number

**postalCode** An optional field which should contain the user’s postal code

**publicSshKey** An optional field which should contain the user’s public encryption key used for the SSH protocol

**role** The number that corresponds to the highest permission role which will be permitted to the user

**stateOrProvince** An optional field which should contain the name of the state or province in which the user resides

**tenantId** The integral, unique identifier of the tenant to which the new user shall belong

---

**Note:** This field is optional if and only if tenancy is not enabled in Traffic Control

---

**username** The new user's username

### 608: Request Example

```
POST /api/3.0/users HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 304
Content-Type: application/json

{
  "username": "mike",
  "addressLine1": "22 Mike Wazowski You've Got Your Life Back Lane",
  "city": "Monstropolis",
  "company": "Monsters Inc.",
  "email": "mwazowski@minc.biz",
  "fullName": "Mike Wazowski",
  "localPasswd": "BFFsully",
  "confirmLocalPasswd": "BFFsully",
  "newUser": true,
  "role": 1,
  "tenantId": 1
}
```

## Response Structure

**addressLine1** The user's address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides

**email** The user's email address

**fullName** The user's full name, e.g. "John Quincy Adams"

**gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always null

**id** An integral, unique identifier for this user



- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually null unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be null
- role** The integral, unique identifier of the highest-privilege role assigned to this user
- rolename** The name of the highest-privilege role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always null
- username** The user's username

#### 609: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 02:28:27 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
    ↳vDqbaMvgeeoIds1czqvIWlyDG8WLnCCJdF14Ub05nsE+oJOakkyeZ8odf4d0Zjtqpk01hoVo14H2tjuWP
Content-Length: 520

{ "alerts": [
  {
    "level": "success",
```

(continues on next page)

(continued from previous page)

```
        "text": "User creation was successful."
      }
    ],
    "response": {
      "registrationSent": null,
      "email": "mwazowski@minc.biz",
      "tenantId": 1,
      "city": "Monstropolis",
      "tenant": "root",
      "id": 8,
      "company": null,
      "rolename": "admin",
      "phoneNumber": null,
      "country": null,
      "fullName": "Mike Wazowski",
      "publicSshKey": null,
      "uid": null,
      "stateOrProvince": null,
      "lastUpdated": null,
      "username": "mike",
      "newUser": false,
      "addressLine2": null,
      "role": 1,
      "addressLine1": "22 Mike Wazowski You've Got Your Life Back Lane",
      "postalCode": null,
      "gid": null
    }
  }
}
```

users/{{ID}}

GET

Retrieves a specific user.

**Auth. Required** Yes

**Roles Required** None

**Response Type** Array

## Request Structure

Table 258: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the user to be retrieved

### 610: Request Example

```
GET /api/3.0/users/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- addressLine1** The user’s address - including street name and number
- addressLine2** An additional address field for e.g. apartment number
- city** The name of the city wherein the user resides
- company** The name of the company for which the user works
- country** The name of the country wherein the user resides
- email** The user’s email address
- fullName** The user’s full name, e.g. “John Quincy Adams”
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- id** An integral, unique identifier for this user
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops’s Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user’s phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user’s public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The integral, unique identifier of the highest-privilege role assigned to this user

**rolename** The name of the highest-privilege role assigned to this user

**stateOrProvince** The name of the state or province where this user resides

**tenant** The name of the tenant to which this user belongs

**tenantId** The integral, unique identifier of the tenant to which this user belongs

**uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always null

**username** The user's username

#### 611: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳9vqUmt8fWEuDb+9LQJ4sGbbF4Z0a7uNyBNSWhyzAi3fBUZ5mGhd4Jx5IuSlEqiLZnYeViJJL8mpRortkH
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 17:46:00 GMT
Content-Length: 588

{ "response": [
  {
    "username": "admin",
    "registrationSent": null,
    "addressLine1": "not a real address",
    "addressLine2": "not a real address either",
    "city": "not a real city",
    "company": "not a real company",
    "country": "not a real country",
    "email": "not@real.email",
    "fullName": "Not a real Full Name",
    "gid": null,
    "id": 2,
    "newUser": false,
    "phoneNumber": "not a real phone number",
    "postalCode": "not a real postal code",
    "publicSshKey": "not a real ssh key",
    "role": 1,
    "rolename": "admin",
    "stateOrProvince": "not a real state or province",
```

(continues on next page)

(continued from previous page)

```

    "tenant": "root",
    "tenantId": 1,
    "uid": null,
    "lastUpdated": "2018-12-13 17:24:23+00"
  }
}]

```

## PUT

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

## Request Structure

Table 259: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the user to be modified

**addressLine1** An optional field which should contain the user’s address - including street name and number

**addressLine2** An optional field which should contain an additional address field for e.g. apartment number

**city** An optional field which should contain the name of the city wherein the user resides

**company** An optional field which should contain the name of the company for which the user works

**confirmLocalPasswd** The ‘confirm’ field in a new user’s password specification - must match `localPasswd`

**country** An optional field which should contain the name of the country wherein the user resides

**email** The user’s email address The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user’s full name, e.g. “John Quincy Adams”

**localPasswd** The user’s password

**newUser** An optional meta field with no apparent purpose - don’t use this

- phoneNumber** An optional field which should contain the user's phone number
- postalCode** An optional field which should contain the user's postal code
- publicSshKey** An optional field which should contain the user's public encryption key used for the SSH protocol
- role** The number that corresponds to the highest permission role which will be permitted to the user
- stateOrProvince** An optional field which should contain the name of the state or province in which the user resides
- tenantId** The integral, unique identifier of the tenant to which the new user shall belong

---

**Note:** This field is optional if and only if tenancy is not enabled in Traffic Control

---

**username** The new user's username

#### 612: Request Structure

```
PUT /api/3.0/users/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 458
Content-Type: application/json

{
  "addressLine1": "not a real address",
  "addressLine2": "not a real address either",
  "city": "not a real city",
  "company": "not a real company",
  "country": "not a real country",
  "email": "not@real.email",
  "fullName": "Not a real fullName",
  "phoneNumber": "not a real phone number",
  "postalCode": "not a real postal code",
  "publicSshKey": "not a real ssh key",
  "stateOrProvince": "not a real state or province",
  "tenantId": 1,
  "role": 1,
  "username": "admin"
}
```

## Response Structure

- addressLine1** The user's address - including street name and number
- addressLine2** An additional address field for e.g. apartment number
- city** The name of the city wherein the user resides
- company** The name of the company for which the user works
- country** The name of the country wherein the user resides
- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- id** An integral, unique identifier for this user
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The integral, unique identifier of the highest-privilege role assigned to this user
- rolename** The name of the highest-privilege role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`
- username** The user's username

### 613: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 17:24:23 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: QKvGSIwSdreMI/OdgWv9WQfI/
→C1JbXSoQGospTGfCVUJ32XNWMhmREGzWojWsilW8os8b14TGYeyMLUWunf2Ug==
Content-Length: 661

{ "alerts": [
  {
    "level": "success",
    "text": "User update was successful."
  }
],
"response": {
  "registrationSent": null,
  "email": "not@real.email",
  "tenantId": 1,
  "city": "not a real city",
  "tenant": "root",
  "id": 2,
  "company": "not a real company",
  "rolename": "admin",
  "phoneNumber": "not a real phone number",
  "country": "not a real country",
  "fullName": "Not a real Full Name",
  "publicSshKey": "not a real ssh key",
  "uid": null,
  "stateOrProvince": "not a real state or province",
  "lastUpdated": "2018-12-12 16:26:32.821187+00",
  "username": "admin",
  "newUser": false,
  "addressLine2": "not a real address either",
  "role": 1,
  "addressLine1": "not a real address",
  "postalCode": "not a real postal code",
  "gid": null
}}
```



## users/register

### POST

Register a user and send registration email.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** undefined

### Request Structure

**email** Email address of the new user The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**role** The integral, unique identifier of the highest permissions *Role* which will be afforded to the new user. It restricts the allowed values to identifiers for *Roles* with at most the same permissions level as the requesting user.

**tenantId** A field containing the integral, unique identifier of the *Tenant* to which the new user will belong. It restricts the allowed values to identifiers for *Tenants* within the requesting user’s *Tenant*’s permissions.

#### 614: Request Example

```
POST /api/3.0/users/register HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 59
Content-Type: application/json

{
  "email": "test@example.com",
  "role": 3,
  "tenantId": 1
}
```

### Response Structure

#### 615: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 22:03:22 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→yvf++Oqxvu3uOIAYbWLUgJKxZ4T60Mi5H9eGTxrKLxnRsHw0PdDIrThbTnWtATBkak4vU/
→dPHLLXKW85LUTEWg==
Content-Length: 160

{ "alerts": [
  {
    "level": "success",
    "text": "Sent user registration to test@example.com with the
→following permissions [ role: read-only | tenant: root ]"
  }
]}
```

## `vault/bucket/{{bucket}}/key/{{key}}/values`

Retrieves the `object` stored under a given `key` from a given `bucket` in *Traffic Vault*.

Deprecated since version ATCv6.

### GET

**Auth. Required** Yes

**Roles Required** “admin”

**Response Type** Object

### Request Structure

Table 260: Request Path Parameters

Name	Description
bucket	The bucket that the key is stored under
key	The key that the object is stored under

#### 616: Request Example

```
GET /api/3.0/vault/bucket/ssl/key/demo1-latest/values HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

The response structure varies according to what is stored. Top-level keys will always be String type, but the values can be any type.

#### 617: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 23:29:27 GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: LUq4+MUCgIqxBWqbuA/
Pbsdq5Qfs7vPUcZ0Cu2Fc0UyP0X8xXit3BJrd0LA+dSSJ3tSQ7Qij1+0ahzuouuLT7Q==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 22:29:27 GMT
Transfer-Encoding: chunked

{
  "alerts": [
    {
      "text": "This endpoint is deprecated, and will be removed in the future",
      "level": "warning"
    }
  ],
  "response": {
    "cdn": "CDN-in-a-Box",
    "certificate": {
      "crt": "...",
      "csr": "...",
      "key": "..."
    },
    "deliveryservice": "demo1",
    "hostname": "*.demo1.mycdn.ciab.test",
    "key": "demo1",
    "version": 1
  }
}
```

## vault/ping

### GET

Pings Traffic Vault to retrieve status.

**Auth. Required** Yes

**Roles Required** “read-only”

**Response Type** Object

## Request Structure

No parameters available.

## Response Properties

**status** The status returned from the ping request to the Traffic Vault server

**server** The Traffic Vault server that was pinged

### 618: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 15:37:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳z9P1NkxGebPncUhaChDHtYKYI+XVZfhE6Y84TuwoASZFIMfISELwADLpvpPTN+wnzBfREksLYn+0313Q
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 14:37:55 GMT
Content-Length: 90

{ "response":
  {
    "status": "OK",
    "server": "trafficvault.infra.ciab.test:8087"
  }
}
```

## 5.1.6 API V4 Routes

API routes available in version 4.

**about**

**GET**

Returns info about the Traffic Ops build that is currently running, generated at startup. The output will be the same until the Traffic Ops *version changes*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** None

**Response Type** Object

### Request Structure

No parameters available.

619: Request Example

```
GET /api/4.0/about HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

**commitHash** The [Git](#) commit hash that Traffic Ops was built at.

**commits** The number of commits in the branch of the commit that Traffic Ops was built at, including that commit. Calculated by running `git rev-list HEAD | wc -l`.

**goVersion** The version of [Go](#) that was used to build Traffic Ops.

**release** The major version of CentOS or Red Hat Enterprise Linux that the build environment was running.

**name** The human-readable name of the [RPM](#) file.

**RPMVersion** The entire name of the RPM file, excluding the file extension.

**Version** The version of ATC that this version of Traffic Control belongs to.

## 620: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 19:35:28.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳7SVQsddCUVRs+sineziRGR60yMli7XLZbjxyMQgW6E506bh5thM0uttPFT7aJckDcgT45PlhexycwlApO
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 18:35:28 GMT
Content-Length: 145

{
  "commitHash": "1c9a2e9c",
  "commits": "10555",
  "goVersion": "go1.11.13",
  "release": "el7",
  "name": "traffic_ops",
  "RPMVersion": "traffic_ops-4.0.0-10555.1c9a2e9c.el7",
  "Version": "4.0.0"
}

```

## acme\_accounts

New in version 3.1.

### GET

Gets information for all *ACME Account* s.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ACME:READ

**Response Type** Array

## Request Structure

No parameters available

## Response Structure

**email** The email connected to the *ACME Account*.

**privateKey** The private key connected to the *ACME Account*.

**uri** The URI for the *ACME Account*. Differs per provider.

**provider** The ACME provider.

### 621: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": [
  {
    "email": "sample@example.com",
    "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----END RSA PRIVATE KEY-----\n",
    "uri": "https://acme.example.com/acct/1",
    "provider": "Lets Encrypt"
  }
]}
```

## POST

Creates a new *ACME Account*.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ACME:CREATE, ACME:READ

**Response Type** Object



## Request Structure

The request body must be a single *ACME Account* object with the following keys:

- email** The email connected to the *ACME Account*.
- privateKey** The private key connected to the *ACME Account*.
- uri** The URI for the *ACME Account*. Differs per provider.
- provider** The ACME provider.

### 622: Request Example

```
POST /api/4.0/acme_accounts HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 181
Content-Type: application/json

{
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
→END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}
```

## Response Structure

- email** The email connected to the *ACME Account*.
- privateKey** The private key connected to the *ACME Account*.
- uri** The URI for the *ACME Account*. Differs per provider.
- provider** The ACME provider.

### 623: Response Example

```
HTTP/1.1 201 Created
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Dec 2020 17:40:54.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512:↵
↵eQr148zWids0kDpfCYmntYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
↵Cfh2X9i3G+MQ36eH95gukJq0BOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:18:21 GMT
Content-Length: 253

{ "alerts": [
  {
    "text": "Acme account created",
    "level": "success"
  }
],
"response": {
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
↵END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}}
```

## PUT

Updates an existing *ACME Account*.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ACME:UPDATE, ACME:READ

**Response Type** Object

## Request Structure

The request body must be a single *ACME Account* object with the following keys:

**email** The email connected to the *ACME Account*.

**privateKey** The private key connected to the *ACME Account*.

**uri** The URI for the *ACME Account*. Differs per provider.

**provider** The ACME provider.

#### 624: Request Example

```
PUT /api/4.0/acme_accounts HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 181
Content-Type: application/json

{
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
→END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}
```

### Response Structure

- email** The email connected to the *ACME Account*.
- privateKey** The private key connected to the *ACME Account*.
- uri** The URI for the *ACME Account*. Differs per provider.
- provider** The ACME provider.

#### 625: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Dec 2020 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
→Cfh2X9i3G+MQ36eH95gukJqOBOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:18:21 GMT
Content-Length: 253

{ "alerts": [
```

(continues on next page)

(continued from previous page)

```
{
  "text": "Acme account updated",
  "level": "success"
},
"response": {
  "email": "sample@example.com",
  "privateKey": "-----BEGIN RSA PRIVATE KEY-----\nSampleKey\n-----
→END RSA PRIVATE KEY-----\n",
  "uri": "https://acme.example.com/acct/1",
  "provider": "Lets Encrypt"
}}
```

`acme_accounts/{provider}/{email}`

New in version 3.1.

## DELETE

Delete *ACME Account* information.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ACME:DELETE, ACME:READ

**Response Type** Object

## Request Structure

Table 261: Request Path Parameters

Name	Description
provider	The ACME provider for the account to be deleted
email	The email used in the <i>ACME Account</i> to be deleted

## Response Structure

### 626: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 10 Dec 2020 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: rGD2sOMHYF0sga1zuTytyLHCUkkc3ZwQRKvZ/
    ↳HuPzObOP4WztKTOVXB4uhs3iJqBg9zRB2TucMxONHN+3/yShQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 10 Dec 2020 14:24:34 GMT
Content-Length: 60

{"alerts": [
  {
    "text": "Acme account deleted",
    "level": "success"
  }
]}
```

### acme\_accounts/providers

New in version 4.0.

#### GET

Gets a list of all ACME providers set up in *cdn.conf* and Let's Encrypt.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ACME:READ

**Response Type** Array

### Request Structure

No parameters available

### Response Structure

627: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": [
  "CertAuth1",
  "CertAuth2",
  "CertAuth3"
]}
```

`acme_autorenew`

### POST

Generates SSL certificates and private keys for all *Delivery Services* that have certificates expiring within the configured time. This uses:abbr:ACME (*Automatic Certificate Management Environment*) or Let's Encrypt depending on the certificate.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ACME:READ, DS-SECURITY-KEY:UPDATE,  
DELIVERY-SERVICE:UPDATE

**Response Type** undefined

### Request Structure

No parameters available

## Response Structure

### 628: Response Example

```
HTTP/1.1 202 Accepted
Content-Type: application/json

{ "alerts": [
  {
    "text": "Beginning async call to renew certificates. This may
    ↳take a few minutes. Status updates can be found here: /api/4.0/async_
    ↳status/1",
    "level": "success"
  }
]}
```

## asns

### See also:

The [Autonomous System Wikipedia page](#) for an explanation of what an ASN actually is.

## GET

List all ASNs.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** ASN:READ, CACHE-GROUP:READ

**Response Type** Array

## Request Structure

Table 262: Request Query Parameters

Parameter	Required	Description
cache-group-id	no	The <i>ID</i> of a <i>Cache Group</i> - only ASNs for this <i>Cache Group</i> will be returned.
id	no	The integral, unique identifier of the desired ASN-to- <i>Cache Group</i> association
order-by	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the <i>n</i> <sup>th</sup> page of results, where “ <i>n</i> ” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

629: Request Example

```
GET /api/4.0/asns HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- asn** An ASN as specified by IANA for identifying a service provider
- cache-group** A string that is the *Name* of the *Cache Group* that is associated with this ASN
- cache-groupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN
- id** An integral, unique identifier for this association between an ASN and a *Cache Group*
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops’s Custom Date/Time Format*



## 630: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 22:51:14
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳F2NmDbTpXqrIQDX7IBKH9+1drtTL4XedSfJv6klMgLEZwbLCkddIXuSLpmgVCID6kTVqy3fTKjZS3U+HJ
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 21:51:14 GMT
Content-Length: 128

{ "response": [
  {
    "asn": 1,
    "cachegroup": "TRAFFIC_ANALYTICS",
    "cachegroupId": 1,
    "id": 1,
    "lastUpdated": "2019-12-02 21:49:08+00"
  }
]}
```

## POST

Creates a new ASN.

---

**Note:** There cannot be two different ASN object with the same `asn`. An ASN may only belong to one cachegroup, but a cachegroup can have zero or more ASNs.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ASN:CREATE, ASN:READ, CACHE-GROUP:READ,  
CACHE-GROUP:UPDATE

**Response Type** Object

## Request Structure

**asn** The value of the new ASN

**cachegroup** An optional field which, if present, is a string that specifies the *Name* of a *Cache Group* to which this ASN will be assigned

---

**Note:** While this endpoint accepts the `cachegroup` field, sending this in the request payload has no effect except that the response will (erroneously) name the *Cache Group* to which the ASN was assigned. Any subsequent requests will reveal that, in fact, the *Cache Group* is set entirely by the `cachegroupId` field, and so the actual *Name* may differ from what was in the request.

---

**cachegroupId** An integer that is the *ID* of a *Cache Group* to which this ASN will be assigned

### 631: Request Example

```
POST /api/4.0/asns HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 29

{"asn": 1, "cachegroupId": 1}
```

## Response Structure

**asn** An ASN as specified by IANA for identifying a service provider

**cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN

**cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN

**id** An integral, unique identifier for this association between an ASN and a *Cache Group*

**lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

### 632: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 22:49:08
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳mx8b2GTyOjz4QtMxXCmoQyZogCB504vs0yv6WGly4dwM81W3XiejWNuUwchRBYyi8QHaWsMZ3DaiGGfQi,
    ↳8Giw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 21:49:08 GMT
Content-Length: 150

{ "alerts": [
  {
    "text": "asn was created.",
    "level": "success"
  }
],
"response": {
  "asn": 1,
  "cachegroup": null,
  "cachegroupId": 1,
  "id": 1,
  "lastUpdated": "2019-12-02 21:49:08+00"
}}
```

## PUT

Updates an existing ASN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ASN:UPDATE, ASN:READ, CACHE-GROUP:READ,  
CACHE-GROUP:UPDATE

**Response Type** Object

## Request Structure

**asn** The value of the new ASN.

**cachegroup** A string that specifies the *Name* of a *Cache Group* to which this ASN will be assigned. If you do not pass this field, the cachegroup will be null.

**cachegroupId** The integral, unique identifier of the status of the *Cache Group*.

### 633: Request Example

```
PUT /api/4.0/asns?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 53

{
  "asn": 1,
  "cachegroup": "TRAFFIC_OPS",
  "cachegroupId": 2
}
```

## Response Structure

**asn** An ASN as specified by IANA for identifying a service provider

**cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN

**cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN

**id** An integral, unique identifier for this association between an ASN and a *Cache Group*

**lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

### 634: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 07:21:10 GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: MjvwJg6AFbdqGPlAhK+2pfiN+VFjzgeNnhXoMVbh6+fRQYKeej6CCj3x09hw0l4uhp9d9RySrE/CQ3+L1b2VGQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 06:21:10 GMT
Content-Length: 164

{
  "alerts": [
    {
      "text": "asn was updated.",
      "level": "success"
    }
  ],
  "response": {
    "asn": 1,
    "cachegroup": "TRAFFIC_OPS",
    "cachegroupId": 2,
    "id": 1,
    "lastUpdated": "2020-02-25 06:21:10+00"
  }
}
```

## DELETE

Deletes an existing ASN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ASN:DELETE, ASN:READ, CACHE-GROUP:READ, CACHE-GROUP:UPDATE

**Response Type** undefined

## Request Structure

### 635: Request Example

```
DELETE /api/4.0/asns?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

### 636: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 08:27:33.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→Woz8NSHIYVpX4V5X4xZWZIX1hvGL2uian7nUhjZ8F23Nb9RWQRMIg/cc+1vXEzkT/
→ehKV9t11FKRLX+avSae0g==
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 07:27:33 GMT
Content-Length: 83

{
  "alerts": [
    {
      "text": "asn was deleted.",
      "level": "success"
    }
  ]
}
```

`asns/{{id}}`

**See also:**

The [Autonomous System Wikipedia](#) page for an explanation of what an ASN actually is.

## PUT

Allows user to edit an existing ASN-to-*Cache Group* association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ASN:UPDATE, ASN:READ, CACHE-GROUP:UPDATE, CACHE-GROUP:READ

**Response Type** Object

## Request Structure

**asn** The new ASN which will be associated with the identified *Cache Group* - must not conflict with existing associations

**cachegroup** An optional field which, if present, is a string that specifies the *Name* of a *Cache Group* to which this ASN will be assigned

---

**Note:** While this endpoint accepts the `cachegroup` field, sending this in the request payload has no effect except that the response will (erroneously) name the *Cache Group* to which the ASN was assigned. Any subsequent requests will reveal that, in fact, the *Cache Group* is set entirely by the `cachegroupId` field, and so the actual *Name* may differ from what was in the request.

---

**cachegroupId** An integer that is the *ID* of a *Cache Group* to which this ASN will be assigned - must not conflict with existing associations

Table 263: Request Path Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the desired ASN-to- <i>Cache Group</i> association

## 637: Request Example

```
PUT /api/4.0/asns/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
Content-Length: 29
Content-Type: application/x-www-form-urlencoded

{"asn": 2, "cachegroupId": 1}
```

## Response Structure

- asn** An ASN as specified by IANA for identifying a service provider
- cachegroup** A string that is the *Name* of the *Cache Group* that is associated with this ASN
- cachegroupId** An integer that is the *ID* of the *Cache Group* that is associated with this ASN
- id** An integral, unique identifier for this association between an ASN and a *Cache Group*
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

### 638: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: /83P4LJVsrQx9BKHFxo5pbhQm1B4o3a9v3Ppkspy0JcpNx1S/
    ↳GJhCPpiANBki547sbY+0vTq76IriHZ4GYp8bA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 08 Nov 2018 14:37:39 GMT
Content-Length: 160

{ "alerts": [
  {
    "text": "asn was updated.",
    "level": "success"
  }
],
"response": {
```

(continues on next page)



(continued from previous page)

```
"asn": 2,
"cachegroup": null,
"cachegroupId": 1,
"id": 1,
"lastUpdated": "2018-11-08 14:37:39+00"
}}
```

## DELETE

Deletes an association between an ASN and a *Cache Group*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ASN:DELETE, ASN:READ, CACHE-GROUP:READ, CACHE-GROUP:UPDATE

**Response Type** undefined

## Request Structure

Table 264: Request Path Parameters

Name	Re- quired	Description
id	yes	The integral, unique identifier of the desired ASN-to- <i>Cache Group</i> association

639: Request Example

```
DELETE /api/4.0/asns/1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

640: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 23:06:24
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳6t3WA+DOcfPJB5UnvDpzEVx5ySfmJgEV9wgk071U5k32L1VXpxcaTdDVLNGgDDl9sdNftmYnKXf5jpfWU
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 22:06:24 GMT
Content-Length: 81

{ "alerts": [
  {
    "text": "asn was deleted.",
    "level": "success"
  }
]}
```

`async_status/{{id}}`

GET

Returns a status update for an asynchronous task.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ASYNC-STATUS:READ

**Response Type** Object

## Request Structure

Table 265: Request Path Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier for the desired asynchronous job status. This will be provided when the asynchronous job is started.

## Response Structure

**id** The integral, unique identifier for the asynchronous job status.

**status** The status of the asynchronous job. This will be *PENDING*, *SUCCEEDED*, or *FAILED*.

**start\_time** The time the asynchronous job was started.

**end\_time** The time the asynchronous job completed. This will be *null* if it has not completed yet.

**message** A message about the job status.

### 641: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response":
  {
    "id":1,
    "status":"PENDING",
    "start_time":"2021-02-18T17:13:56.352261Z",
    "end_time":null,
    "message":"Async job has started."
```

(continues on next page)

(continued from previous page)

```
}  
}
```

### cache\_stats

Retrieves detailed, aggregated statistics for caches in a specific CDN.

#### See also:

This gives an aggregate of statistics for *all caches* within a particular CDN and time range. For statistics basic statistics from all caches regardless of CDN and at the current time, use [caches/stats](#).

### GET

Retrieves statistics about the caches within the CDN

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Object

## Request Structure

Table 266: Request Query Parameters

Name	Required	Description
cdnName	yes	The name of a CDN. Results will represent caches within this CDN
end-Date	yes	The date and time until which statistics shall be aggregated in <a href="#">RFC 3339</a> format (with or without sub-second precision), the number of nanoseconds since the Unix Epoch, or in the same, proprietary format as the <code>lastUpdated</code> fields prevalent throughout the Traffic Ops API
exclude	no	Either “series” to omit the data series from the result, or “summary” to omit the summary data from the result - directly corresponds to fields in the <a href="#">Response Structure</a>
interval	no	Specifies the interval within which data will be “bucketed”; e.g. when requesting data from 2019-07-25T00:00:00Z to 2019-07-25T23:59:59Z with an interval of “1m”, the resulting data series (assuming it is not excluded) should contain $24 \frac{\text{hours}}{\text{day}} \times 60 \frac{\text{minutes}}{\text{hour}} \times 1 \text{day} \times 1 \frac{\text{minute}}{\text{datapoint}} = 1440 \text{datapoints}$ The allowed values for this parameter are valid InfluxQL duration literal strings matching <code>^d+[mhdw]\$</code>
limit	no	A natural number indicating the maximum amount of data points should be returned in the <code>series</code> object
metric-Type	yes	The metric type being reported - one of: ‘connections’, ‘bandwidth’, ‘maxkbps’
offset	no	A natural number of data points to drop from the beginning of the returned data set
orderby	no	Though one struggles to imagine why, this can be used to specify “time” to sort data points by their “time” (which is the default behavior)
start-Date	yes	The date and time from which statistics shall be aggregated in <a href="#">RFC 3339</a> format (with or without sub-second precision), the number of nanoseconds since the Unix Epoch, or in the same, proprietary format as the <code>lastUpdated</code> fields prevalent throughout the Traffic Ops API

### 642: Request Example

```
GET /api/4.0/cache_stats?cdnName=CDN&endDate=2019-10-28T20:49:00Z&
→metricType=bandwidth&startDate=2019-10-28T20:45:00Z HTTP/1.1
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: application/json;timestamp=unix, application/json;
→timestamp=rfc;q=0.9, application/json;q=0.8, */*;q=0.7
Connection: keep-alive
Cookie: mojolicious=...
```

## Content Format

It's important to note in *Request Example* the use of a complex “Accept” header. This endpoint accepts two special media types in the “Accept” header that instruct it on how to format the timestamps associated with the returned data. Specifically, Traffic Ops will recognize the special, optional, non-standard parameter of *application/json: timestamp*. The values of this parameter are restricted to one of

**rfc** Returned timestamps will be formatted according to **RFC 3339** (no sub-second precision).

**unix** Returned timestamps will be formatted as the number of nanoseconds since the Unix Epoch (midnight on January 1<sup>st</sup> 1970 UTC).

---

### Implementation Detail

The endpoint passes back nanoseconds, specifically, because that is the form used both by InfluxDB, which is used to store the data being served, and Go's standard library. Clients may need to convert the value to match their own standard libraries - e.g. the `Date()` class in Javascript expects milliseconds.

---

The default behavior - when only e.g. *application/json* or *\*/\** is given - is to use **RFC 3339** formatting. It will, however, respect quality parameters. It is suggested that clients request timestamps they can handle specifically, rather than relying on this default behavior, as it is **subject to change** and is in fact **expected to invert in the next major release** as string-based time formats become deprecated.

### See also:

For more information on the “Accept” HTTP header, consult [its dedicated page on MDN](#).

## Response Structure

**series** An object containing the actual data series and information necessary for working with it.

**columns** This is an array of names of the columns of the data contained in the “values” array - should always be `["time", "sum_count"]`

**count** The number of data points contained in the “values” array

**name** The name of the data set. Should always match `metric.ds.1min` where `metric` is the requested `metricType`

**values** The actual array of data points. Each represents a length of time specified by the `interval` query parameter

**time** The time at which the measurement was taken. This corresponds to the *beginning* of the interval. This time comes in the format of either an [RFC 3339](#)-formatted string, or a number containing the number of nanoseconds since the Unix Epoch depending on the “Accept” header sent by the client, according to the rules outlined in [Content Format](#).

**value** The value of the requested `metricType` at the time given by `time`. This will always be a floating point number, unless no data is available for the data interval, in which case it will be `null`

**summary** A summary of the data contained in the “series” object

**average** The arithmetic mean of the data’s values

**count** The number of measurements taken within the requested timespan. This is, in general, **not** the same as the `count` field of the `series` object, as it reflects the number of underlying, un-“bucketed” data points, and is therefore dependent on the implementation of Traffic Stats.

**fifthPercentile** Data points with values less than or equal to this number constitute the “bottom” 5% of the data set

**max** The maximum value that can be found in the requested data set

**min** The minimum value that can be found in the requested data set

**ninetyEighthPercentile** Data points with values greater than or equal to this number constitute the “top” 2% of the data set

**ninetyFifthPercentile** Data points with values greater than or equal to this number constitute the “top” 5% of the data set

#### 643: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: p4asf1n7fXGtgPW/
    ↳dWgolJWdXjwDcCjyvJOPFqkckbgoXGUHEj5/
    ↳wlz7brlQ48t3ZnOWCq0lbsu2eSiBssBtUQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 28 Oct 2019 20:49:51 GMT

{ "response": {
  "series": {
    "columns": [
      "time",
      "sum_count"
    ],
    "count": 4,
    "name": "bandwidth.cdn.1min",
    "tags": {
      "cdn": "CDN-in-a-Box"
    },
    "values": [
      [
        1572295500000000000,
        null
      ],
      [
        1572295560000000000,
        113.66666666666666
      ],
      [
        1572295620000000000,
        108.83333333333334
      ]
    ]
  }
}
```

(continues on next page)



(continued from previous page)

```

        ],
        [
            15722956800000000000,
            113
        ]
    ],
    },
    "summary": {
        "average": 111.83333333333333,
        "count": 3,
        "fifthPercentile": 0,
        "max": 113.66666666666666,
        "min": 108.83333333333334,
        "ninetyEighthPercentile": 113.66666666666666,
        "ninetyFifthPercentile": 113.66666666666666
    }
}
}

```

## cachegroups

### GET

Extract information about *Cache Groups*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CACHE-GROUP:READ, TYPE:READ

**Response Type** Array

## Request Structure

Table 267: Request Query Parameters

Name	Required	Description
id	no	Return the only <i>Cache Group</i> that has this id
name	no	Return only the <i>Cache Group</i> identified by this <i>Name</i>
type	no	Return only <i>Cache Groups</i> that are of the <i>Type</i> identified by this integral, unique identifier
topology	no	Return only <i>Cache Groups</i> that are used in the <i>Topology</i> identified by this unique identifier
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

644: Request Example

```
GET /api/4.0/cachegroups?type=23 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**fallbacks** An array of strings that are *Cache Group names* that are registered as *Fallbacks* for this *Cache Group*<sup>1</sup>

**fallbackToClosest** A boolean value that defines the *Fallback to Closest* behavior of this *Cache Group*<sup>Page 964, 1</sup>

**id** An integer that is the *ID* of the *Cache Group*

**lastUpdated** The time and date at which this entry was last updated in *Traffic Ops's Custom Date/Time Format*

<sup>1</sup> Traffic Router will first check for a fallbacks array and, when that is empty/unset/all the *Cache Groups* in it are also unavailable, will subsequently check for fallbackToClosest. If that is true, then it falls back to the geographically closest *Cache Group* capable of serving the same content or, when it is false/no such *Cache Group* exists/said *Cache Group* is also unavailable, will respond to clients with a failure response indicating the problem.

**latitude** A floating-point *Latitude* for the *Cache Group*

**localizationMethods** An array of *Localization Methods* as strings

**longitude** A floating-point *Longitude* for the *Cache Group*

**name** A string containing the *Name* of the *Cache Group*

**parentCachegroupId** An integer that is the *ID* of this *Cache Group*'s *Parent* - or null if it doesn't have a *Parent*

**parentCachegroupName** A string containing the *Name* of this *Cache Group*'s *Parent* - or null if it doesn't have a *Parent*

**secondaryParentCachegroupId** An integer that is the *ID* of this *Cache Group*'s *Secondary Parent* - or null if it doesn't have a *Secondary Parent*

**secondaryParentCachegroupName** A string containing the *Name* of this *Cache Group*'s *Secondary Parent Cache Group* - or null if it doesn't have a *Secondary Parent*

**shortName** A string containing the *Short Name* of the *Cache Group*

**typeId** An integral, unique identifier for the '*Type*' of the *Cache Group*

**typeName** A string that names the *Type* of this *Cache Group*

**Note:** The default value of `fallbackToClosest` is 'true', and if it is 'null' Traffic Control components will still interpret it as 'true'.

#### 645: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳oV6ifEgoFy+v049tVjSsRdWQf4bxjrUvIYfDdgpUtlxiC7gzCv31m5bXQ8EUBW4eg2hfYM+BsGvJpnNDZ
X-Server-Name: traffic_ops_golang/
Date: Wed, 07 Nov 2018 19:46:36 GMT
Content-Length: 379

{ "response": [
  {
    "id": 7,
    "name": "CDN_in_a_Box_Edge",
```

(continues on next page)

(continued from previous page)

```

    "shortName": "ciabEdge",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "parentCachegroupName": "CDN_in_a_Box_Mid",
    "parentCachegroupId": 6,
    "secondaryParentCachegroupName": null,
    "secondaryParentCachegroupId": null,
    "fallbackToClosest": [],
    "localizationMethods": [],
    "typeName": "EDGE_LOC",
    "typeId": 23,
    "lastUpdated": "2018-11-07 14:45:43+00",
    "fallbacks": []
  }
}]

```

## POST

Creates a *Cache Group*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CACHE-GROUP:CREATE, CACHE-GROUP:READ,  
TYPE:READ

**Response Type** Object

## Request Structure

**fallbacks** An optional field which, when present, should contain an array of strings that are the *Names* of other *Cache Groups* which will be the *Fallbacks*<sup>Page 964, 1</sup>

**fallbackToClosest** A boolean that sets the *Fallback to Closest* behavior of the *Cache Group*<sup>Page 964, 1</sup>

---

**Note:** The default value of `fallbackToClosest` is `true`, and if it is `null` Traffic Control components will still interpret it as though it were `true`.

---

**latitude** An optional field which, if present, should be a floating-point number that will define the *Latitude* for the *Cache Group*<sup>2</sup>

---

<sup>2</sup> While these fields are technically optional, note that if they are not specified many things may break. For this reason, Traffic Portal requires them when creating or editing *Cache Groups*.

**localizationMethods** Array of *Localization Methods* (as strings)

---

**Tip:** This field has no defined meaning if the *Type* identified by typeId is not “EDGE\_LOC”.

---

**longitude** An optional field which, if present, should be a floating-point number that will define the *Longitude* for the *Cache Group*?

**name** The *Name* of the *Cache Group*

**parentCachegroupId** An optional field which, if present, should be an integer that is the *ID* of a *Parent* for this *Cache Group*.

**secondaryParentCachegroupId** An optional field which, if present, should be an integral, unique identifier for this *Cache Group*’s secondary parent

**shortName** An abbreviation of the name

**typeId** An integral, unique identifier for the *Cache Group*’s *Type*

---

**Note:** The actual, integral, unique identifiers for these *Types* must first be obtained, generally via *types*.

---

#### 646: Request Example

```
POST /api/4.0/cachegroups HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 252
Content-Type: application/json

{
  "name": "test",
  "shortName": "test",
  "latitude": 0,
  "longitude": 0,
  "fallbackToClosest": true,
  "localizationMethods": [
    "DEEP_CZ",
    "CZ",
    "GEO"
  ],
  "typeId": 23,
}
```

## Response Structure

- fallbacks** An array of strings that are *Cache Group names* that are registered as *Fallbacks* for this *Cache Group*<sup>Page 964, 1</sup>
- fallbackToClosest** A boolean value that defines the *Fallback to Closest* behavior of this *Cache Group*<sup>Page 964, 1</sup>
- id** An integer that is the *ID* of the *Cache Group*
- lastUpdated** The time and date at which this entry was last updated in *Traffic Ops's Custom Date/Time Format*
- latitude** A floating-point *Latitude* for the *Cache Group*
- localizationMethods** An array of *Localization Methods* as strings
- longitude** A floating-point *Longitude* for the *Cache Group*
- name** A string containing the *Name* of the *Cache Group*
- parentCachegroupId** An integer that is the *ID* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*
- parentCachegroupName** A string containing the *Name* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*
- secondaryParentCachegroupId** An integer that is the *ID* of this *Cache Group's Secondary Parent* - or null if it doesn't have a *Secondary Parent*
- secondaryParentCachegroupName** A string containing the *Name* of this *Cache Group's Secondary Parent Cache Group* - or null if it doesn't have a *Secondary Parent*
- shortName** A string containing the *Short Name* of the *Cache Group*
- typeId** An integral, unique identifier for the '*Type*' of the *Cache Group*
- typeName** A string that names the *Type* of this *Cache Group*

### 647: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→YvZlh3rpf13nBq6SbNVhbkt3IvckbB9amqGW2JhLxWK9K3cxjBq5J2sIHBUhrLKUhE9afpxtvaYrLRxjt
→YMQ==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Wed, 07 Nov 2018 22:11:50 GMT
Content-Length: 379

{ "alerts": [
  {
    "text": "cachegroup was created.",
    "level": "success"
  }
],
"response": {
  "id": 8,
  "name": "test",
  "shortName": "test",
  "latitude": 0,
  "longitude": 0,
  "parentCachegroupName": null,
  "parentCachegroupId": null,
  "secondaryParentCachegroupName": null,
  "secondaryParentCachegroupId": null,
  "fallbackToClosest": true,
  "localizationMethods": [
    "DEEP_CZ",
    "CZ",
    "GEO"
  ],
  "typeName": "EDGE_LOC",
  "typeId": 23,
  "lastUpdated": "2019-12-02 22:21:08+00",
  "fallbacks": []
}}
```

`cachegroups/{ID}`

**PUT**

Update *Cache Group*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CACHE-GROUP:UPDATE, CACHE-GROUP:READ,  
TYPE:READ

**Response Type** Object

## Request Structure

Table 268: Request Path Parameters

Parameter	Description
ID	The <i>ID</i> of a <i>Cache Group</i>

**fallbacks** An optional field which, when present, should contain an array of strings that are the *Names* of other *Cache Groups* which will be the *Fallbacks*<sup>1</sup>

**fallbackToClosest** A boolean that sets the *Fallback to Closest* behavior of the *Cache Group*<sup>Page 964, 1</sup>

---

**Note:** The default value of `fallbackToClosest` is `true`, and if it is `null` Traffic Control components will still interpret it as though it were `true`.

---

**latitude** An optional field which, if present, should be a floating-point number that will define the *Latitude* for the *Cache Group*<sup>2</sup>

**localizationMethods** Array of *Localization Methods* (as strings)

---

**Tip:** This field has no defined meaning if the *Type* identified by `typeId` is not “EDGE\_LOC”.

---

**longitude** An optional field which, if present, should be a floating-point number that will define the *Longitude* for the *Cache Group*<sup>?</sup>

**name** The *Name* of the *Cache Group*

**parentCachegroupId** An optional field which, if present, should be an integer that is the *ID* of a *Parent* for this *Cache Group*.

**secondaryParentCachegroupId** An optional field which, if present, should be an integral, unique identifier for this *Cache Group*’s secondary parent

**shortName** An abbreviation of the name

**typeId** An integral, unique identifier for the *Cache Group*’s *Type*

---

**Note:** The actual, integral, unique identifiers for these *Types* must first be obtained, generally via *types*.

---



---

<sup>1</sup> Traffic Router will first check for a `fallbacks` array and, when that is empty/unset/all the *Cache Groups* in it are also unavailable, will subsequently check for `fallbackToClosest`. If that is `true`, then it falls back to the geographically closest *Cache Group* capable of serving the same content or, when it is `false`/no such *Cache Group* exists/said *Cache Group* is also unavailable, will respond to clients with a failure response indicating the problem.

<sup>2</sup> While these fields are technically optional, note that if they are not specified many things may break. For this reason, Traffic Portal requires them when creating or editing *Cache Groups*.



#### 648: Request Example

```
PUT /api/4.0/cachegroups/8 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 118
Content-Type: application/json

{
  "latitude": 0.0,
  "longitude": 0.0,
  "name": "test",
  "fallbacks": [],
  "fallbackToClosest": true,
  "shortName": "test",
  "typeId": 23,
  "localizationMethods": ["GEO"]
}
```

## Response Structure

**fallbacks** An array of strings that are *Cache Group names* that are registered as *Fallbacks* for this *Cache Group*<sup>Page 964, 1</sup>

**fallbackToClosest** A boolean value that defines the *Fallback to Closest* behavior of this *Cache Group*<sup>Page 964, 1</sup>

**id** An integer that is the *ID* of the *Cache Group*

**lastUpdated** The time and date at which this entry was last updated in *Traffic Ops's Custom Date/Time Format*

**latitude** A floating-point *Latitude* for the *Cache Group*

**localizationMethods** An array of *Localization Methods* as strings

**longitude** A floating-point *Longitude* for the *Cache Group*

**name** A string containing the *Name* of the *Cache Group*

**parentCachegroupId** An integer that is the *ID* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*

**parentCachegroupName** A string containing the *Name* of this *Cache Group's Parent* - or null if it doesn't have a *Parent*

**secondaryParentCachegroupId** An integer that is the *ID* of this *Cache Group's Secondary Parent* - or null if it doesn't have a *Secondary Parent*

**secondaryParentCachegroupName** A string containing the *Name* of this *Cache Group's Secondary Parent Cache Group* - or `null` if it doesn't have a *Secondary Parent*

**shortName** A string containing the *Short Name* of the *Cache Group*

**typeId** An integral, unique identifier for the '*Type*' of the *Cache Group*

**typeName** A string that names the *Type* of this *Cache Group*

#### 649: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: t1W65/2kj25QyHt0Ib0xpBaAR2sXu2k0sRZ49WjKZp/
    ↳AK5S1YWhX7VNWCUUGiN1VNM4QRNqODC/7ewhYDFUncA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 19:14:28 GMT
Content-Length: 385

{ "alerts": [
  {
    "text": "cachegroup was updated.",
    "level": "success"
  }
],
"response": {
  "id": 8,
  "name": "test",
  "shortName": "test",
  "latitude": 0,
  "longitude": 0,
  "parentCachegroupName": null,
  "parentCachegroupId": null,
  "secondaryParentCachegroupName": null,
  "secondaryParentCachegroupId": null,
  "fallbacks": [],
  "fallbackToClosest": true,
  "localizationMethods": [
    "GEO"
  ],
  "typeName": "EDGE_LOC",
```

(continues on next page)

(continued from previous page)

```
"typeId": 23,
"lastUpdated": "2018-11-14 19:14:28+00"
}}
```

## DELETE

Delete a *Cache Group*. A *Cache Group* which has assigned servers or is the *Parent* of one or more other *Cache Groups* cannot be deleted.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CACHE-GROUP:DELETE, CACHE-GROUP:READ

**Response Type** undefined

## Request Structure

Table 269: Request Path Parameters

Parameter	Description
ID	The <i>ID</i> of a <i>Cache Group</i> to be deleted

### 650: Request Example

```
DELETE /api/4.0/cacheGroups/42 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 651: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
➔GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: 5jZBg07h1eNF70J/cmlbi3Hf9KJPx+WLMb1H/pSKF3FWb/
→10GUHIN35ZOB+1N5LZYCkmk3izGbTFkiruG8I41Q==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:31:04 GMT
Content-Length: 57

{ "alerts": [
  {
    "text": "cachegroup was deleted.",
    "level": "success"
  }
]}
```

## cachegroups/{ID}/deliveryservices

### POST

Assigns all of the “assignable” servers within a *Cache Group* to one or more *Delivery Services*.

**Note:** “Assignable” here means all of the *Cache Group’s servers* that have a *Type* that matches one of the glob patterns *EDGE\** or *ORG\**. If even one server of any *Type* exists within the *Cache Group* that is not assigned to the same CDN as the *Delivery Service* to which an attempt is being made to assign them, the request will fail.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CACHE-GROUP:UPDATE, DELIVERY-  
SERVICE:UPDATE, CACHE-GROUP:READ, DELIVERY-  
SERVICE:READ

**Response Type** Object

### Request Structure

**deliveryServices** The integral, unique identifiers of the *Delivery Services* to which the *Cache Group’s servers* are being assigned

652: Request Example

```
POST /api/4.0/cachegroups/8/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
```

(continues on next page)

(continued from previous page)

```
Cookie: mojolicious=...
Content-Length: 25
Content-Type: application/json

{"deliveryServices": [2]}
```

## Response Structure

**deliveryServices** An array of integral, unique identifiers for *Delivery Services* to which the *Cache Group's servers* have been assigned

**id** An integer that is the *Cache Group's ID*

**serverNames** An array of the (short) hostnames of all of the *Cache Group's* “assignable” *Servers*

### 653: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: j/yH0gvJoaGjiLZU/
→0MA8o5He2004aJ5wh1eF9ex6F6IB01liM9Wk9RkWCw7sdiUHoy13/
→mf7gDntisZwzP7yw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 19:54:17 GMT
Content-Length: 183

{ "alerts": [
  {
    "text": "Delivery services successfully assigned to all the
→servers of cache group 8.",
    "level": "success"
  }
],
"response": {
  "id": 8,
  "serverNames": [
    "foo"
  ],

```

(continues on next page)

(continued from previous page)

```
"deliveryServices": [  
  2  
]  
}}
```

cachegroups/{ID}/queue\_update

POST

*Queue* or “dequeue” updates for all of a *Cache Group’s servers*, limited to a specific CDN.

- Auth. Required** Yes
- Roles Required** “admin” or “operations”
- Permissions Required** CACHE-GROUP:READ, CDN:READ, SERVER:READ, SERVER:QUEUE
- Response Type** Object

Request Structure

Table 270: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Cache Group</i> for which updates are being <i>Queued</i> /dequeued

- action** The action to perform; one of “queue” or “dequeue”
- cdn** The full name of the CDN in need of *Queue Updates*, or a “dequeue” thereof<sup>1</sup>
- cdnId** The integral, unique identifier for the CDN in need of *Queue Updates*, or a “dequeue” thereof<sup>Page 964, 1</sup>

654: Request Example

```
POST /api/4.0/cachegroups/8/queue_update HTTP/1.1  
Host: trafficops.infra.ciab.test  
User-Agent: curl/7.47.0  
Accept: */*  
Cookie: mojolicious=...  
Content-Length: 42  
Content-Type: application/json  
  
{  
  "action": "queue",  
  "cdn": "CDN-in-a-Box"  
}
```

<sup>1</sup> Either ‘cdn’ or ‘cdnID’ *must* be in the request data (but not both).

## Response Structure

- action** The action processed, one of “queue” or “dequeue”
- cachegroupId** An integer that is the *ID* of the *Cache Group* for which *Queue Updates* was performed or cleared
- cachegroupName** The name of the *Cache Group* for which updates were queued/dequeued
- cdn** The name of the CDN to which the queue/dequeue operation was restricted
- serverNames** An array of the (short) hostnames of the *Cache Group's servers* which are also assigned to the CDN specified in the "cdn" field

### 655: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳UAcP7LrflU1Rnfr4UqbQrJczlk5rkrcLOtTXJTFvIUXxK1Ek1ZkHkE4vewjDaVIhJJ6YQg8jmPGQpr+x1
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:19:46 GMT
Content-Length: 115

{ "response": {
  "cachegroupName": "test",
  "action": "queue",
  "serverNames": [
    "foo"
  ],
  "cdn": "CDN-in-a-Box",
  "cachegroupID": 8
}}
```

### caches/stats

An API endpoint that returns cache statistics using the *Traffic Monitor APIs*.

#### See also:

This gives a set of basic statistics for *all cache servers* at the current time. For statistics from time ranges and/or aggregated over a specific CDN, use *cache\_stats*.

### GET

Retrieves cache stats from Traffic Monitor. Also includes rows for aggregates.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CACHE-GROUP:READ, PROFILE:READ

**Response Type** Array

### Request Structure

No parameters available.

### Response Structure

**cachegroup** A string that is the *Name* of the *Cache Group* to which this *cache server* belongs

**connections** Current number of TCP connections maintained by the *cache server*

**healthy** true if Traffic Monitor has marked the *cache server* as “healthy”, false otherwise

#### See also:

*Health Protocol*

**hostname** The (short) hostname of the *cache server*

**ip** The IP address of the *cache server*

**kbps** The *cache server*’s upload speed (to clients) in Kilobits per second

**profile** The *Name* of the *Profile* in use by this *cache server*

**status** The status of the *cache server*



## 656: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 20:25:01 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: DqbLgitanS8q81/
    ↳qKC1i+ImMiEMF+SW4G9rb79FWdeWcgwFjL810tlTRp1nNNfHV+tajgjyK+wMHobqVyaNEfA==
Content-Length: 133

{ "response": [
  {
    "profile": "ALL",
    "connections": 0,
    "ip": null,
    "status": "ALL",
    "healthy": true,
    "kbps": 0,
    "hostname": "ALL",
    "cachegroup": "ALL"
  }
]}
```

## cdn\_locks

New in version 4.0.

## GET

Gets information for all CDN locks.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Array

## Request Structure

Table 271: Request Query Parameters

Parameter	Required	Description
username	no	Return only the CDN lock that the user with <code>username</code> possesses
cdn	no	Return only the CDN lock for the CDN that has the name <code>cdn</code>

## Response Structure

**userName** The username for which the lock exists.

**cdn** The name of the CDN for which the lock exists.

**message** The message or reason that the user specified while acquiring the lock.

**soft** Whether or not this is a soft(shared) lock.

**sharedUserNames** An array of the usernames that the creator of the lock has shared their lock with.

**lastUpdated** Time that this lock was last updated(created).

#### 657: Response Example

```
HTTP/2 200
Content-Type: application/json

{ "response": [
  {
    "userName": "foo",
    "cdn": "bar",
    "message": "acquiring lock to snap CDN",
    "soft": true,
    "sharedUserNames": [
      "user1"
    ],
    "lastUpdated": "2021-05-26T09:31:57-06"
  }
]}
```

## POST

Allows user to acquire a lock on a CDN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN-LOCK:CREATE, CDN:READ

**Response Type** Object

## Request Structure

The request body must be a single CDN Lock object with the following keys: `:cdn`: The name of the CDN for which the user wants to acquire a lock. `:message`: The message or reason for the user to acquire the lock. This is an optional field. `:sharedUserNames`: An array of the usernames that the creator of the lock wants to share their lock with. This is an optional field. `:soft`: Whether or not this is a soft(shared) lock. This is an optional field; `soft` will be set to `true` by default.

#### 658: Request Example

```
POST /api/4.0/cdn_locks HTTP/2
Host: localhost:8443
User-Agent: curl/7.64.2
Accept: */*
Cookie: mojolicious=...
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Content-Length: 81

{
  "cdn": "bar",
  "message": "acquiring lock to snap CDN",
  "sharedUserNames": [
    "user1"
  ],
  "soft": true
}
```

## Response Structure

**userName** The username for which the lock was created.

**cdn** The name of the CDN for which the lock was created.

**message** The message or reason that the user specified while acquiring the lock.

**soft** Whether or not this is a soft(shared) lock.

**sharedUserNames** An array of the usernames that the creator of the lock has shared their lock with.

**lastUpdated** Time that this lock was last updated(created).

### 659: Response Example

```
HTTP/2 201
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 26 May 2021 17:59:10
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→IWjt4zhg40lPDTf0ebjMTS1uHsZ8LycEaHgSS3KHnmc6Vvmw5/
→S6q70CCnbAePV2x1bxKkVEiftIxfft8vq3sg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 May 2021 16:59:10 GMT
Content-Length: 204

{ "alerts": [
  {
    "text": "soft CDN lock acquired!",
    "level": "success"
  }
]}
```

(continues on next page)

(continued from previous page)

```

    }
  ],
  "response": {
    "userName": "foo",
    "cdn": "bar",
    "message": "acquiring lock to snap CDN",
    "soft": true,
    "sharedUserNames": [
      "user1"
    ],
    "lastUpdated": "2021-05-26T10:59:10-06"
  }
}
```

## DELETE

Deletes an existing CDN Lock.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN-LOCK:DELETE, CDN:READ

**Response Type** Object

## Request Structure

Table 272: Request Query Parameters

Parameter	Required	Description
cdn	yes	Delete the CDN lock for the CDN that has the name cdn

### 660: Request Example

```

DELETE /api/4.0/cdn_locks?cdn=bar HTTP/2
Host: localhost:8443
User-Agent: curl/7.64.1
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
Content-Type: application/json
```

## Response Structure

661: Response Example

```
HTTP/2 200
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 26 May 2021 22:20:10
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: p/M20Emhaws6QLhzzoSBvpC5UnIM+/
    ↳84RI1wO42PYXiyUKWnxoQQEtm4lkN+K5NOKIH+OkyULI2ovQZP6lG0cg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 May 2021 21:20:10 GMT
Content-Length: 202

{ "alerts": [
  {
    "text": "cdn lock deleted",
    "level": "success"
  }
],
"response": {
  "userName": "foo",
  "cdn": "bar",
  "message": "acquiring lock to snap CDN",
  "soft": true,
  "sharedUserNames": [
    "user1"
  ],
  "lastUpdated": "2021-05-26T10:59:10-06"
}}
```

### cdn\_notifications

#### GET

List CDN notifications.

**Auth. Required** Yes

**Roles Required** Read-Only

**Permissions Required** CDN:READ

**Response Type** Array

## Request Structure

Table 273: Request Query Parameters

Parameter	Required	Description
cdn	no	The CDN name of the notifications you wish to retrieve.
id	no	The integral, unique identifier of the notification you wish to retrieve.
user	no	The username of the user responsible for creating the CDN notifications.

### 662: Request Example

```
GET /api/4.0/cdn_notifications HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**id** The integral, unique identifier of the notification

**cdn** The name of the CDN to which the notification belongs to

**lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*

**notification** The content of the notification

**user** The user responsible for creating the notification

### 663: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 22:51:14
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512:↵
↵F2NmDbTpXqrIQDX7IBKH9+1drtTL4XedSfJv6klMgLEZwbLCkddIXuSLpmgVCID6kTVqy3fTKjZS3U+HJ
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 21:51:14 GMT
Content-Length: 128

{ "response": [
  {
    "id": 42,
    "cdn": "cdn1",
    "lastUpdated": "2019-12-02 21:49:08+00",
    "notification": "the content of the notification",
    "user": "username123",
  }
]}
```

## POST

Creates a notification for a specific CDN.

---

**Note:** Currently only one notification per CDN is supported.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN:UPDATE

**Response Type** Object

## Request Structure

**cdn** The name of the CDN to which the notification shall belong

**notification** The content of the notification

664: Request Example

```
POST /api/4.0/cdn_notifications HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 29
```

(continues on next page)



(continued from previous page)

```
{"cdn": "cdn1", "notification": "the content of the notification"}
```

## Response Structure

- id** The integral, unique identifier of the notification
- cdn** The name of the CDN to which the notification belongs to
- lastUpdated** The time and date this server entry was last updated in *Traffic Ops's Custom Date/Time Format*
- notification** The content of the notification
- user** The user responsible for creating the notification

### 665: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 02 Dec 2019 22:49:08.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→mx8b2GTyOjz4QtMxXCmoQyZogCB504vs0yv6WGly4dwM81W3XiejWNuUwchRBYyi8QHaWsMZ3DaiGGfQi,
→8Giw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 02 Dec 2019 21:49:08 GMT
Content-Length: 150

{
  "alerts":
    [
      {
        "text": "notification was created.",
        "level": "success"
      }
    ],
  "response":
    {
      "id": 42,
      "cdn": "cdn1",
```

(continues on next page)

(continued from previous page)

```

    "lastUpdated": "2019-12-02 21:49:08+00",
    "notification": "the content of the notification",
    "user": "username123",
  }
}

```

## DELETE

Deletes an existing CDN notification.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN:UPDATE

**Response Type** undefined

## Request Structure

Table 274: Request Query Parameters

Parameter	Required	Description
id	yes	The integral, unique identifier of the notification you wish to delete.

#### 666: Request Example

```
DELETE /api/4.0/cdn_notifications?id=42 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

#### 667: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 08:27:33
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Woz8NSHIYVpX4V5X4xZWZIX1hvGL2uian7nUhjZ8F23Nb9RWQRMig/cc+1vXEzkt/
    ↳ehKV9t11FKRLX+avSae0g==
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 07:27:33 GMT
Content-Length: 83

{
  "alerts": [
    {
      "text": "notification was deleted.",
      "level": "success"
    }
  ]
}
```

## cdns

Extract information about all CDNs

### GET

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Array

### Request Structure

Table 275: Request Query Parameters

Parameter	Required	Description
domain-Name	no	Return only the CDN that has this domain name
dnssecEnabled	no	Return only the CDNs that are either dnssec enabled or not
id	no	Return only the CDN that has this id
name	no	Return only the CDN that has this name
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

## Response Structure

**dnssecEnabled** true if DNSSEC is enabled on this CDN, otherwise false

**domainName** Top Level Domain name within which this CDN operates

**id** The integral, unique identifier for the CDN

**lastUpdated** Date and time when the CDN was last modified in *Traffic Ops's Custom Date/Time Format*

**name** The name of the CDN

### 668: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳z9P1NkxGebPncUhaChDhtYKYI+XVZfhE6Y84TuwoASZFIMfISELwADLpvpPTN+wnnzBfREksLYn+0313Q
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:46:57 GMT
Content-Length: 237

{ "response": [
  {
    "dnssecEnabled": false,
    "domainName": "-",
    "id": 1,
    "lastUpdated": "2018-11-14 18:21:06+00",
    "name": "ALL"
  },
  {
    "dnssecEnabled": false,
    "domainName": "mycdn.ciab.test",
    "id": 2,
    "lastUpdated": "2018-11-14 18:21:14+00",
    "name": "CDN-in-a-Box"
  }
]}
```

### POST

Allows user to create a CDN

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN:CREATE, CDN:READ

**Response Type** Object

### Request Structure

**dnssecEnabled** If true, this CDN will use DNSSEC, if false it will not

**domainName** The top-level domain (TLD) belonging to the new CDN

**name** Name of the new CDN

669: Request Structure

```
POST /api/4.0/cdns HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 63
Content-Type: application/json

{"name": "test", "domainName": "quest", "dnssecEnabled": false}
```

### Response Structure

**dnssecEnabled** true if the CDN uses DNSSEC, false otherwise

**domainName** The top-level domain (TLD) assigned to the newly created CDN

**id** An integral, unique identifier for the newly created CDN

**name** The newly created CDN’s name

670: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→1rZRl0fQioGRrEb4nCfjGGx7y3Ub2h7BZ4z6NbhcY4acPslKSUNM8QLjWTVwLU4WpkfJNxc0yy8NlKULF
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:49:28 GMT
Content-Length: 174

{ "alerts": [
  {
    "text": "cdn was created.",
    "level": "success"
  }
],
"response": {
  "dnssecEnabled": false,
  "domainName": "quest",
  "id": 3,
  "lastUpdated": "2018-11-14 20:49:28+00",
  "name": "test"
}}
```

## cdns/capacity

### GET

Retrieves the aggregate capacity percentages of all *Cache Groups* for a given CDN.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Object

### Request Structure

No parameters available.

## Response Structure

**availablePercent** The percent of available (unused) bandwidth to 64 bits of precision<sup>1</sup>

**unavailablePercent** The percent of unavailable (used) bandwidth to 64 bits of precision<sup>1</sup>

**utilizedPercent** The percent of bandwidth currently in use to 64 bits of precision<sup>1</sup>

**maintenancePercent** The percent of bandwidth being used for administrative or analytical processes internal to the CDN to 64 bits of precision<sup>1</sup>

### 671: Response Example

```
{ "response": {
  "availablePercent": 89.0939840205533,
  "unavailablePercent": 0,
  "utilizedPercent": 10.9060020300395,
  "maintenancePercent": 0.0000139494071146245
}}
```

## cdns/dnsseckey/generate

### POST

Generates ZSK and KSK keypairs for a CDN and all associated *Delivery Services*.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DNS-SEC:CREATE, CDN:UPDATE, CDN:READ

**Response Type** Object (string)

## Request Structure

**effectiveDate** An optional string containing the date and time at which the newly-generated ZSK and KSK become effective, in **RFC 3339** format. Defaults to the current time if not specified.

**key** Name of the CDN

**kskExpirationDays** Expiration (in days) for the KSKs

**ttl** Time, in seconds, for which the keypairs shall remain valid

**zskExpirationDays** Expiration (in days) for the ZSKs

---

<sup>1</sup> Following [IEEE 754](#)



#### 672: Request Example

```
POST /api/4.0/cdns/dnsseckey/generate HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 130

{
  "key": "CDN-in-a-Box",
  "kskExpirationDays": 1095,
  "ttl": 3600,
  "zskExpirationDays": 1095
}
```

### Response Structure

#### 673: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 19:42:15
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 09SPWzeMNFgg6I/PPeXittBIh3/
  ↳zUKK1NwNlYIM9SszSrK0h/Dfz7tnwgnA7h/s6M4eYBJxykDpCfVC7xpeg==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 18:42:15 GMT
Content-Length: 89

{
  "response": "Successfully created dnssec keys for CDN-in-a-Box"
}
```

## cdns/dnsseckey/refresh

### GET

Refresh the DNSSEC keys for all CDNs. This call initiates a background process to refresh outdated keys, and immediately returns a response that the process has started.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DNS-SEC:UPDATE, CDN:UPDATE, CDN:READ

**Response Type** Object (string)

### Request Structure

No parameters available

### Response Structure

#### 674: Response Example

```
HTTP/1.1 202 Accepted
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Location: /api/4.0/async_status/3
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 20 Jul 2021 23:55:11.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: yJUGNCYygBYvHft4z0nxJ0/
→p230s3PdPT5Tld+8hIWfxmpmKDciY4D7+1Bf8S69ckmZR/yxY95kIZEbg9/jFgw==
X-Server-Name: traffic_ops_golang/
Date: Tue, 20 Jul 2021 22:55:11 GMT
Content-Length: 176

{
  "alerts": [
    {
      "text": "Starting DNSSEC key refresh in the background.
→This may take a few minutes. Status updates can be found here: /api/
→4.0/async_status/3",
```

(continues on next page)

(continued from previous page)

```

        "level": "success"
      }
    ]
  }

```

## cdns/domains

### GET

Gets a list of domains and their related Traffic Router *Profiles* for all CDNs.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ, PROFILE:READ, PARAMETER:READ

**Response Type** Array

### Request Structure

No parameters available.

### Response Structure

**domainName** The TLD assigned to this CDN

**parameterId** The *ID* for the *Parameter* that sets this TLD on the Traffic Router

**profileDescription** A short, human-readable description of the Traffic Router's profile

**profileId** The *ID* of the *Profile* assigned to the Traffic Router responsible for serving domainName

**profileName** The *Name* of the *Profile* assigned to the Traffic Router responsible for serving domainName

#### 675: Response Example

```

{ "response": [
  {
    "profileId": 12,
    "parameterId": -1,
    "profileName": "CCR_CIAB",
    "profileDescription": "Traffic Router for CDN-In-A-Box",
    "domainName": "mycdn.ciab.test"
  }
]
}

```

(continues on next page)

(continued from previous page)

```
}  
]}
```

### cdns/health

Extract health information from all *Cache Groups* across all CDNs

**See also:**

*Health Protocol*

**GET**

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CACHE-GROUP:READ

**Response Type** Object

### Request Structure

No parameters available

### Response Structure

**cachegroups** An array of objects describing the health of each Cache Group

**name** The name of the Cache Group

**offline** The number of OFFLINE caches in the Cache Group

**online** The number of ONLINE caches in the Cache Group

**totalOffline** Total number of OFFLINE caches across all Cache Groups which are assigned to any CDN

**totalOnline** Total number of ONLINE caches across all Cache Groups which are assigned to any CDN

676: Response Example

```
{ "response": {  
  "totalOffline": 0,  
  "totalOnline": 1,  
  "cachegroups": [  

```

(continues on next page)

(continued from previous page)

```
{
  "offline": 0,
  "name": "CDN_in_a_Box_Edge",
  "online": 1
}
]
```

`cdns/{ID}`

## PUT

Allows a user to edit a specific CDN

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN:UPDATE, CDN:READ

**Response Type** Object

## Request Structure

Table 276: Request Path Parameters

Name	Description
ID	Integral, unique identifier for the CDN to update

**dnssecEnabled** If true, this CDN will use DNSSEC, if false it will not

**domainName** The top-level domain (TLD) belonging to the CDN

**name** Name of the new CDN

## 677: Request Example

```
PUT /api/4.0/cdns/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 63
Content-Type: application/json

{"name": "quest", "domainName": "test", "dnssecEnabled": false}
```

## Response Structure

- dnssecEnabled** true if the CDN uses DNSSEC, false otherwise
- domainName** The top-level domain (TLD) assigned to the newly created CDN
- id** An integral, unique identifier for the newly created CDN
- name** The newly created CDN's name

### 678: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: sI1hzBwG+/
    ↳VAzoFY20kqGFA2RgrU0ThtMeeJqk0ZxH3TRxTWuA8BetACct/
    ↳XICC3n7hPDLlRVpwckEyBdyJkXg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:54:33 GMT
Content-Length: 174

{ "alerts": [
  {
    "text": "cdn was updated.",
    "level": "success"
  }
],
"response": {
  "dnssecEnabled": false,
  "domainName": "test",
  "id": 4,
  "lastUpdated": "2018-11-14 20:54:33+00",
  "name": "quest"
}}
```

## DELETE

Allows a user to delete a specific CDN

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN:DELETE, CDN:READ

**Response Type** undefined

## Request Structure

Table 277: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the CDN to delete

## Response Structure

679: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Zy4cJN6BEct4ltFLN4e296mM8XnzOs0EQ3/
→jp4TA3L+g8qtkI0WrL+ThcFq4xbJPU+KHVDSi+b0JBav3xsYPqQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:51:23 GMT
Content-Length: 58

{ "alerts": [
  {
    "text": "cdn was deleted.",
    "level": "success"
  }
]}
```

`cdns/{{ID}}/queue_update`

**POST**

*Queue* or “dequeue” updates for all servers assigned to a specific CDN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:QUEUE, CDN:READ

**Response Type** Object

## Request Structure

Table 278: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the CDN on which to (de)queue updates

Table 279: Request Query Parameters

Name	Re- quired	Description
type	no	The name of the <b>type</b> of servers, for which the updates need to be queued or dequeued.
pro- file	no	The name of the <b>profile</b> of servers, for which the updates need to be queued or dequeued.

**action** One of “queue” or “dequeue” as appropriate



#### 680: Request Example

```
POST /api/4.0/cdns/2/queue_update?type=EDGE HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 19
Content-Type: application/json

{"action": "queue"}
```

### Response Structure

**action** The action processed, either "queue" or "dequeue"

**cdnId** The integral, unique identifier for the CDN on which *Queue Updates* was performed or cleared

#### 681: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→rBpFfrrP+9IFkwsRloEM+v+I8MuBZDXqFu+WUTGtRGypnAn2gHooPoNQRYVvJGjyIQRXLvqjEtve+lH2
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:02:07 GMT
Content-Length: 41

{ "response": {
  "action": "queue",
  "cdnId": 2
}}
```

`cdns/{{name}}/configs/monitoring`

See also:

*Health Protocol*

GET

Retrieves information concerning the monitoring configuration for a specific CDN.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** MONITOR-CONFIG:READ

**Response Type** Object

## Request Structure

Table 280: Request Path Parameters

Name	Description
name	The name of the CDN for which monitoring configuration will be fetched

## Response Structure

**cacheGroups** An array of objects representing each of the *Cache Groups* being monitored within this CDN

**coordinates** An object representing the geographic location of this *Cache Group*

**latitude** This *Cache Group's latitude* as a floating-point number

**longitude** This *Cache Group's longitude* as a floating-point number

**name** A string that is this *Cache Group's name*

**config** A collection of parameters used to configure the monitoring behaviour of Traffic Monitor

**health.polling.interval** An interval in milliseconds on which to poll for cache statistics

**heartbeat.polling.interval** An interval in milliseconds on which to poll for health statistics. If missing, defaults to `health.polling.interval`.

- tm.polling.interval** The interval at which to poll for configuration updates
- deliveryServices** An array of objects representing each *Delivery Service* provided by this CDN
- hostRegexes** An array of strings which are the Delivery Service's HOST\_REGEXP-type regexes
- status** The *Delivery Service*'s status
- topology** A string that is the name of the Delivery Service's *Topology* (if assigned one)
- totalKbpsThreshold** A threshold rate of data transfer this *Delivery Service* is configured to handle, in Kilobits per second
- totalTpsThreshold** A threshold amount of transactions per second that this *Delivery Service* is configured to handle
- type** A string that is the Delivery Service's type category ("HTTP" or "DNS")
- xmlId** A string that is the *Delivery Service*'s *XMLID*
- profiles** An array of the *Profiles* in use by the *cache servers* and *Delivery Services* belonging to this CDN
- name** A string that is the *Profile's Name*
- parameters** An array of the *Parameters* in this *Profile* that relate to monitoring configuration. This can be null if the servers using this *Profile* cannot be monitored (e.g. Traffic Routers)
- health.connection.timeout** A timeout value, in milliseconds, to wait before giving up on a health check request
- health.polling.url** A URL to request for polling health. Substitutions can be made in a shell-like syntax using the properties of an object from the "trafficServers" array
- health.threshold.availableBandwidthInKbps**  
The total amount of bandwidth that servers using this profile are allowed - across all network interfaces - in Kilobits per second. This is a string and using comparison operators to specify ranges, e.g. ">10" means "more than 10 kbps"

**health.threshold.loadavg** The UNIX load-avg at which the server should be marked “unhealthy”

**See also:**

*uptime(1)*

**health.threshold.queryTime** The highest allowed length of time for completing health queries (after connection has been established) in milliseconds

**history.count** The number of past events to store; once this number is reached, the oldest event will be forgotten before a new one can be added

**type** A string that names the *Profile's Type*

**topologies** A map of *Topology* names to objects

**nodes** An array of strings which are the names of the EDGE\_LOC-type cache groups in the topology

**trafficMonitors** An array of objects representing each Traffic Monitor that monitors this CDN (this is used by Traffic Monitor's “peer polling” function)

**cachegroup** The name of the *Cache Group* to which this Traffic Monitor belongs

**fqdn** An FQDN that resolves to the IPv4 (and/or IPv6) address of the server running this Traffic Monitor instance

**hostname** The hostname of the server running this Traffic Monitor instance

**ip6** The IPv6 address of this Traffic Monitor - when applicable

**ip** The IPv4 address of this Traffic Monitor

**port** The port on which this Traffic Monitor listens for incoming connections

**profile** A string that is the *Name* of the *Profile* assigned to this Traffic Monitor

**status** The status of the server running this Traffic Monitor instance

**trafficServers** An array of objects that represent the *cache servers* being monitored within this CDN

**cachegroup** The *Cache Group* to which this *cache server* belongs

**deliveryServices** An array of objects which contain the XML IDs of the delivery services to which this cache server is assigned (this field is omitted entirely if no delivery services are assigned to this cache server)

**xmlId** A string which is the XML ID of the delivery service

**fqdn** An FQDN that resolves to the *cache server*'s IPv4 (or IPv6) address

**hashId** The (short) hostname for the *cache server* - named "hashId" for legacy reasons

**hostName** The (short) hostname of the *cache server*

**port** The port on which the *cache server* listens for incoming connections

**profile** A string that is the *Name* of the *Profile* assigned to this *cache server*

**status** The status of the *cache server*

**type** A string that names the *Type* of the *cache server* - should (ideally) be either EDGE or MID

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or null to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at

this IP address. When this property is `true`, the IP address is referred to as a “service address”. It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the “service interface” for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning “no limit”. It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network in-

interface's device name on the server, e.g.  
eth0.

## 682: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→uLR+tRoqR8SY038j3DV9wQ+IkJ7Kf+MCoFkcWZtsgbpLJ+0S6f+IiI8laNVeDgrM/
→P23MAQ6BSepm+EJRL1AXQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:09:31 GMT
Transfer-Encoding: chunked

{ "response": {
  "topologies": {
    "example-topology": {
      "nodes": [
        "CDN_in_a_Box_Edge"
      ]
    }
  },
  "trafficServers": [
    {
      "profile": "ATS_EDGE_TIER_CACHE",
      "status": "REPORTED",
      "port": 80,
      "interfaces": [
        {
          "ipAddresses": [
            {
              "address": "172.16.239.100",
              "gateway": "172.16.239.0/24",
              "serviceAddress": "true"
            },
            {
              "address": "fc01:9400:1000:8::100",
              "gateway": "fc01::",
              "serviceAddress": "true"
            }
          ]
        }
      ]
    }
  ],
}
```

(continues on next page)

(continued from previous page)

```

        "name": "eth0",
        "monitor": "true",
        "mtu": 9000,
        "maxBandwidth": 150
    },
    ],
    "cachegroup": "CDN_in_a_Box_Edge",
    "hostname": "edge",
    "fqdn": "edge.infra.ciab.test",
    "type": "EDGE",
    "hashid": "edge"
},
{
    "profile": "ATS_MID_TIER_CACHE",
    "status": "REPORTED",
    "port": 80,
    "interfaces": [
        {
            "ipAddresses": [
                {
                    "address": "172.16.239.120",
                    "gateway": "172.16.239.0/24",
                    "serviceAddress": "true"
                },
                {
                    "address": "fc02:9400:1000:8::100",
                    "gateway": "fc02::",
                    "serviceAddress": "true"
                }
            ],
            "name": "eth0",
            "monitor": "true",
            "mtu": 9000,
            "maxBandwidth": 150
        }
    ],
    "cachegroup": "CDN_in_a_Box_Mid",
    "hostname": "mid",
    "fqdn": "mid.infra.ciab.test",
    "type": "MID",
    "hashid": "mid"
}
],
"trafficMonitors": [
    {
        "profile": "RASCAL-Traffic_Monitor",

```

(continues on next page)



(continued from previous page)

```

        "status": "ONLINE",
        "ip": "172.16.239.40",
        "ip6": "fc01:9400:1000:8::40",
        "port": 80,
        "cachegroup": "CDN_in_a_Box_Edge",
        "hostname": "trafficmonitor",
        "fqdn": "trafficmonitor.infra.ciab.test"
    }
],
"cacheGroups": [
    {
        "name": "CDN_in_a_Box_Mid",
        "coordinates": {
            "latitude": 38.897663,
            "longitude": -77.036574
        }
    },
    {
        "name": "CDN_in_a_Box_Edge",
        "coordinates": {
            "latitude": 38.897663,
            "longitude": -77.036574
        }
    }
],
"profiles": [
    {
        "name": "CCR_CIAB",
        "type": "CCR",
        "parameters": null
    },
    {
        "name": "ATS_EDGE_TIER_CACHE",
        "type": "EDGE",
        "parameters": {
            "health.connection.timeout": 2000,
            "health.polling.url": "http://${hostname}/_astats?
→application=&inf.name=${interface_name}",
            "health.threshold.availableBandwidthInKbps": ">1750000
→",
            "health.threshold.loadavg": "25.0",
            "health.threshold.queryTime": 1000,
            "history.count": 30
        }
    }
],
{

```

(continues on next page)

(continued from previous page)

```

        "name": "ATS_MID_TIER_CACHE",
        "type": "MID",
        "parameters": {
            "health.connection.timeout": 2000,
            "health.polling.url": "http://${hostname}/_astats?
→application=&inf.name=${interface_name}",
            "health.threshold.availableBandwidthInKbps": ">1750000
→",
            "health.threshold.loadavg": "25.0",
            "health.threshold.queryTime": 1000,
            "history.count": 30
        }
    },
    ],
    "deliveryServices": [
        {
            "xmlId": "example-ds",
            "totalTpsThreshold": 0,
            "status": "REPORTED",
            "totalKbpsThreshold": 0,
            "type": "DNS",
            "topology": "example-topology",
            "hostRegexes": [
                ".*\\.example-ds\\..*"
            ]
        }
    ],
    "config": {
        "health.polling.interval": 6000,
        "heartbeat.polling.interval": 3000,
        "peers.polling.interval": 3000,
        "tm.polling.interval": 2000
    }
}
}

```

`cdns/{{name}}/dnsseckey/ksk/generate`

**POST**

Generates a new KSK for a specific CDN.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DNS-SEC:CREATE, CDN:UPDATE, CDN:READ

**Response Type** Object (string)

## Request Structure

Table 281: Request Path Parameters

Name	Required	Description
name	yes	The name of the CDN for which the KSK will be generated

**expirationDays** The integral number of days until the newly generated KSK expires

**effectiveDate** An optional string containing the date and time at which the newly generated KSK becomes effective, in **RFC 3339** format. Defaults to the current time if not specified

## Response Structure

683: Response Example

```
{ "response": "Successfully generated ksk dnssec keys for my-cdn-name" }
```

**cdns/{{name}}/federations**

**GET**

Retrieves a list of federations in use by a specific CDN.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ, FEDERATION:READ, DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

Table 282: Request Path Parameters

Name	Description
name	The name of the CDN for which federations will be listed

Table 283: Request Query Parameters

Name	Description
id	Return only the federation that has this id
order-by	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

684: Request Example

```
GET /api/4.0/cdns/CDN-in-a-Box/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cname** The Canonical Name (CNAME) used by the federation

**deliveryService** An object with keys that provide identifying information for the *Delivery Service* using this federation

**id** The integral, unique identifier for the *Delivery Service*

**xmlId** The *Delivery Service*’s uniquely identifying ‘xml\_id’

**description** An optionally-present field containing a description of the field

---

**Note:** This key will only be present if the description was provided when the federation was created. Refer to the POST method of this endpoint to see how federations can be created.

---

**lastUpdated** The date and time at which this federation was last modified, in *Traffic Ops’s Custom Date/Time Format*

**ttl** Time to Live (TTL) for the `cname`, in hours

## 685: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
    ↳SJA7G+7G5Kc0fCtnE3Dq5DCobWtGRUKSppiDkflZoG5+paq4E1aZGqUb6vGVsd+TpPg75MLlhyqfdfCHnl
    ↳g==
x-server-name: traffic_ops_golang/
content-length: 170
date: Wed, 05 Dec 2018 00:35:40 GMT

{ "response": [
  {
    "id": 1,
    "cname": "test.quest.",
    "ttl": 48,
    "description": "A test federation",
    "lastUpdated": "2018-12-05 00:05:16+00",
    "deliveryService": {
      "id": 1,
      "xmlId": "demo1"
    }
  }
]}
```

## POST

Creates a new federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION:CREATE, FEDERATION:READ,  
CDN:READ

**Response Type** Object

## Request Structure

Table 284: Request Path Parameters

Name	Description
name	The name of the CDN for which a new federation will be created

**cname** The Canonical Name (CNAME) used by the federation

---

**Note:** The CNAME must end with a “.”

---

**description** An optional description of the federation

**ttl** Time to Live (TTL) for the name record used for **cname**

### 686: Request Example

```
POST /api/4.0/cdns/CDN-in-a-Box/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 72
Content-Type: application/json

{
  "cname": "test.quest.",
  "ttl": 48,
  "description": "A test federation"
}
```

## Response Structure

**cname** The Canonical Name (CNAME) used by the federation

**description** An optionally-present field containing a description of the field

---

**Note:** This key will only be present if the description was provided when the federation was created

---

**lastUpdated** The date and time at which this federation was last modified, in *Traffic Ops's Custom Date/Time Format*

**ttl** Time to Live (TTL) for the **cname**, in hours

## 687: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
    ↳rRsWAIhXzVlj8Hy+8aFjp4Jo1QGTK49m0N1AP5QDyyAZ1TfNIdgtcgiuehu7FiN1IPWRFiv6D9CygFYKG
x-server-name: traffic_ops_golang/
content-length: 192
date: Wed, 05 Dec 2018 00:05:16 GMT

{ "alerts": [
  {
    "text": "cdnfederation was created.",
    "level": "success"
  }
],
"response": {
  "id": 1,
  "cname": "test.quest.",
  "ttl": 48,
  "description": "A test federation",
  "lastUpdated": "2018-12-05 00:05:16+00"
}}
```

cdns/{name}/federations/{ID}

## PUT

Updates a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION:UPDATE, FEDERATION:READ,  
CDN:READ

**Response Type** Object

## Request Structure

Table 285: Request Path Parameters

Name	Description
name	The name of the CDN for which the federation identified by ID will be inspected
ID	An integral, unique identifier for the federation to be inspected

**cname** The Canonical Name (CNAME) used by the federation

---

**Note:** The CNAME must end with a “.”

---

**description** An optional description of the federation

**ttl** Time to Live (TTL) for the name record used for **cname**

688: Request Example

```
PUT /api/4.0/cdns/CDN-in-a-Box/federations/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 33
Content-Type: application/json

{
  "cname": "foo.bar.",
  "ttl": 48
}
```

## Response Structure

**cname** The Canonical Name (CNAME) used by the federation

**description** An optionally-present field containing a description of the field

---

**Note:** This key will only be present if the description was provided when the federation was created

---

**lastUpdated** The date and time at which this federation was last modified, in *Traffic Ops's Custom Date/Time Format*

**ttl** Time to Live (TTL) for the **cname**, in hours



## 689: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
    ↳qcjfQ+gDjNxYQ1aq+dlddgrkFWnkFYxsFF+SHDqqH0uVHBVksmU0aTFgltozek/
    ↳u6wbrGoR1LFf9Fr1C1SbigA==
x-server-name: traffic_ops_golang/
content-length: 174
date: Wed, 05 Dec 2018 01:03:40 GMT

{ "alerts": [
    {
        "text": "cdnfederation was updated.",
        "level": "success"
    }
],
"response": {
    "id": 1,
    "cname": "foo.bar.",
    "ttl": 48,
    "description": null,
    "lastUpdated": "2018-12-05 01:03:40+00"
}}
```

## DELETE

Deletes a specific federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION:DELETE, FEDERATION:READ,  
CDN:READ

**Response Type** undefined

## Request Structure

Table 286: Request Path Parameters

Name	Description
name	The name of the CDN for which the federation identified by ID will be inspected
ID	An integral, unique identifier for the federation to be inspected

### 690: Request Example

```
DELETE /api/4.0/cdns/CDN-in-a-Box/federations/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 691: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512: Cnkfj6dmzTD3if9oiDq33tqf7CnAflKK/
→SPgqJyfu6HUf0jLJ0gKIZvkcs2wWY6EjLVdw5qsatsd4FPoCyjvcw==
x-server-name: traffic_ops_golang/
content-length: 68
date: Wed, 05 Dec 2018 01:17:24 GMT

{ "alerts": [
  {
    "text": "cdnfederation was deleted.",
    "level": "success"
  }
]}
```

`cdns/{{name}}/health`

**GET**

Retrieves the health of all *Cache Groups* for a given CDN.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ, CACHE-GROUP:READ

**Response Type** Object

## Request Structure

Table 287: Request Path Parameters

Name	Description
name	The name of the CDN for which health will be reported

692: Request Example

```
GET /api/4.0/cdns/CDN-in-a-Box/health HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**cachegroups** An array of objects describing the health of each *Cache Group*

**name** A string that is the *Cache Group's Name*

**offline** The number of OFFLINE *cache servers* in the *Cache Group*

**online** The number of ONLINE *cache servers* in the *Cache Group*

**totalOffline** Total number of OFFLINE *cache servers* across all *Cache Groups* which are assigned to the CDN defined by the name request path parameter

**totalOnline** Total number of ONLINE *cache servers* across all *Cache Groups* which are assigned to the CDN defined by the name request path parameter

## 693: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Encoding: gzip
Content-Length: 108
Content-Type: application/json
Date: Tue, 03 Dec 2019 21:33:59 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; expires=Wed, 04 Dec 2019 01:33:59 GMT;
    ↳path=/; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
    ↳KpXViXeAgch58ueQqdyU8NuINBw1EUedE6Rv2ewcLUajJp6kowdbVynpwW7XiSvAyHdtClIOuT30khImq

{ "response": {
  "totalOffline": 0,
  "totalOnline": 1,
  "cachegroups": [
    {
      "offline": 0,
      "name": "CDN_in_a_Box_Edge",
      "online": 1
    }
  ]
}}
```

`cdns/name/{name}}`

### DELETE

Allows a user to delete a CDN by name

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN:DELETE

**Response Type** undefined

## Request Structure

Table 288: Request Path Parameters

Name	Description
name	The name of the CDN to be deleted

## Response Structure

694: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Zy4cJN6BEct4ltFLN4e296mM8XnzOs0EQ3/
    ↳jp4TA3L+g8qtkI0WrL+ThcFq4xbJPU+KHVDSi+b0JBav3xsYPqQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 20:59:22 GMT
Content-Length: 58

{ "alerts": [
  {
    "text": "cdn was deleted.",
    "level": "success"
  }
]}
```

`cdns/name/{{name}}/dnsseckey`

### GET

Gets a list of DNSSEC keys for CDN and all associated *Delivery Services*. Before returning response to user, this will make sure DNSSEC keys for all *Delivery Services* exist and are not expired. If they don't exist or are expired, they will be (re-)generated.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DNS-SEC:READ,      CDN:READ,      DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

Table 289: Request Path Parameters

Name	Description
name	The name of the CDN for which keys will be fetched

## Response Structure

**name** The name of the CDN or *Delivery Service* to which the enclosed keys belong

**zsk** The short-term ZSK

**expirationDate** A Unix epoch timestamp (in seconds) representing the date and time whereupon the key will expire

**inceptionDate** A Unix epoch timestamp (in seconds) representing the date and time when the key was created

**name** The name of the domain for which this key will be used

**private** Encoded private key

**public** Encoded public key

**ttd** The time for which the key should be trusted by the client

**ksk** The long-term KSK

**dsRecord** An optionally present object containing information about the algorithm used to generate the key

**algorithm** The name of the algorithm used to generate the key

**digest** A hash of the DNSKEY record

**digestType** The type of hash algorithm used to create the value of digest

**expirationDate** A Unix epoch timestamp (in seconds) representing the date and time whereupon the key will expire

**inceptionDate** A Unix epoch timestamp (in seconds) representing the date and time when the key was created

**name** The name of the domain for which this key will be used

**private** Encoded private key

**public** Encoded public key

**ttl** The time for which the key should be trusted by the client

#### 695: Response Example

```
{ "response": {
  "cdn1": {
    "zsk": {
      "ttl": "60",
      "inceptionDate": "1426196750",
      "private": "zsk private key",
      "public": "zsk public key",
      "expirationDate": "1428788750",
      "name": "foo.kabletown.com."
    },
    "ksk": {
      "name": "foo.kabletown.com.",
      "expirationDate": "1457732750",
      "public": "ksk public key",
      "private": "ksk private key",
      "inceptionDate": "1426196750",
      "ttl": "60",
      "dsRecord": {
        "algorithm": "5",
        "digestType": "2",
        "digest": "abc123def456"
      }
    }
  },
  "ds-01": {
    "zsk": {
      "ttl": "60",
      "inceptionDate": "1426196750",
      "private": "zsk private key",
      "public": "zsk public key",
```

(continues on next page)

(continued from previous page)

```

        "expirationDate": "1428788750",
        "name": "ds-01.foo.kabletown.com."
    },
    "ksk": {
        "name": "ds-01.foo.kabletown.com.",
        "expirationDate": "1457732750",
        "public": "ksk public key",
        "private": "ksk private key",
        "inceptionDate": "1426196750"
    }
}
}}
```

## DELETE

Delete DNSSEC keys for a CDN and all associated *Delivery Services*.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DNS-SEC:DELETE, CDN:UPDATE, DELIVERY-SERVICE:UPDATE, CDN:READ

**Response Type** Object (string)

## Request Structure

Table 290: Request Path Parameters

Name	Description
name	The name of the CDN for which DNSSEC keys will be deleted

## Response Structure



#### 696: Response Example

```
{
  "response": "Successfully deleted dnssec keys for test"
}
```

`cdns/name/{{name}}/sslkeys`

**GET**

Returns SSL certificates for all *Delivery Services* that are a part of the CDN.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DS-SECURITY-KEY:READ, CDN:READ, DELIVERY-SERVICE:READ

**Response Type** Array

### Request Structure

Table 291: Request Path Parameters

Name	Description
name	The name of the CDN for which keys will be fetched

### Response Structure

**certificate** An object representing The SSL keys used for the *Delivery Service* identified by `deliveryservice`

**key** Base 64-encoded private key for SSL certificate

**crt** Base 64-encoded SSL certificate

**deliveryservice** A string that is the *xml\_id* of the *Delivery Service* using the SSL key within `certificate`

#### 697: Response Example

```
{ "response": [
  {
    "deliveryservice": "ds1",
    "certificate": {
      "crt": "base64encodedcrt1",
```

(continues on next page)

(continued from previous page)

```

        "key": "base64encodedkey1"
    },
    {
        "deliveryservice": "ds2",
        "certificate": {
            "crt": "base64encodedcrt2",
            "key": "base64encodedkey2"
        }
    }
]

```

`cdns/{name}/snapshot`

**Caution:** This page is a stub! Much of it may be missing or just downright wrong - it needs a lot of love from people with the domain knowledge required to update it.

## GET

Retrieves the *current Snapshot* for a CDN, which represents the current *operating state* of the CDN, **not** the current *configuration* of the CDN. The contents of this *Snapshot* are currently used by Traffic Monitor and Traffic Router.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN-SNAPSHOT:READ

**Response Type** Object

## Request Structure

Table 292: Request Path Parameters

Name	Description
name	The name of the CDN for which a <i>Snapshot</i> shall be returned

## 698: Request Example

```

GET /api/4.0/cdns/CDN-in-a-Box/snapshot HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*

```

(continues on next page)

(continued from previous page)

**Connection:** keep-alive  
**Cookie:** mojolicious=...

## Response Structure

**config** An object containing basic configurations on the actual CDN object

**api.cache-control.max-age** A string containing an integer which specifies the value of `max-age` in the `Cache-Control` header of some HTTP responses, likely the *Traffic Router API* responses

**certificates.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for updated SSL certificates

**consistent.dns.routing** A string containing a boolean which indicates whether DNS routing will use a consistent hashing method or “round-robin”

“false” The “round-robin” method will be used to define DNS routing

“true” A consistent hashing method will be used to define DNS routing

**coveragezone.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should check for a new Coverage Zone file

**coveragezone.polling.url** The URL where a *Coverage Zone File* may be requested by Traffic Routers

**dnssec.dynamic.response.expiration** A string containing a number and unit suffix that specifies the length of time for which dynamic responses to DNSSEC lookup queries should remain valid

**dnssec.dynamic.concurrencylevel** An integer that defines the size of the concurrency level (threads) of the Guava cache used by ZoneManager to store zone material

**dnssec.dynamic.initialcapacity** An integer that defines the initial size of the Guava cache, default is 10000. Too low of a value can lead to expensive resizing

**dnssec.init.timeout** An integer that defines the number of minutes to allow for zone generation, this bounds the zone priming activity

**dnssec.enabled** A string that tells whether or not the CDN uses DNSSEC; one of:

“false” DNSSEC is not used within this CDN

“true” DNSSEC is used within this CDN

**domain\_name** A string that is the TLD served by the CDN

**federationmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new federation mappings

**federationmapping.polling.url** The URL where Traffic Control components can request federation mappings

**geolocation.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new IP-to-geographic-location mapping databases

**geolocation.polling.url** The URL where Traffic Control components can request IP-to-geographic-location mapping database files

**keystore.maintenance.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should refresh their zone caches

**neustar.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new “Neustar” databases

**neustar.polling.url** The URL where Traffic Control components can request “Neustar” databases

**soa** An object defining the SOA for the CDN’s TLD (defined in `domain_name`)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

**See also:**

[The Wikipedia page on Start of Authority records.](#)

**steeringmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Control components should check for new steering mappings

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

**zonemanager.cache.maintenance.interval** A configuration option for the ZoneManager Java class of Traffic Router

**zonemanager.threadpool.scale** A configuration option for the ZoneManager Java class of Traffic Router

**contentRouters** An object containing keys which are the (short) hostnames of the Traffic Routers that serve requests for *Delivery Services* in this CDN

**api.port** A string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTP

**secure.api.port** An optionally present string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTPS

**fqdn** This Traffic Router's FQDN

**httpsPort** The port number on which this Traffic Router listens for incoming HTTPS requests

**ip** This Traffic Router's IPv4 address

**ip6** This Traffic Router's IPv6 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Router belongs

**port** The port number on which this Traffic Router listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by this Traffic Router

**status** The health status of this Traffic Router

**See also:**

*Health Protocol*

**contentServers** An object containing keys which are the (short) hostnames of the *Edge-tier cache servers* in the CDN; the values corresponding to those keys are routing information for said servers

**cacheGroup** A string that is the *Name* of the *Cache Group* to which the server belongs

**capabilities** An array of this *Cache Server's Server Capabilities*. If the Cache Server has no Server Capabilities, this field is omitted.

**deliveryServices** An object containing keys which are the names of *Delivery Services* to which this *cache server* is assigned; the values corresponding to those keys are arrays of FQDNs that resolve to this *cache server*

---

**Note:** Only *Edge-tier cache servers* can be assigned to a *Delivery Service*, and therefore this field will only be present when **type** is "EDGE".

---

**fqdn** The server's FQDN

**hashCount** The number of servers to be placed into a single "hash ring" in Traffic Router

**hashId** A unique string to be used as the key for hashing servers - as of version 3.0.0 of Traffic Control, this is always the same as the server's (short) hostname and only still exists for legacy compatibility reasons

**httpsPort** The port on which the *cache server* listens for incoming HTTPS requests

**interfaceName** The name of the main network interface device used by this *cache server*

**ip6** The server's IPv6 address

**ip** The server's IPv4 address

**locationId** This field is exactly the same as `cacheGroup` and only exists for legacy compatibility reasons

**port** The port on which this *cache server* listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by the *cache server*

**routingDisabled** An integer representing the boolean concept of whether or not Traffic Routers should route client traffic to this *cache server*; one of:

**0** Do not route traffic to this server

**1** Route traffic to this server normally

**status** This *cache server*'s status

**See also:**

*Health Protocol*

**type** The *Type* of this *cache server*; which ought to be one of (but in practice need not be in certain special circumstances):

**EDGE** This is an *Edge-tier cache server*

**MID** This is a *Mid-tier cache server*

**deliveryServices** An object containing keys which are the *xml\_ids* of all of the *Delivery Services* within the CDN

**anonymousBlockingEnabled** A string containing a boolean that tells whether or not *Anonymous Blocking* is set on this *Delivery Service*; one of:

**“true”** Anonymized IP addresses are blocked by this *Delivery Service*

**“false”** Anonymized IP addresses are not blocked by this *Delivery Service*

**See also:**

*Configure Anonymous Blocking*

**consistentHashQueryParameters** A set of query parameters that Traffic Router should consider when determining a consistent hash for a given client request.

**consistentHashRegex** An optional regular expression that will ensure clients are consistently routed to a *cache server* based on matches to it.

**coverageZoneOnly** A string containing a boolean that tells whether or not this *Delivery Service* routes traffic based only on its *Coverage Zone File*

**See also:**

*Geo Limit*

**deepCachingType** A string that defines the *Deep Caching* setting of this *Delivery Service*

**dispersion** An object describing the “dispersion” - or number of *cache servers* within a single *Cache Group* across which the same content is spread - within the *Delivery Service*

**limit** The maximum number of *cache servers* in which the response to a single request URL will be stored

---

**Note:** If this is greater than the number of *cache servers* in the *Cache Group* chosen to service the request, then content will be spread across all of them. That is, it causes no problems.

---

**shuffled** A string containing a boolean that tells whether the *cache servers* chosen for content dispersion are chosen randomly or based on a consistent hash of the request URL; one of:

“false” *cache servers* will be chosen consistently

“true” *cache servers* will be chosen at random

**domains** An array of domains served by this *Delivery Service*

**ecsEnabled** A string containing a boolean from *EDNS0 Client Subnet Enabled* that tells whether EDNS0 client subnet is enabled on this *Delivery Service*; one of:

“false” EDNS0 client subnet is not enabled on this *Delivery Service*

“true” EDNS0 client subnet is enabled on this *Delivery Service*

**geolocationProvider** The name of a provider for IP-to-geographic-location mapping services - currently the only valid value is "maxmindGeolocationService"



**ip6RoutingEnabled** A string containing a boolean that defines the *IPv6 Routing Enabled* setting for this *Delivery Service*; one of:

“false” IPv6 traffic will not be routed by this *Delivery Service*

“true” IPv6 traffic will be routed by this *Delivery Service*

**matchList** An array of methods used by Traffic Router to determine whether or not a request can be serviced by this *Delivery Service*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integral, unique identifier for the set of types to which the **type** field belongs

**type** The name of the *Type* of match performed using **pattern** to determine whether or not to use this *Delivery Service*

**HOST\_REGEX** Use the *Delivery Service* if **pattern** matches the *Host* HTTP header of an HTTP request, or the name requested for resolution in a DNS request

**HEADER\_REGEX** Use the *Delivery Service* if **pattern** matches an HTTP header (both the name and value) in an HTTP request<sup>1</sup>

**PATH\_REGEX** Use the *Delivery Service* if **pattern** matches the request path of this *Delivery Service*'s URL Page 1040, 1

**STEERING\_REGEX** Use the *Delivery Service* if **pattern** matches the *xml\_id* of one of this *Delivery Service*'s “Steering” target *Delivery Services*

**missLocation** An object representing the default geographic coordinates to use for a client when lookup of their IP has failed in both the *Coverage Zone File* (and/or possibly

---

<sup>1</sup> These only apply to HTTP-routed *Delivery Services*

the *Deep Coverage Zone File*) and the IP-to-geographic-location database

**lat** Geographic latitude as a floating point number

**long** Geographic longitude as a floating point number

**protocol** An object that describes how the *Delivery Service* ought to handle HTTP requests both with and without TLS encryption

**acceptHttps** A string containing a boolean that tells whether HTTPS requests should be normally serviced by this *Delivery Service*; one of:

**“false”** Refuse to service HTTPS requests

**“true”** Service HTTPS requests normally

**redirectToHttps** A string containing a boolean that tells whether HTTP requests ought to be re-directed to use HTTPS; one of:

**“false”** Do not redirect unencrypted traffic; service it normally

**“true”** Respond to HTTP requests with instructions to use HTTPS instead

See also:

*Protocol*

**regionalGeoBlocking** A string containing a boolean that defines the *Regional Geoblocking* setting of this *Delivery Service*; one of:

**“false”** Regional Geographic Blocking is not used by this *Delivery Service*

**“true”** Regional Geographic Blocking is used by this *Delivery Service*

See also:

*Configure Regional Geo-blocking (RGB)*

**requiredCapabilities** An array of this Delivery Service’s *required capabilities*. If there are no required capabilities,

this field is omitted.

**routingName** A string that is this *Delivery Service's Routing Name*

**soa** An object defining the SOA record for the *Delivery Service's* TLDs (defined in domains)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

**See also:**

[The Wikipedia page on Start of Authority records.](#)

**sslEnabled** A string containing a boolean that tells whether this *Delivery Service* uses SSL; one of:

**“false”** SSL is not used by this *Delivery Service*

**“true”** SSL is used by this *Delivery Service*

See also:

*Protocol*

**topology** The name of the *Topology* that this *Delivery Service* is assigned to. If the Delivery Service is not assigned to a topology, this field is omitted.

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

---

**Note:** This overrides `config.ttls`.

---

**edgeLocations** An object containing keys which are the names of Edge-Tier *Cache Groups* within the CDN

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines the *Fallback to Closest* behavior of this *Cache Group*; one of:

**“false”** Do not fall back on the closest available *Cache Group*

**“true”** Fall back on the closest available *Cache Group*

**list** If this *Cache Group* has any *Fallbacks*, this key will appear and will be an array of those *Cache Groups' Names*

**latitude** A floating point number that defines this *Cache Group's Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

**monitors** An object containing keys which are the (short) hostnames of Traffic Monitors within this CDN

**fqdn** The FQDN of this Traffic Monitor

**httpsPort** The port number on which this Traffic Monitor listens for incoming HTTPS requests

**ip6** This Traffic Monitor's IPv6 address

**ip** This Traffic Monitor's IPv4 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Monitor belongs

**port** The port number on which this Traffic Monitor listens for incoming HTTP requests

**profile** A string which is the *Name* of the *Profile* used by this Traffic Monitor

---

**Note:** For legacy reasons, this must always start with "RASCAL-".

---

**status** The health status of this Traffic Monitor

**See also:**

*Health Protocol*

**stats** An object containing metadata information regarding the CDN

**CDN\_name** The name of this CDN

**date** The UNIX epoch timestamp date in the Traffic Ops server's own timezone

**tm\_host** The FQDN of the Traffic Ops server

**tm\_user** The username of the currently logged-in user

**tm\_version** The full version number of the Traffic Ops server, including release number, git commit hash, and supported Enterprise Linux version

**topologies** An array of *Topologies* where each key is the name of that Topology.

**nodes** An array of the names of the *Edge-Tier Cache Groups* in this *Topology*. *Mid-Tier* Cache Groups in the topology are not included.

**trafficRouterLocations** An object containing keys which are the *names of Cache Groups* within the CDN which contain Traffic Routers

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines this *Cache Group's Fallback to Closest* setting; one of:

**"false"** Do not fall back on the closest available *Cache Group*

“true” Fall back on the closest available *Cache Group*

**latitude** A floating point number that defines this *Cache Group’s Latitude*

**localizationMethods** An array of strings that represents this *Cache Group’s Localization Methods*

**longitude** A floating point number that defines this *Cache Group’s Longitude*

#### 699: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 27 May 2020 18:33:17
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→B5qdN9URIfu1lgQxPZ8YaaMvy2HMrzsnrpt6vF037yv60QiKCRyrUMX6wYs7QW4YVaeUrvmS2ya512YC0
X-Server-Name: traffic_ops_golang/
Date: Wed, 27 May 2020 17:33:17 GMT
Content-Length: 1360

{
  "response": {
    "config": {
      "api.cache-control.max-age": "10",
      "certificates.polling.interval": "300000",
      "consistent.dns.routing": "true",
      "coveragezone.polling.interval": "3600000",
      "coveragezone.polling.url": "https://static.infra.ciab.
→test:443/coverage-zone.json",
      "dnssec.dynamic.response.expiration": "300s",
      "dnssec.enabled": "false",
      "domain_name": "mycdn.ciab.test",
      "federationmapping.polling.interval": "60000",
      "federationmapping.polling.url": "https://${toHostname}/
→api/4.0/federations/all",
      "geolocation.polling.interval": "86400000",
      "geolocation.polling.url": "https://static.infra.ciab.
→test:443/GeoLite2-City.mmdb.gz",
```

(continues on next page)

(continued from previous page)

```

    "keystore.maintenance.interval": "300",
    "neustar.polling.interval": "86400000",
    "neustar.polling.url": "https://static.infra.ciab.test:443/
↪neustar.tar.gz",
    "soa": {
        "admin": "twelve_monkeys",
        "expire": "604800",
        "minimum": "30",
        "refresh": "28800",
        "retry": "7200"
    },
    "steeringmapping.polling.interval": "60000",
    "ttls": {
        "A": "3600",
        "AAAA": "3600",
        "DNSKEY": "30",
        "DS": "30",
        "NS": "3600",
        "SOA": "86400"
    },
    "zonemanager.cache.maintenance.interval": "300",
    "zonemanager.threadpool.scale": "0.50"
},
"contentRouters": {
    "trafficrouter": {
        "api.port": "3333",
        "fqdn": "trafficrouter.infra.ciab.test",
        "httpsPort": 443,
        "ip": "172.26.0.15",
        "ip6": "",
        "location": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "CCR_CIAB",
        "secure.api.port": "3443",
        "status": "ONLINE"
    }
},
"contentServers": {
    "edge": {
        "cacheGroup": "CDN_in_a_Box_Edge",
        "capabilities": [
            "RAM_DISK_STORAGE"
        ],
        "fqdn": "edge.infra.ciab.test",
        "hashCount": 999,
        "hashId": "edge",

```

(continues on next page)

(continued from previous page)

```

        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.3",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "ATS_EDGE_TIER_CACHE",
        "routingDisabled": 0,
        "status": "REPORTED",
        "type": "EDGE"
    },
    "mid": {
        "cacheGroup": "CDN_in_a_Box_Mid",
        "capabilities": [
            "RAM_DISK_STORAGE"
        ],
        "fqdn": "mid.infra.ciab.test",
        "hashCount": 999,
        "hashId": "mid",
        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.4",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Mid",
        "port": 80,
        "profile": "ATS_MID_TIER_CACHE",
        "routingDisabled": 0,
        "status": "REPORTED",
        "type": "MID"
    }
},
"deliveryServices": {
    "demo1": {
        "anonymousBlockingEnabled": "false",
        "consistentHashQueryParams": [
            "abc",
            "pdq",
            "xxx",
            "zyx"
        ],
        "coverageZoneOnly": "false",
        "deepCachingType": "NEVER",
        "dispersion": {
            "limit": 1,
            "shuffled": "true"
        }
    },

```

(continues on next page)



(continued from previous page)

```

"domains": [
    "demo1.mycdn.ciab.test"
],
"ecsEnabled": "false",
"geolocationProvider": "maxmindGeolocationService",
"ip6RoutingEnabled": "true",
"matchsets": [
    {
        "matchlist": [
            {
                "match-type": "HOST",
                "regex": ".*\\.demo1\\.\\.\\.*"
            }
        ],
        "protocol": "HTTP"
    }
],
"missLocation": {
    "lat": 42,
    "long": -88
},
"protocol": {
    "acceptHttps": "true",
    "redirectToHttps": "false"
},
"regionalGeoBlocking": "false",
"requiredCapabilities": [
    "RAM_DISK_STORAGE"
],
"routingName": "video",
"soa": {
    "admin": "traffic_ops",
    "expire": "604800",
    "minimum": "30",
    "refresh": "28800",
    "retry": "7200"
},
"sslEnabled": "true",
"topology": "my-topology",
"ttls": {
    "A": "",
    "AAAA": "",
    "NS": "3600",
    "SOA": "86400"
}
}

```

(continues on next page)

(continued from previous page)

```

    },
    "edgeLocations": {
      "CDN_in_a_Box_Edge": {
        "backupLocations": {
          "fallbackToClosest": "true"
        },
        "latitude": 38.897663,
        "localizationMethods": [
          "GEO",
          "CZ",
          "DEEP_CZ"
        ],
        "longitude": -77.036574
      }
    },
    "monitors": {
      "trafficmonitor": {
        "fqdn": "trafficmonitor.infra.ciab.test",
        "httpsPort": 443,
        "ip": "172.26.0.14",
        "ip6": "",
        "location": "CDN_in_a_Box_Edge",
        "port": 80,
        "profile": "RASCAL-Traffic-Monitor",
        "status": "ONLINE"
      }
    },
    "stats": {
      "CDN_name": "CDN-in-a-Box",
      "date": 1590600715,
      "tm_host": "trafficops.infra.ciab.test:443",
      "tm_user": "admin",
      "tm_version": "development"
    },
    "topologies": {
      "my-topology": {
        "nodes": [
          "CDN_in_a_Box_Edge"
        ]
      }
    },
    "trafficRouterLocations": {
      "CDN_in_a_Box_Edge": {
        "backupLocations": {
          "fallbackToClosest": "false"
        }
      },

```

(continues on next page)

(continued from previous page)

```

        "latitude": 38.897663,
        "localizationMethods": [
            "GEO",
            "CZ",
            "DEEP_CZ"
        ],
        "longitude": -77.036574
    }
}
}
}

```

`cdns/{name}/snapshot/new`

**GET**

Retrieves the *pending Snapshot* for a CDN, which represents the current *configuration* of the CDN, **not** the current *operating state* of the CDN. The contents of this *Snapshot* are currently used by Traffic Monitor and Traffic Router.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN-SNAPSHOT:READ

**Response Type** Object

## Request Structure

Table 293: Request Path Parameters

Name	Description
name	The name of the CDN for which a <i>Snapshot</i> shall be returned

700: Request Example

```

GET /api/4.0/cdns/CDN-in-a-Box/snapshot/new HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...

```

## Response Structure

**config** An object containing basic configurations on the actual CDN object

**api.cache-control.max-age** A string containing an integer which specifies the value of `max-age` in the `Cache-Control` header of some HTTP responses, likely the *Traffic Router API* responses

**certificates.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for updated SSL certificates

**consistent.dns.routing** A string containing a boolean which indicates whether DNS routing will use a consistent hashing method or “round-robin”

“false” The “round-robin” method will be used to define DNS routing

“true” A consistent hashing method will be used to define DNS routing

**coveragezone.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should check for a new Coverage Zone file

**coveragezone.polling.url** The URL where a *Coverage Zone File* may be requested by Traffic Routers

**dnssec.dynamic.response.expiration** A string containing a number and unit suffix that specifies the length of time for which dynamic responses to DNSSEC lookup queries should remain valid

**dnssec.dynamic.concurrencylevel** An integer that defines the size of the concurrency level (threads) of the Guava cache used by ZoneManager to store zone material

**dnssec.dynamic.initialcapacity** An integer that defines the initial size of the Guava cache, default is 10000. Too low of a value can lead to expensive resizing

**dnssec.init.timeout** An integer that defines the number of minutes to allow for zone generation, this bounds the zone priming activity

**dnssec.enabled** A string that tells whether or not the CDN uses DNSSEC; one of:

“false” DNSSEC is not used within this CDN

“true” DNSSEC is used within this CDN

- domain\_name** A string that is the TLD served by the CDN
- federationmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new federation mappings
- federationmapping.polling.url** The URL where Traffic Control components can request federation mappings
- geolocation.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new IP-to-geographic-location mapping databases
- geolocation.polling.url** The URL where Traffic Control components can request IP-to-geographic-location mapping database files
- keystore.maintenance.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Routers should refresh their zone caches
- neustar.polling.interval** A string containing an integer which specifies the interval, in seconds, on which other Traffic Control components should check for new “Neustar” databases
- neustar.polling.url** The URL where Traffic Control components can request “Neustar” databases
- soa** An object defining the SOA for the CDN’s TLD (defined in `domain_name`)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which

secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than `refresh`.

---

**See also:**

[The Wikipedia page on Start of Authority records.](#)

**steeringmapping.polling.interval** A string containing an integer which specifies the interval, in seconds, on which Traffic Control components should check for new steering mappings

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

**zonemanager.cache.maintenance.interval** A configuration option for the ZoneManager Java class of Traffic Router

**zonemanager.threadpool.scale** A configuration option for the ZoneManager Java class of Traffic Router

**contentRouters** An object containing keys which are the (short) hostnames of the Traffic Routers that serve requests for *Delivery Services* in this CDN

**api.port** A string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTP

**secure.api.port** An optionally present string containing the port number on which the *Traffic Router API* is served by this Traffic Router via HTTPS

**fqdn** This Traffic Router's FQDN

**httpsPort** The port number on which this Traffic Router listens for incoming HTTPS requests

**ip** This Traffic Router's IPv4 address

**ip6** This Traffic Router's IPv6 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Router belongs

**port** The port number on which this Traffic Router listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by this Traffic Router

**status** The health status of this Traffic Router

**See also:**

*Health Protocol*

**contentServers** An object containing keys which are the (short) hostnames of the *Edge-tier cache servers* in the CDN; the values corresponding to those keys are routing information for said servers

**cacheGroup** A string that is the *Name* of the *Cache Group* to which the server belongs

**capabilities** An array of this *Cache Server's Server Capabilities*. If the Cache Server has no Server Capabilities, this field is omitted.

**deliveryServices** An object containing keys which are the names of *Delivery Services* to which this *cache server* is assigned; the values corresponding to those keys are arrays of FQDNs that resolve to this *cache server*

---

**Note:** Only *Edge-tier cache servers* can be assigned to a *Delivery Service*, and therefore this field will only be present when **type** is "EDGE".

---

**fqdn** The server's FQDN

**hashCount** The number of servers to be placed into a single "hash ring" in Traffic Router

**hashId** A unique string to be used as the key for hashing servers - as of version 3.0.0 of Traffic Control, this is always the same as the server's (short) hostname and only still exists for legacy compatibility reasons

**httpsPort** The port on which the *cache server* listens for incoming HTTPS requests

**interfaceName** The name of the main network interface device used by this *cache server*

**ip6** The server's IPv6 address

**ip** The server's IPv4 address

**locationId** This field is exactly the same as **cacheGroup** and only exists for legacy compatibility reasons

**port** The port on which this *cache server* listens for incoming HTTP requests

**profile** The *Name* of the *Profile* used by the *cache server*

**routingDisabled** An integer representing the boolean concept of whether or not Traffic Routers should route client traffic to this *cache server*; one of:

**0** Do not route traffic to this server

**1** Route traffic to this server normally

**status** This *cache server*'s status

**See also:**

*Health Protocol*

**type** The *Type* of this *cache server*; which ought to be one of (but in practice need not be in certain special circumstances):

**EDGE** This is an *Edge-tier cache server*

**MID** This is a *Mid-tier cache server*

**deliveryServices** An object containing keys which are the *xml\_ids* of all of the *Delivery Services* within the CDN

**anonymousBlockingEnabled** A string containing a boolean that tells whether or not *Anonymous Blocking* is set on this *Delivery Service*; one of:

**“true”** Anonymized IP addresses are blocked by this *Delivery Service*

**“false”** Anonymized IP addresses are not blocked by this *Delivery Service*

**See also:**

*Configure Anonymous Blocking*

**consistentHashQueryParameters** A set of query parameters that Traffic Router should consider when determining a consistent hash for a given client request.

**consistentHashRegex** An optional regular expression that will ensure clients are consistently routed to a *cache server* based on matches to it.

**coverageZoneOnly** A string containing a boolean that tells whether or not this *Delivery Service* routes traffic based only on its *Coverage Zone File*



See also:

*Geo Limit*

**deepCachingType** A string that defines the *Deep Caching* setting of this *Delivery Service*

**dispersion** An object describing the “dispersion” - or number of *cache servers* within a single *Cache Group* across which the same content is spread - within the *Delivery Service*

**limit** The maximum number of *cache servers* in which the response to a single request URL will be stored

---

**Note:** If this is greater than the number of *cache servers* in the *Cache Group* chosen to service the request, then content will be spread across all of them. That is, it causes no problems.

---

**shuffled** A string containing a boolean that tells whether the *cache servers* chosen for content dispersion are chosen randomly or based on a consistent hash of the request URL; one of:

“false” *cache servers* will be chosen consistently

“true” *cache servers* will be chosen at random

**domains** An array of domains served by this *Delivery Service*

**ecsEnabled** A string containing a boolean from *EDNS0 Client Subnet Enabled* that tells whether EDNS0 client subnet is enabled on this *Delivery Service*; one of:

“false” EDNS0 client subnet is not enabled on this *Delivery Service*

“true” EDNS0 client subnet is enabled on this *Delivery Service*

**geolocationProvider** The name of a provider for IP-to-geographic-location mapping services - currently the only valid value is "maxmindGeolocationService"

**ip6RoutingEnabled** A string containing a boolean that defines the *IPv6 Routing Enabled* setting for this *Delivery Service*; one of:

“false” IPv6 traffic will not be routed by this *Delivery Service*

“true” IPv6 traffic will be routed by this *Delivery Service*

**matchList** An array of methods used by Traffic Router to determine whether or not a request can be serviced by this *Delivery Service*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integral, unique identifier for the set of types to which the **type** field belongs

**type** The name of the *Type* of match performed using **pattern** to determine whether or not to use this *Delivery Service*

**HOST\_REGEX** Use the *Delivery Service* if **pattern** matches the *Host* HTTP header of an HTTP request, or the name requested for resolution in a DNS request

**HEADER\_REGEX** Use the *Delivery Service* if **pattern** matches an HTTP header (both the name and value) in an HTTP request<sup>1</sup>

**PATH\_REGEX** Use the *Delivery Service* if **pattern** matches the request path of this *Delivery Service*'s URL Page 1040, 1

**STEERING\_REGEX** Use the *Delivery Service* if **pattern** matches the *xml\_id* of one of this *Delivery Service*'s “Steering” target *Delivery Services*

**missLocation** An object representing the default geographic coordinates to use for a client when lookup of their IP has failed in both the *Coverage Zone File* (and/or possibly the *Deep Coverage Zone File*) and the IP-to-geographic-location database

**lat** Geographic latitude as a floating point number

---

<sup>1</sup> These only apply to HTTP-routed *Delivery Services*

**long** Geographic longitude as a floating point number

**protocol** An object that describes how the *Delivery Service* ought to handle HTTP requests both with and without TLS encryption

**acceptHttps** A string containing a boolean that tells whether HTTPS requests should be normally serviced by this *Delivery Service*; one of:

**“false”** Refuse to service HTTPS requests

**“true”** Service HTTPS requests normally

**redirectToHttps** A string containing a boolean that tells whether HTTP requests ought to be re-directed to use HTTPS; one of:

**“false”** Do not redirect unencrypted traffic; service it normally

**“true”** Respond to HTTP requests with instructions to use HTTPS instead

**See also:**

*Protocol*

**regionalGeoBlocking** A string containing a boolean that defines the *Regional Geoblocking* setting of this *Delivery Service*; one of:

**“false”** Regional Geographic Blocking is not used by this *Delivery Service*

**“true”** Regional Geographic Blocking is used by this *Delivery Service*

**See also:**

*Configure Regional Geo-blocking (RGB)*

**requiredCapabilities** An array of this *Delivery Service*’s *required capabilities*. If there are no required capabilities, this field is omitted.

**routingName** A string that is this *Delivery Service*’s *Routing Name*

**soa** An object defining the SOA record for the *Delivery Service*'s TLDs (defined in domains)

**admin** The name of the administrator for this zone - i.e. the RNAME

---

**Note:** This rarely represents a proper email address, unfortunately.

---

**expire** A string containing an integer that sets the number of seconds after which secondary name servers should stop answering requests for this zone if the master does not respond

**minimum** A string containing an integer that sets the TTL - in seconds - of the record for the purpose of negative caching

**refresh** A string containing an integer that sets the number of seconds after which secondary name servers should query the master for the SOA record, to detect zone changes

**retry** A string containing an integer that sets the number of seconds after which secondary name servers should retry to request the serial number from the master if the master does not respond

---

**Note:** **RFC 1035** dictates that this should always be less than **refresh**.

---

**See also:**

[The Wikipedia page on Start of Authority records.](#)

**sslEnabled** A string containing a boolean that tells whether this *Delivery Service* uses SSL; one of:

**“false”** SSL is not used by this *Delivery Service*

**“true”** SSL is used by this *Delivery Service*

**See also:**

*Protocol*

**topology** The name of the *Topology* that this *Delivery Service* is assigned to. If the Delivery Service is not assigned to a

topology, this field is omitted.

**ttls** An object that contains keys which are types of DNS records that have values which are strings containing integers that specify the time for which a response to the specific type of record request should remain valid

---

**Note:** This overrides `config.ttls`.

---

**edgeLocations** An object containing keys which are the names of Edge-Tier *Cache Groups* within the CDN

**backupLocations** An object that describes this *Cache Group's Fallbacks*

**fallbackToClosest** A string containing a boolean which defines the *Fallback to Closest* behavior of this *Cache Group*; one of:

**“false”** Do not fall back on the closest available *Cache Group*

**“true”** Fall back on the closest available *Cache Group*

**list** If this *Cache Group* has any *Fallbacks*, this key will appear and will be an array of those *Cache Groups' Names*

**latitude** A floating point number that defines this *Cache Group's Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

**monitors** An object containing keys which are the (short) hostnames of Traffic Monitors within this CDN

**fqdn** The FQDN of this Traffic Monitor

**httpsPort** The port number on which this Traffic Monitor listens for incoming HTTPS requests

**ip6** This Traffic Monitor's IPv6 address

**ip** This Traffic Monitor's IPv4 address

**location** A string which is the *Name* of the *Cache Group* to which this Traffic Monitor belongs

**port** The port number on which this Traffic Monitor listens for incoming HTTP requests

**profile** A string which is the *Name* of the *Profile* used by this Traffic Monitor

---

**Note:** For legacy reasons, this must always start with “RASCAL-“.

---

**status** The health status of this Traffic Monitor

**See also:**

*Health Protocol*

**stats** An object containing metadata information regarding the CDN

**CDN\_name** The name of this CDN

**date** The UNIX epoch timestamp date in the Traffic Ops server’s own timezone

**tm\_host** The FQDN of the Traffic Ops server

**tm\_user** The username of the currently logged-in user

**tm\_version** The full version number of the Traffic Ops server, including release number, git commit hash, and supported Enterprise Linux version

**topologies** An array of *Topologies* where each key is the name of that Topology.

**nodes** An array of the names of the *Edge-Tier Cache Groups* in this *Topology*. *Mid-Tier* Cache Groups in the topology are not included.

**trafficRouterLocations** An object containing keys which are the *names of Cache Groups* within the CDN which contain Traffic Routers

**backupLocations** An object that describes this *Cache Group’s Fallbacks*

**fallbackToClosest** A string containing a boolean which defines this *Cache Group’s Fallback to Closest* setting; one of:

**“false”** Do not fall back on the closest available *Cache Group*

**“true”** Fall back on the closest available *Cache Group*

**latitude** A floating point number that defines this *Cache Group’s Latitude*

**localizationMethods** An array of strings that represents this *Cache Group's Localization Methods*

**longitude** A floating point number that defines this *Cache Group's Longitude*

#### 701: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 27 May 2020 20:31:13
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
    ↳M6uhE2oPpjpTUR7gALsPOnM2CepD+VCAjp4dj5Xnppo0G5zL31PQgiteD23q67r7/bq/
    ↳JJpMvIvdaENVYFtrqQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 27 May 2020 19:31:13 GMT
Content-Length: 1374

{
  "response": {
    "config": {
      "api.cache-control.max-age": "10",
      "certificates.polling.interval": "3000000",
      "consistent.dns.routing": "true",
      "coveragezone.polling.interval": "36000000",
      "coveragezone.polling.url": "https://static.infra.ciab.
    ↳test:443/coverage-zone.json",
      "dnssec.dynamic.response.expiration": "300s",
      "dnssec.enabled": "false",
      "domain_name": "mycdn.ciab.test",
      "federationmapping.polling.interval": "60000",
      "federationmapping.polling.url": "https://${toHostname}/
    ↳api/4.0/federations/all",
      "geolocation.polling.interval": "864000000",
      "geolocation.polling.url": "https://static.infra.ciab.
    ↳test:443/GeoLite2-City.mmdb.gz",
      "keystore.maintenance.interval": "300",
      "neustar.polling.interval": "864000000",
      "neustar.polling.url": "https://static.infra.ciab.test:443/
    ↳neustar.tar.gz",
```

(continues on next page)

(continued from previous page)

```

    "soa": {
      "admin": "twelve_monkeys",
      "expire": "604800",
      "minimum": "30",
      "refresh": "28800",
      "retry": "7200"
    },
    "steeringmapping.polling.interval": "60000",
    "ttls": {
      "A": "3600",
      "AAAA": "3600",
      "DNSKEY": "30",
      "DS": "30",
      "NS": "3600",
      "SOA": "86400"
    },
    "zonemanager.cache.maintenance.interval": "300",
    "zonemanager.threadpool.scale": "0.50"
  },
  "contentServers": {
    "edge": {
      "cacheGroup": "CDN_in_a_Box_Edge",
      "capabilities": [
        "RAM_DISK_STORAGE"
      ],
      "fqdn": "edge.infra.ciab.test",
      "hashCount": 999,
      "hashId": "edge",
      "httpsPort": 443,
      "interfaceName": "eth0",
      "ip": "172.26.0.3",
      "ip6": "",
      "locationId": "CDN_in_a_Box_Edge",
      "port": 80,
      "profile": "ATS_EDGE_TIER_CACHE",
      "status": "REPORTED",
      "type": "EDGE",
      "routingDisabled": 0
    },
    "mid": {
      "cacheGroup": "CDN_in_a_Box_Mid",
      "capabilities": [
        "RAM_DISK_STORAGE"
      ],
      "fqdn": "mid.infra.ciab.test",
      "hashCount": 999,

```

(continues on next page)



(continued from previous page)

```

        "hashId": "mid",
        "httpsPort": 443,
        "interfaceName": "eth0",
        "ip": "172.26.0.4",
        "ip6": "",
        "locationId": "CDN_in_a_Box_Mid",
        "port": 80,
        "profile": "ATS_MID_TIER_CACHE",
        "status": "REPORTED",
        "type": "MID",
        "routingDisabled": 0
    },
    },
    "contentRouters": {
        "trafficrouter": {
            "api.port": "3333",
            "fqdn": "trafficrouter.infra.ciab.test",
            "httpsPort": 443,
            "ip": "172.26.0.15",
            "ip6": "",
            "location": "CDN_in_a_Box_Edge",
            "port": 80,
            "profile": "CCR_CIAB",
            "secure.api.port": "3443",
            "status": "ONLINE"
        }
    },
    "deliveryServices": {
        "demo1": {
            "anonymousBlockingEnabled": "false",
            "consistentHashQueryParams": [
                "abc",
                "pdq",
                "xxx",
                "zyx"
            ],
            "coverageZoneOnly": "false",
            "deepCachingType": "NEVER",
            "dispersion": {
                "limit": 1,
                "shuffled": "true"
            },
            "domains": [
                "demo1.mycdn.ciab.test"
            ],
            "ecsEnabled": "false",

```

(continues on next page)

(continued from previous page)

```

    "geolocationProvider": "maxmindGeolocationService",
    "ip6RoutingEnabled": "true",
    "matchsets": [
      {
        "protocol": "HTTP",
        "matchlist": [
          {
            "regex": ".*\\.demo1\\.\\.\\.*",
            "match-type": "HOST"
          }
        ]
      }
    ],
    "missLocation": {
      "lat": 42,
      "long": -88
    },
    "protocol": {
      "acceptHttps": "true",
      "redirectToHttps": "false"
    },
    "regionalGeoBlocking": "false",
    "requiredCapabilities": [
      "RAM_DISK_STORAGE"
    ],
    "routingName": "video",
    "soa": {
      "admin": "traffic_ops",
      "expire": "604800",
      "minimum": "30",
      "refresh": "28800",
      "retry": "7200"
    },
    "sslEnabled": "true",
    "topology": "my-topology",
    "ttls": {
      "A": "",
      "AAAA": "",
      "NS": "3600",
      "SOA": "86400"
    }
  },
  "edgeLocations": {
    "CDN_in_a_Box_Edge": {
      "latitude": 38.897663,

```

(continues on next page)

(continued from previous page)

```

        "longitude": -77.036574,
        "backupLocations": {
            "fallbackToClosest": "true"
        },
        "localizationMethods": [
            "GEO",
            "CZ",
            "DEEP_CZ"
        ]
    },
    "trafficRouterLocations": {
        "CDN_in_a_Box_Edge": {
            "latitude": 38.897663,
            "longitude": -77.036574,
            "backupLocations": {
                "fallbackToClosest": "false"
            },
            "localizationMethods": [
                "GEO",
                "CZ",
                "DEEP_CZ"
            ]
        }
    },
    "monitors": {
        "trafficmonitor": {
            "fqdn": "trafficmonitor.infra.ciab.test",
            "httpsPort": 443,
            "ip": "172.26.0.14",
            "ip6": "",
            "location": "CDN_in_a_Box_Edge",
            "port": 80,
            "profile": "RASCAL-Traffic-Monitor",
            "status": "ONLINE"
        }
    },
    "stats": {
        "CDN_name": "CDN-in-a-Box",
        "date": 1590607873,
        "tm_host": "trafficops.infra.ciab.test:443",
        "tm_user": "admin",
        "tm_version": "development"
    },
    "topologies": {
        "my-topology": {

```

(continues on next page)

(continued from previous page)

```

        "nodes": [
            "CDN_in_a_Box_Edge"
        ]
    }
}

```

## cdns/routing

### GET

Retrieves the aggregated routing percentages across all CDNs. This is accomplished by polling stats from all online Traffic Routers via the `/crs/stats` route.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Object

## Request Structure

No parameters available

## Response Structure

**cz** The percent of requests to online Traffic Routers that were satisfied by a *Coverage Zone File*

**deepCz** The percent of requests to online Traffic Routers that were satisfied by a *Deep Coverage Zone File*

**dsr** The percent of requests to online Traffic Routers that were satisfied by sending the client to an overflow *Delivery Service*

**err** The percent of requests to online Traffic Routers that resulted in an error

**fed** The percent of requests to online Traffic Routers that were satisfied by sending the client to a federated CDN

**geo** The percent of requests to online Traffic Routers that were satisfied using 3rd party geographic IP mapping

**miss** The percent of requests to online Traffic Routers that could not be satisfied

**regionalAlternate** The percent of requests to online Traffic Routers that were satisfied by sending the client to the alternate, Regional Geo-blocking URL

**regionalDenied** The percent of requests to online Traffic Routers that were denied due to geographic location policy

**staticRoute** The percent of requests to online Traffic Routers that were satisfied with *Static DNS Entries*

#### 702: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 14 Nov 2018 21:29:32 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→7LjytwKyRzSKM4cRIol40MIJxApFpTWJaSK73rbtUIQdASZjI64XxLVzZP00GRU7XeJ22YKUyQ30qbKHDP
Content-Length: 130

{ "response": {
  "cz": 79,
  "deepCz": 0.50,
  "dsr": 0,
  "err": 0,
  "fed": 0.25,
  "geo": 20,
  "miss": 0.25,
  "regionalAlternate": 0,
  "regionalDenied": 0,
  "staticRoute": 0
}}
```

### consistenthash

Test Pattern-Based Consistent Hashing for a *Delivery Service* using a regular expression and a request path

#### POST

Queries database for an active Traffic Router on a given CDN and sends GET request to get the resulting path to consistent hash with a given regex and request path.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Object

#### Request Structure

**regex** The regular expression to apply to the request path to get a resulting path that will be used for consistent hashing

**requestPath** The request path to use to test the regular expression against

**cdnId** The unique identifier of a CDN that will be used to query for an active Traffic Router

### 703: Request Example

```
POST /api/4.0/consistenthash HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.54.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 80
Content-Type: application/x-www-form-urlencoded

{"regex":"/.*?(/.*?/).*(m3u8)","requestPath":"/test/path/asset.m3u8",
  ↪ "cdnId":2}
```

## Response Structure

**resultingPathToConsistentHash** The resulting path that Traffic Router will use for consistent hashing

**consistentHashRegex** The regex used by Traffic Router derived from POST 'regex' parameter

**requestPath** The request path used by Traffic Router to test regex against

### 704: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, ↵
  ↪ Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54 ↵
  ↪ GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ↵
  ↪ QMDFOnUfqH4TcZ4YnUQyqnXDier0YiUMIfwBGDcT7ySjw9uASBGsLQW35lpnKFi4as0vYlHuSSGpe4hHG
X-Server-Name: traffic_ops_golang/
Date: Tue, 12 Feb 2019 21:32:05 GMT
Content-Length: 142

{ "response": {
  "resultingPathToConsistentHash":"/path/m3u8",
  "consistentHashRegex":"/.*?(/.*?/).*(m3u8)",
  "requestPath":"/test/path/asset.m3u8"
}}
```

## coordinates

### GET

Gets a list of all coordinates in the Traffic Ops database

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** COORDINATE:READ

**Response Type** Array

### Request Structure

Table 294: Request Query Parameters

Name	Re-quired	Description
id	no	Return only coordinates that have this integral, unique identifier
name	no	Return only coordinates with this name
or-derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### Response Structure

**id** Integral, unique, identifier for this coordinate pair

**lastUpdated** The time and date at which this entry was last updated, in a ctime-like format

**latitude** Latitude of the coordinate

**longitude** Longitude of the coordinate

**name** The name of the coordinate - typically this just reflects the name of the Cache Group for which the coordinate was created



705: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Y2vxC3hpxIg6aRNBBT7i2hbAViIJp+dJoqHIzu3acFM+vGay/
    ↳I5E+eZYOC9RY8hcJPrKNXysZOD8DOb9KsFgaw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:32:28 GMT
Content-Length: 942

{ "response": [
  {
    "id": 1,
    "name": "from_cachegroup_TRAFFIC_ANALYTICS",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 2,
    "name": "from_cachegroup_TRAFFIC_OPS",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 3,
    "name": "from_cachegroup_TRAFFIC_OPS_DB",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 4,
    "name": "from_cachegroup_TRAFFIC_PORTAL",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {

```

(continues on next page)

(continued from previous page)

```
    "id": 5,
    "name": "from_cachegroup_TRAFFIC_STATS",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 6,
    "name": "from_cachegroup_CDN_in_a_Box_Mid",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:04+00"
  },
  {
    "id": 7,
    "name": "from_cachegroup_CDN_in_a_Box_Edge",
    "latitude": 38.897663,
    "longitude": -77.036574,
    "lastUpdated": "2018-10-24 16:07:05+00"
  }
}
```

## POST

Creates a new coordinate pair

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** COORDINATE:CREATE, COORDINATE:READ

**Response Type** Object

## Request Structure

**name** The name of the new coordinate

**latitude** The desired latitude of the new coordinate (must be on the interval [-180, 180])

**longitude** The desired longitude of the new coordinate (must be on the interval [-90, 90])

#### 706: Request Example

```
POST /api/4.0/coordinates HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 47
Content-Type: application/json

{"name": "test", "latitude": 0, "longitude": 0}
```

### Response Structure

- id** Integral, unique, identifier for the newly created coordinate pair
- lastUpdated** The time and date at which this entry was last updated, in a ctime-like format
- latitude** Latitude of the newly created coordinate
- longitude** Longitude of the newly created coordinate
- name** The name of the coordinate

#### 707: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 7pWdeZyIIXE1P7o/
→JVon+5eSCbDw+FGamAzdXzWHXJ8IhF+Vh+/
→tWfCkzHYw3rP2kBVwZu+gqLffjQpBCMjt7A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 17:48:55 GMT
Content-Length: 165

{ "alerts": [
  {
    "text": "coordinate was created.",
    "level": "success"
  }
]}
```

(continues on next page)

(continued from previous page)

```
],
"response": {
  "id": 9,
  "name": "test",
  "latitude": 0,
  "longitude": 0,
  "lastUpdated": "2018-11-15 17:48:55+00"
}}
```

## PUT

Updates a coordinate

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** COORDINATE:UPDATE, COORDINATE:READ

**Response Type** Object

## Request Structure

Table 295: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the coordinate to edit

**name** The name of the new coordinate

**latitude** The desired new latitude of the coordinate (must be on the interval [-180, 180])

**longitude** The desired new longitude of the coordinate (must be on the interval [-90, 90])

### 708: Request Example

```
PUT /api/4.0/coordinates?id=9 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 48
Content-Type: application/json

{"name": "quest", "latitude": 0, "longitude": 0}
```

## Response Structure

- id** Integral, unique, identifier for the coordinate pair
- lastUpdated** The time and date at which this entry was last updated, in a ctime-like format
- latitude** Latitude of the coordinate
- longitude** Longitude of the coordinate
- name** The name of the coordinate

### 709: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳zd03Uvbnv8EbSZZ75Xp5tnnYStZsZTdyPxXnoqK4QZ5WhELLPL8iHlRf0aiLTbrUWUeJ8ue2HRz6aBS/
    ↳iXCCGA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 17:54:30 GMT
Content-Length: 166

{ "alerts": [
  {
    "text": "coordinate was updated.",
    "level": "success"
  }
],
"response": {
  "id": 9,
  "name": "quest",
  "latitude": 0,
  "longitude": 0,
  "lastUpdated": "2018-11-15 17:54:30+00"
}}
```

## DELETE

Deletes a coordinate

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** COORDINATE:DELETE, COORDINATE:READ

**Response Type** undefined

## Request Structure

Table 296: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the coordinate to delete

## Response Structure

710: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
↪GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 82x/Wdckqgk4LN5Ll1ZfBJ26xkDrUVUGDjs5QFa/
↪Lzap7dU3OZkjb8XW41xeFYj8PDmxHIpb7hiV0bvLaxnEDA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 15 Nov 2018 17:57:42 GMT
Content-Length: 65

{ "alerts": [
  {
    "text": "coordinate was deleted.",
    "level": "success"
  }
]}
```

## current\_stats

An API endpoint that returns current statistics for each CDN and an aggregate across them all.

### GET

Retrieves current stats for each CDN. Also includes aggregate stats across them.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CDN:READ

**Response Type** Array

## Request Structure

No parameters available.

### 711: Request Example

```
GET /api/4.0/current_stats HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cdn** The name of the CDN

**connections** Current number of TCP connections maintained

**capacity** 85 percent capacity of the CDN in Gps

**bandwidth** The total amount of bandwidth in Gbs

---

**Note:** If `cdn` name is `total` and `capacity` is omitted it represents the aggregate stats across CDNs

---

### 712: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=; Path=/; HttpOnly
Whole-Content-Sha512:
→Rs3wgd7v5dP0b0Qs4I3J1q6mnWIMSM2AKSAWirK1kymvDY0oFISArF7Kkypgy10I34yn7FtFdMh6U7ABa
X-Server-Name: traffic_ops_golang/
Date: Thu, 14 Nov 2019 15:35:31 GMT
Content-Length: 138

{"response": {
  "currentStats": [
    {
      "bandwidth": null,
      "capacity": null,
      "cdn": "ALL",
      "connections": null
    },
    {
      "bandwidth": 0.000104,
      "capacity": 17,
      "cdn": "CDN-in-a-Box",
      "connections": 4
    },
    {
      "bandwidth": 0.000104,
      "cdn": "total",
      "connections": 4
    }
  ]
}}
```

## dbdump

**Caution:** This is an extremely dangerous thing to do, as it exposes the entirety of the database, including possibly sensitive information. Administrators and systems engineers are advised to instead use database-specific tools to make server transitions more securely.

Dumps the Traffic Ops database as an SQL script that should recreate its schema and contents exactly.

## Implementation Detail

The script is output using the *pg\_dump(1)* utility, and is thus compatible for use with the



`pg_restore(1)` utility.

## GET

Fetches the database dump.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** DBDUMP:READ

**Response Type** undefined - outputs an SQL script, not JSON

## Request Structure

No parameters available

713: Request Example

```
GET /api/4.0/dbdump HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

714: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/sql
Content-Disposition: attachment
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→YwvPB0ZToyzT8ilBnDlWWdwV+E3f2Xgus10KrkNaipQqgrw5zGwq0rC1U9TZ8Zl6kAGcRZgCYnr1EWfHX
X-Server-Name: traffic_ops_golang/
Date: Mon, 09 Sep 2019 21:08:28 GMT
Transfer-Encoding: chunked
```

(continues on next page)

(continued from previous page)

```
-- Actual text omitted - it's huge
```

## deliveryservice\_request\_comments

### GET

Gets delivery service request comments.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DS-REQUEST:READ, DELIVERY-SERVICE:READ,  
USER:READ

**Response Type** Array

### Request Structure

Table 297: Request Query Parameters

Name	Re- quired	Description
author	no	Filter for <i>Delivery Service Request</i> comments submitted by the user identified by this username
authorId	no	Filter for <i>Delivery Service Request</i> comments submitted by the user identified by this integral, unique identifier
deliveryServiceRequestId	no	Filter for <i>Delivery Service Request</i> comments submitted for the delivery service identified by this integral, unique identifier
id	no	Filter for the <i>Delivery Service Request</i> comment identified by this integral, unique identifier

#### 715: Request Example

```
GET /api/4.0/deliveryservice_request_comments HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

- author** The username of the user who created the comment.
- authorId** The integral, unique identifier of the user who created the comment.
- deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.
- id** The integral, unique identifier of the *DSR* comment.
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- value** The text of the comment that was posted.
- xmlId** This is the xmlId value that you provided in the request.

#### 716: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:00:26.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→RaJZS1XFJ4oIxVKyyDjTuoQY7gP0mm5EuIL4AgHpyQpuaaNviw0XhGC4V/AKf/
→Ws6zXLgIUc40yvMsTxnrlww==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:00:26 GMT
Content-Length: 207

{
  "response": [
    {
```

(continues on next page)

(continued from previous page)

```

        "authorId": 2,
        "author": "admin",
        "deliveryServiceRequestId": 2,
        "id": 3,
        "lastUpdated": "2020-02-24 19:59:46+00",
        "value": "Changing to a different origin for now.",
        "xmlId": "demo1"
    },
    {
        "authorId": 2,
        "author": "admin",
        "deliveryServiceRequestId": 2,
        "id": 4,
        "lastUpdated": "2020-02-24 19:59:55+00",
        "value": "Using HTTPS.",
        "xmlId": "demo1"
    }
]
}

```

## POST

Allows user to create a *Delivery Service Request* comment.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:UPDATE, DELIVERY-SERVICE:READ, USER:READ

**Response Type** Object

## Request Structure

**deliveryServiceRequestId** The integral, unique identifier of the delivery service that you are commenting on.

**value** The comment text itself.

**xmlId** This can be any string. It is not validated or used, though it is returned in the response.

717: Request Example

```

POST /api/4.0/deliveryservice_request_comments HTTP/1.1
User-Agent: python-requests/2.22.0

```

(continues on next page)

(continued from previous page)

```
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 111

{
  "deliveryServiceRequestId": 2,
  "value": "Does anyone have time to review my delivery service_
→request?"
}
```

## Response Structure

**author** The username of the user who created the comment.

**authorId** The integral, unique identifier of the user who created the comment.

**deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.

**id** The integral, unique identifier of the *DSR* comment.

**lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*

**value** The text of the comment that was posted.

**xmlId** This is the xmlId value that you provided in the request.

### 718: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:02:20
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→LiakFP6L7PrnF05kLXftx7WQoKn3bGpIJT5N15PvNG2sHridRMV3k23eRJM66ET0LcRfMOrQgRiydE+Xg
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:02:20 GMT
Content-Length: 223
```

(continues on next page)

(continued from previous page)

```
{
  "alerts": [
    {
      "text": "deliveryservice_request_comment was created.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": 2,
    "author": null,
    "deliveryServiceRequestId": 2,
    "id": 6,
    "lastUpdated": "2020-02-24 20:02:20+00",
    "value": "Does anyone have time to review my delivery service_
↪request?",
    "xmlId": null
  }
}
```

## PUT

Updates a delivery service request comment.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:UPDATE, DELIVERY-SERVICE:READ,  
USER:READ

**Response Type** Object

## Request Structure

**deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.

**value** The comment text itself.

**xmlId** This can be any string. It is not validated or used, though it is returned in the response.

Table 298: Request Query Parameters

Parameter	Required	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> comment that you wish to update.

#### 719: Request Example

```
PUT /api/4.0/deliveryservice_request_comments?id=6 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 166

{
  "deliveryServiceRequestId": 2,
  "value": "Update: We no longer need this, feel free to reject.\n\
  ↪nDoes anyone have time to review my delivery service request?"
}
```

### Response Structure

**author** The username of the user who created the comment.

**authorId** The integral, unique identifier of the user who created the comment.

**deliveryServiceRequestId** The integral, unique identifier of the *Delivery Service Request* that the comment was posted on.

**id** The integral, unique identifier of the *DSR* comment.

**lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*

**value** The text of the comment that was posted.

**xmlId** This is the xmlId value that you provided in the request.

#### 720: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, ↪
  ↪Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:05:46 ↪
  ↪GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ↪
  ↪RalS34imPw7c42nlnu5eTuv6FSxuGcAvxEdeIyNma1zpE3ZojAMFbhj8qi1s+hOVDYybfFPzMz82c+xc1
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

(continued from previous page)

Date: Mon, 24 Feb 2020 20:05:46 GMT

Content-Length: 255

```
{
  "alerts": [
    {
      "text": "deliveryservice_request_comment was updated.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": null,
    "author": null,
    "deliveryServiceRequestId": 2,
    "id": 6,
    "lastUpdated": "2020-02-24 20:05:46+00",
    "value": "Update: We no longer need this, feel free to reject.\n\nDoes anyone have time to review my delivery service request?",
    "xmlId": null
  }
}
```

## DELETE

Deletes a delivery service request comment.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:UPDATE, DELIVERY-SERVICE:READ, USER:READ

**Response Type** undefined

## Request Structure

Table 299: Request Query Parameters

Parameter	Required	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> comment that you wish to delete.



#### 721: Request Example

```
DELETE /api/4.0/deliveryservice_request_comments?id=6 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

#### 722: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 21:07:40
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 10pGzqeIh/1JAx85mz3MI/
→5A1i1g5beTSLtfvgcfQmCjNKQvOMs/4TLviuVzOCRRrEIPmNcjy35tmvfxwlv7RMQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 20:07:40 GMT
Content-Length: 101

{
  "alerts": [
    {
      "text": "deliveryservice_request_comment was deleted.",
      "level": "success"
    }
  ]
}
```

## deliveryservice\_requests

### GET

Retrieves *Delivery Service Requests*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DS-REQUEST:READ, DELIVERY-SERVICE:READ,  
USER:READ

**Response Type** Array

## Request Structure

Table 300: Request Query Parameters

Name	Re-quired	Description
as-signee	no	Filter for <i>Delivery Service Requests</i> that are assigned to the user identified by this username.
as-signeeId	no	Filter for <i>Delivery Service Requests</i> that are assigned to the user identified by this integral, unique identifier
author	no	Filter for <i>Delivery Service Requests</i> submitted by the user identified by this username
authorId	no	Filter for <i>Delivery Service Requests</i> submitted by the user identified by this integral, unique identifier
changeType	no	Filter for <i>Delivery Service Requests</i> of the change type specified. Can be create, update, or delete.
createdAt	no	Filter for <i>Delivery Service Requests</i> created on a certain date/time. Value must be <b>RFC 3339</b> compliant. Eg. 2019-09-19T19:35:38.828535Z
id	no	Filter for the <i>Delivery Service Requests</i> identified by this integral, unique identifier.
status	no	Filter for <i>Delivery Service Requests</i> whose status is the status specified. The status can be draft, submitted, pending, rejected, or complete.
xmlId	no	Filter for <i>Delivery Service Requests</i> that have the given <i>xml_id</i> .
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n</i> <sup>th</sup> page of results, where “ <i>n</i> ” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

New in version ATCv6: The *createdAt* query parameter was added to this in endpoint across all API versions in ATC version 6.0.0.

### 723: Request Example

```
GET /api/4.0/deliveryservice_requests?status=draft HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

The response is an array of representations of *Delivery Service Requests*.

### 724: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:14:07
    ↳GMT; Max-Age=3600; HttpOnly
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:14:07 GMT
Content-Length: 872

{ "response": [{
  "authorId": 2,
  "author": "admin",
  "changeType": "update",
  "createdAt": "2020-02-24 19:11:12+00",
  "id": 1,
  "lastEditedBy": "admin",
  "lastEditedById": 2,
  "lastUpdated": "2020-02-24 19:11:12+00",
  "requested": {
    "active": false,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "displayName": "Demo 1",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "firstHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
```

(continues on next page)

(continued from previous page)

```

"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "0001-01-01T00:00:00Z",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"topology": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [

```

(continues on next page)

(continued from previous page)

```

        "http://video.demo1.mycdn.ciab.test",
        "https://video.demo1.mycdn.ciab.test"
    ],
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": "root",
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": [
        "abc",
        "pdq",
        "xxx",
        "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "tlsVersions": null
},
"status": "draft"
}}}

```

## POST

Creates a new *Delivery Service Request*. “Closed” *Delivery Service Requests* cannot be created, an existing *Delivery Service Request* must be placed into a closed *Status*. A *Delivery Service Request* to create, modify or delete a *Delivery Service* cannot be created if an open *Delivery Service Request* exists for a *Delivery Service* with the same *xml\_id*. Because of this, *Delivery Service Requests* cannot be used to change a *Delivery Service*’s *xml\_id*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:CREATE, DELIVERY-SERVICE:READ,  
USER:READ

**Response Type** Object

## Request Structure

The request must be a well-formed representation of a *Delivery Service Request*, without any response-only fields, of course.

### 725: Request Example

```
POST /api/4.0/deliveryservice_requests HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 1979

{
  "changeType": "update",
  "status": "draft",
  "requested": {
    "active": false,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "displayName": "Demo 1",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "firstHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
    "globalMaxTps": null,
    "httpBypassFqdn": null,
    "id": 1,
    "infoUrl": null,
    "initialDispersion": 1,
    "innerHeaderRewrite": null,
    "ipv6RoutingEnabled": true,
    "lastHeaderRewrite": null,
```

(continues on next page)

(continued from previous page)

```

"lastUpdated": "2020-02-13T16:43:54Z",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
  "http://video.demo1.mycdn.ciab.test",
  "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"topology": null,
"trResponseHeaders": null,
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [

```

(continues on next page)



(continued from previous page)

```

        "abc",
        "pdq",
        "xxx",
        "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "serviceCategory": null,
    "tlsVersions": null
}

```

## Response Structure

The response will be a representation of the created *Delivery Service Request*.

### 726: Response Example

```

HTTP/1.1 201 CREATED
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:11:12.
    ↳GMT; Max-Age=3600; HttpOnly
Location: /api/4.0/deliveryservice_requests/2
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:11:12 GMT
Content-Length: 901

{
  "alerts": [
    {
      "text": "deliveryservice_request was created.",
      "level": "success"
    }
  ],
  "response": {
    "authorId": 2,
    "author": null,
    "changeType": "update",
    "createdAt": null,

```

(continues on next page)

(continued from previous page)

```

    "id": 2,
    "lastEditedBy": null,
    "lastEditedById": 2,
    "lastUpdated": "2020-02-24 19:11:12+00",
    "requested": {
      "active": false,
      "anonymousBlockingEnabled": false,
      "cacheurl": null,
      "ccrDnsTtl": null,
      "cdnId": 2,
      "cdnName": "CDN-in-a-Box",
      "checkPath": null,
      "displayName": "Demo 1",
      "dnsBypassCname": null,
      "dnsBypassIp": null,
      "dnsBypassIp6": null,
      "dnsBypassTtl": null,
      "dscp": 0,
      "edgeHeaderRewrite": null,
      "firstHeaderRewrite": null,
      "geoLimit": 0,
      "geoLimitCountries": null,
      "geoLimitRedirectURL": null,
      "geoProvider": 0,
      "globalMaxMbps": null,
      "globalMaxTps": null,
      "httpBypassFqdn": null,
      "id": 1,
      "infoUrl": null,
      "initialDispersion": 1,
      "innerHeaderRewrite": null,
      "ipv6RoutingEnabled": true,
      "lastHeaderRewrite": null,
      "lastUpdated": "0001-01-01T00:00:00Z",
      "logsEnabled": true,
      "longDesc": "Apachecon North America 2018",
      "matchList": [
        {
          "type": "HOST_REGEX",
          "setNumber": 0,
          "pattern": ".*\\.demo1\\.\\.\\.*"
        }
      ],
      "maxDnsAnswers": null,
      "midHeaderRewrite": null,
      "missLat": 42,

```

(continues on next page)

(continued from previous page)

```

    "missLong": -88,
    "multiSiteOrigin": false,
    "originShield": null,
    "orgServerFqdn": "http://origin.infra.ciab.test",
    "profileDescription": null,
    "profileId": null,
    "profileName": null,
    "protocol": 2,
    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "video",
    "signed": false,
    "sslKeyVersion": 1,
    "tenantId": 1,
    "topology": null,
    "type": "HTTP",
    "typeId": 1,
    "xmlId": "demo1",
    "exampleURLs": [
        "http://video.demo1.mycdn.ciab.test",
        "https://video.demo1.mycdn.ciab.test"
    ],
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": "root",
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": [
        "abc",
        "pdq",
        "xxx",
        "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "tlsVersions": null
},
"original": {
    "active": true,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,

```

(continues on next page)

(continued from previous page)

```

"ccrDnsTtl": null,
"cdnId": 2,
"cdnName": "CDN-in-a-Box",
"checkPath": null,
"displayName": "Demo 1",
"dnsBypassCname": null,
"dnsBypassIp": null,
"dnsBypassIp6": null,
"dnsBypassTtl": null,
"dscp": 0,
"edgeHeaderRewrite": null,
"firstHeaderRewrite": null,
"geoLimit": 0,
"geoLimitCountries": null,
"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "2020-02-13T16:43:54Z",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.\\.*"
    }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,

```

(continues on next page)

(continued from previous page)

```

    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "video",
    "signed": false,
    "sslKeyVersion": 1,
    "tenantId": 1,
    "type": "HTTP",
    "typeId": 1,
    "xmlId": "demo1",
    "exampleURLs": [
        "http://video.demo1.mycdn.ciab.test",
        "https://video.demo1.mycdn.ciab.test"
    ],
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": "root",
    "topology": null,
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": [
        "abc",
        "pdq",
        "xxx",
        "zyx"
    ],
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "serviceCategory": null,
    "tlsVersions": null
  },
  "status": "draft"
}

```

## PUT

Updates an existing *Delivery Service Request*. Note that “closed” *Delivery Service Requests* are uneditable.

### See also:

The proper way to change a *Delivery Service Request*’s *Status* is by using the *deliveryservice\_requests/{{ID}}/status* endpoint’s PUT handler.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:UPDATE, DELIVERY-SERVICE:READ, USER:READ

**Response Type** Object

## Request Structure

Table 301: Request Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> that you want to update.

The request body must be a representation of a *Delivery Service Request* without any response-only fields.

### 727: Request Example

```
PUT /api/4.0/deliveryservice_requests?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 2256

{
  "changeType": "update",
  "requested": {
    "active": true,
    "cdnId": 2,
    "ccrDnsTtl": 30,
    "deepCachingType": "NEVER",
    "displayName": "Demo 1 but I modified the DSR",
```

(continues on next page)

(continued from previous page)

```

        "dscp": 0,
        "geoLimit": 0,
        "geoProvider": 0,
        "initialDispersion": 3,
        "logsEnabled": false,
        "longDesc": "long desc",
        "regionalGeoBlocking": false,
        "tenantId": 1,
        "typeId": 8,
        "xmlId": "demo1",
        "id": 1
    },
    "status": "draft"
}

```

## Response Structure

The response is a full representation of the edited *Delivery Service Request*.

### 728: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:36:16.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +W0vFm96yFkZUJqa0GAX7uzIpRKh/
→ohyBm0uH3egpiERTcxy50fVVtoP3h8Ee2teLu8KFooDYXJ6rpQg6UhbNQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:36:16 GMT
Content-Length: 913

{ "alerts": [{
    "text": "Delivery Service Request #2 updated",
    "level": "success"
}],
  "response": {
    "assignee": null,
    "author": "",
    "changeType": "update",

```

(continues on next page)

(continued from previous page)

```

"createdAt": "2020-09-25T06:23:30.683058Z",
"id": null,
"lastEditedBy": "admin",
"lastUpdated": "2020-09-25T02:38:04.180237Z",
"original": {
  "active": true,
  "anonymousBlockingEnabled": false,
  "cacheurl": null,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "checkPath": null,
  "displayName": "Demo 1",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "edgeHeaderRewrite": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectURL": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "id": 1,
  "infoUrl": null,
  "initialDispersion": 1,
  "ipv6RoutingEnabled": true,
  "lastUpdated": "2020-09-25T02:09:54Z",
  "logsEnabled": true,
  "longDesc": "Apachecon North America 2018",
  "matchList": [
    {
      "type": "HOST_REGEX",
      "setNumber": 0,
      "pattern": ".*\\.demo1\\..*"
    }
  ],
  "maxDnsAnswers": null,
  "midHeaderRewrite": null,
  "missLat": 42,
  "missLong": -88,
  "multiSiteOrigin": false,
  "originShield": null,

```

(continues on next page)



(continued from previous page)

```

"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
    "http://video.demo1.mycdn.ciab.test",
    "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
"tenant": "root",
"trResponseHeaders": null,
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [
    "abc",
    "pdq",
    "xxx",
    "zyx"
],
"maxOriginConnections": 0,
"ecsEnabled": false,
"rangeSliceBlockSize": null,
"topology": "demo1-top",
"firstHeaderRewrite": null,
"innerHeaderRewrite": null,
"lastHeaderRewrite": null,
"serviceCategory": null,
"tlsVersions": null
},
"requested": {
    "active": true,

```

(continues on next page)

(continued from previous page)

```
"anonymousBlockingEnabled": false,
"cacheurl": null,
"ccrDnsTtl": 30,
"cdnId": 2,
"cdnName": null,
"checkPath": null,
"displayName": "Demo 1 but I modified the DSR",
"dnsBypassCname": null,
"dnsBypassIp": null,
"dnsBypassIp6": null,
"dnsBypassTtl": null,
"dscp": 0,
"edgeHeaderRewrite": null,
"geoLimit": 0,
"geoLimitCountries": null,
"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 3,
"ipv6RoutingEnabled": null,
"lastUpdated": null,
"logsEnabled": false,
"longDesc": "long desc",
"matchList": null,
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": null,
"missLong": null,
"multiSiteOrigin": null,
"originShield": null,
"orgServerFqdn": null,
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": null,
"qstringIgnore": null,
"rangeRequestHandling": null,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "cdn",
"signed": false,
```

(continues on next page)

(continued from previous page)

```

    "sslKeyVersion": null,
    "tenantId": 1,
    "type": null,
    "typeId": 8,
    "xmlId": "demo1",
    "exampleURLs": null,
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": null,
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": null,
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "rangeSliceBlockSize": null,
    "topology": null,
    "firstHeaderRewrite": null,
    "innerHeaderRewrite": null,
    "lastHeaderRewrite": null,
    "serviceCategory": null,
    "tlsVersions": null
  },
  "status": "draft"
}
```

## DELETE

Deletes a *Delivery Service Request*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:DELETE, DELIVERY-SERVICE:READ,  
USER:READ

**Response Type** Object

## Request Structure

Table 302: Request Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Delivery Service Request</i> that you want to delete.

729: Request Example

```
DELETE /api/4.0/deliveryservice_requests?id=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

The response is a full representation of the deleted *Delivery Service Request*.

730: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 20:48:55.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→jNCbNo8Tw+JMMaWpAYQgntSXPq2XuJ+n2zSEVRaDQFWMV1SYbT9djes6SPdwiBoKq6W0lNE04h0E92jBV
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 19:48:55 GMT
Content-Length: 96

{
  "alerts": [
    {
      "text": "deliveryservice_request was deleted.",
      "level": "success"
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
    }  
  ]  
}
```

**deliveryservice\_requests/{ID}/assign**

Assign a *Delivery Service Request* to a user.

**GET**

New in version 4.0.

- Auth. Required** Yes
- Roles Required** “admin” or “operations”
- Permissions Required** DS-REQUEST:READ, USER:READ
- Response Type** Object (string)

**Request Structure**

Table 303: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service Request</i> for which assignment is being retrieved

### 731: Request Example

```
GET /api/4.0/deliveryservice_requests/1/assign HTTP/1.1
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

The response is the username of the user to whom the *Delivery Service Request* is assigned - or null if it is unassigned.

### 732: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 02 Feb 2021 22:48:48.
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 02 Feb 2021 21:48:48 GMT
Content-Length: 45

{ "response": "admin" }
```

## PUT

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-REQUEST:UPDATE, DS-REQUEST:READ,  
USER:READ

**Response Type** Object

## Request Structure

Table 304: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service Request</i> being assigned

**assignee** The username of the user to whom the *Delivery Service Request* is assigned

New in version 4.0.

**assigneeId** The integral, unique identifier of the user to whom the *Delivery Service Request* is assigned

Changed in version 4.0: Prior to APIv4.0, this was the only property that could be used to change a *Delivery Service Request*'s Assignee - and thus was a required field.

It is not required to send both of these; either property is sufficient to determine an *Assignee*. In most cases, it's easier to use just *assignee*. If both *are* given, then *assigneeId* will take precedence in the event that the two properties do not refer to the same user. Sending a request that sets the assignee to null un-assigns the *DSR* from any assignees it previously had<sup>1</sup>.

### 733: Request Example

```
PUT /api/4.0/deliveryservice_requests/1/assign HTTP/1.1
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 20

{"assignee": "admin"}
```

## Response Structure

The response contains a full representation of the newly assigned *Delivery Service Request*.

### 734: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie (continues on next page)
```

<sup>1</sup> Because of how the Traffic Ops API parses requests, there is no distinction between null and undefined/missing properties. This means that sending the request payload {} in this case will result in the *DSR* being unassigned.

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 14:45:51
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→h7uBZHLQtRYbOSOR5AtQQRZ4uMeEWivWNT74fCf6WtLbAMwGpRrMjNmBYKduv48DnRqG6WVM/
→4nBu3AkCUqPw==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 13:45:51 GMT
Content-Length: 931

{ "alerts": [{
  "text": "Changed assignee of 'demo1' Delivery Service Request to
→'admin'",
  "level": "success"
}],
"response": {
  "assignee": "admin",
  "author": "admin",
  "changeType": "update",
  "createdAt": "2020-09-25T06:52:23.758877Z",
  "id": 6,
  "lastEditedBy": "admin",
  "lastUpdated": "2020-09-25T07:01:24.600029Z",
  "original": {
    "active": true,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "displayName": "Demo 1",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
```

(continues on next page)



(continued from previous page)

```
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"ipv6RoutingEnabled": true,
"lastUpdated": "2020-09-25T02:09:54Z",
"logsEnabled": true,
"longDesc": "Apachecon North America 2018",
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"signed": false,
"sslKeyVersion": 1,
"tenantId": 1,
"type": "HTTP",
"typeId": 1,
"xmlId": "demo1",
"exampleURLs": [
  "http://video.demo1.mycdn.ciab.test",
  "https://video.demo1.mycdn.ciab.test"
],
"deepCachingType": "NEVER",
"fqPacingRate": null,
"signingAlgorithm": null,
```

(continues on next page)

(continued from previous page)

```

"tenant": "root",
"trResponseHeaders": null,
"trRequestHeaders": null,
"consistentHashRegex": null,
"consistentHashQueryParams": [
    "abc",
    "pdq",
    "xxx",
    "zyx"
],
"maxOriginConnections": 0,
"ecsEnabled": false,
"rangeSliceBlockSize": null,
"topology": "demo1-top",
"firstHeaderRewrite": null,
"innerHeaderRewrite": null,
"lastHeaderRewrite": null,
"serviceCategory": null,
"tlsVersions": null
},
"requested": {
    "active": true,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": 30,
    "cdnId": 2,
    "cdnName": null,
    "checkPath": null,
    "displayName": "Demo 1 but modified by a DSR",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
    "globalMaxTps": null,
    "httpBypassFqdn": null,
    "id": 1,
    "infoUrl": null,
    "initialDispersion": 3,
    "ipv6RoutingEnabled": null,

```

(continues on next page)

(continued from previous page)

```

    "lastUpdated": null,
    "logsEnabled": false,
    "longDesc": "long desc",
    "matchList": null,
    "maxDnsAnswers": null,
    "midHeaderRewrite": null,
    "missLat": null,
    "missLong": null,
    "multiSiteOrigin": null,
    "originShield": null,
    "orgServerFqdn": null,
    "profileDescription": null,
    "profileId": null,
    "profileName": null,
    "protocol": null,
    "qstringIgnore": null,
    "rangeRequestHandling": null,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "cdn",
    "signed": false,
    "sslKeyVersion": null,
    "tenantId": 1,
    "type": null,
    "typeId": 8,
    "xmlId": "demo1",
    "exampleURLs": null,
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": null,
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": null,
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "rangeSliceBlockSize": null,
    "topology": null,
    "firstHeaderRewrite": null,
    "innerHeaderRewrite": null,
    "lastHeaderRewrite": null,
    "serviceCategory": null,
    "tlsVersions": null
  },

```

(continues on next page)

(continued from previous page)

```
"status": "draft"
}}
```

## deliveryservice\_requests/{ID}/status

Get or set the status of a *Delivery Service Request*.

### GET

Gets the status of a *DSR*.

New in version 4.0.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:READ

**Response Type** Object (string)

## Request Structure

Table 305: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service Request</i> being inspected

### 735: Request Example

```
GET /api/4.0/deliveryservice_requests/1/status HTTP/1.1
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

The response is the status of the requested *DSR*.

### 736: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 02 Feb 2021 22:56:47
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 02 Feb 2021 21:56:47 GMT
Content-Length: 45

{ "response": "draft" }
```

## PUT

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** DS-REQUEST:UPDATE, DS-REQUEST:READ

**Response Type** Object

## Request Structure

Table 306: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service Request</i> being modified

**status** The status of the *DSR*. Can be “draft”, “submitted”, “rejected”, “pending”, or “complete”.

737: Request Example

```
PUT /api/4.0/deliveryservice_requests/1/status HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 28

{
  "status": "rejected"
}
```

## Response Structure

The response is a full representation of the modified *DSR*.

738: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 15:54:53
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳C8Nhciy1jv5X7CGgHwAnLp1qmLIzHq+4dv1AApb3cFSz5V2dABl7+N1Z4ndzB7GertB7rNLP31pVcat8v
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 14:54:53 GMT
Content-Length: 930

{ "alerts": [{
```

(continues on next page)

(continued from previous page)

```

    "text": "Changed status of 'demo1' Delivery Service Request from
    ↪ 'draft' to 'submitted'",
    "level": "success"
  }],
  "response": {
    "assignee": "admin",
    "author": "admin",
    "changeType": "update",
    "createdAt": "2020-09-25T06:52:23.758877Z",
    "id": 6,
    "lastEditedBy": "admin",
    "lastUpdated": "2020-09-25T07:13:28.753352Z",
    "original": {
      "active": true,
      "anonymousBlockingEnabled": false,
      "cacheurl": null,
      "ccrDnsTtl": null,
      "cdnId": 2,
      "cdnName": "CDN-in-a-Box",
      "checkPath": null,
      "displayName": "Demo 1",
      "dnsBypassCname": null,
      "dnsBypassIp": null,
      "dnsBypassIp6": null,
      "dnsBypassTtl": null,
      "dscp": 0,
      "edgeHeaderRewrite": null,
      "geoLimit": 0,
      "geoLimitCountries": null,
      "geoLimitRedirectURL": null,
      "geoProvider": 0,
      "globalMaxMbps": null,
      "globalMaxTps": null,
      "httpBypassFqdn": null,
      "id": 1,
      "infoUrl": null,
      "initialDispersion": 1,
      "ipv6RoutingEnabled": true,
      "lastUpdated": "2020-09-25T02:09:54Z",
      "logsEnabled": true,
      "longDesc": "Apachecon North America 2018",
      "matchList": [
        {
          "type": "HOST_REGEX",
          "setNumber": 0,
          "pattern": ".*\\.demo1\\.\\.\\.*"
        }
      ]
    }
  }
}

```

(continues on next page)

(continued from previous page)

```

    }
  ],
  "maxDnsAnswers": null,
  "midHeaderRewrite": null,
  "missLat": 42,
  "missLong": -88,
  "multiSiteOrigin": false,
  "originShield": null,
  "orgServerFqdn": "http://origin.infra.ciab.test",
  "profileDescription": null,
  "profileId": null,
  "profileName": null,
  "protocol": 2,
  "qstringIgnore": 0,
  "rangeRequestHandling": 0,
  "regexRemap": null,
  "regionalGeoBlocking": false,
  "remapText": null,
  "routingName": "video",
  "signed": false,
  "sslKeyVersion": 1,
  "tenantId": 1,
  "type": "HTTP",
  "typeId": 1,
  "xmlId": "demo1",
  "exampleURLs": [
    "http://video.demo1.mycdn.ciab.test",
    "https://video.demo1.mycdn.ciab.test"
  ],
  "deepCachingType": "NEVER",
  "fqPacingRate": null,
  "signingAlgorithm": null,
  "tenant": "root",
  "trResponseHeaders": null,
  "trRequestHeaders": null,
  "consistentHashRegex": null,
  "consistentHashQueryParams": [
    "abc",
    "pdq",
    "xxx",
    "zyx"
  ],
  "maxOriginConnections": 0,
  "ecsEnabled": false,
  "rangeSliceBlockSize": null,
  "topology": "demo1-top",

```

(continues on next page)



(continued from previous page)

```

    "firstHeaderRewrite": null,
    "innerHeaderRewrite": null,
    "lastHeaderRewrite": null,
    "serviceCategory": null,
    "tlsVersions": null
  },
  "requested": {
    "active": true,
    "anonymousBlockingEnabled": false,
    "cacheurl": null,
    "ccrDnsTtl": 30,
    "cdnId": 2,
    "cdnName": null,
    "checkPath": null,
    "displayName": "Demo 1 but modified by a DSR",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "edgeHeaderRewrite": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
    "globalMaxMbps": null,
    "globalMaxTps": null,
    "httpBypassFqdn": null,
    "id": 1,
    "infoUrl": null,
    "initialDispersion": 3,
    "ipv6RoutingEnabled": null,
    "lastUpdated": null,
    "logsEnabled": false,
    "longDesc": "long desc",
    "matchList": null,
    "maxDnsAnswers": null,
    "midHeaderRewrite": null,
    "missLat": null,
    "missLong": null,
    "multiSiteOrigin": null,
    "originShield": null,
    "orgServerFqdn": null,
    "profileDescription": null,
    "profileId": null,
    "profileName": null,

```

(continues on next page)

(continued from previous page)

```

    "protocol": null,
    "qstringIgnore": null,
    "rangeRequestHandling": null,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "cdn",
    "signed": false,
    "sslKeyVersion": null,
    "tenantId": 1,
    "type": null,
    "typeId": 8,
    "xmlId": "demo1",
    "exampleURLs": null,
    "deepCachingType": "NEVER",
    "fqPacingRate": null,
    "signingAlgorithm": null,
    "tenant": null,
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "consistentHashRegex": null,
    "consistentHashQueryParams": null,
    "maxOriginConnections": 0,
    "ecsEnabled": false,
    "rangeSliceBlockSize": null,
    "topology": null,
    "firstHeaderRewrite": null,
    "innerHeaderRewrite": null,
    "lastHeaderRewrite": null,
    "serviceCategory": null,
    "tlsVersions": null
  },
  "status": "submitted"
}}

```

## deliveryservice\_stats

### GET

Retrieves time-aggregated statistics on a specific *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

<sup>1</sup> This endpoint respects *Tenancy*, and users whose *Tenant* does not have access to a *Delivery Service* will be unable to view the statistics of said *Delivery Service*.

**Permissions Required** STAT:READ, DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

Table 307: Request Query Parameters

Name	Required	Description
deliveryService	yes <sup>2</sup>	Either the <i>xml_id</i> of a <i>Delivery Service</i> for which statistics will be aggregated or the integral, unique identifier of said <i>Delivery Service</i>
deliveryServiceName	yes <sup>?</sup>	The <i>xml_id</i> of the <i>Delivery Service</i> for which statistics will be aggregated
endDate	yes	The date and time until which statistics shall be aggregated in <b>RFC 3339</b> format (with or without sub-second precision), the number of nanoseconds since the Unix Epoch, or in the same, proprietary format as the <code>lastUpdated</code> fields prevalent throughout the Traffic Ops API
exclude	no	Either “series” to omit the data series from the result, or “summary” to omit the summary data from the result - directly corresponds to fields in the <i>Response Structure</i>
interval	no	Specifies the interval within which data will be “bucketed”; e.g. when requesting data from 2019-07-25T00:00:00Z to 2019-07-25T23:59:59Z with an interval of “1m”, the resulting data series (assuming it is not excluded) should contain $24 \frac{\text{hours}}{\text{day}} \times 60 \frac{\text{minutes}}{\text{hour}} \times 1 \text{day} \times 1 \frac{\text{minute}}{\text{datapoint}} = 1440 \text{data points}$ The allowed values for this parameter are valid InfluxQL duration literal strings matching <code>^d+[mhdw]\$</code>
limit	no	A natural number indicating the maximum amount of data points should be returned in the series
metricType	yes	The metric type being reported - one of: <b>Chapter 5. APIs</b> <code>kbps</code> The total traffic rate

### 739: Request Example

```
GET /api/4.0/deliveryservice_stats?deliveryServiceName=demo1&
→startDate=2019-07-22T17:55:00Z&endDate=2019-07-22T17:56:00.000Z&
→metricType=tps_total HTTP/1.1
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: application/json;timestamp=unix, application/json;
→timestamp=rfc;q=0.9, application/json;q=0.8, */*;q=0.7
Connection: keep-alive
Cookie: mojolicious=...
```

## Content Format

It's important to note in *Request Example* the use of a complex “Accept” header. This endpoint accepts two special media types in the “Accept” header that instruct it on how to format the timestamps associated with the returned data. Specifically, Traffic Ops will recognize the special, optional, non-standard parameter of *application/json: timestamp*. The values of this parameter are restricted to one of

**rfc** Returned timestamps will be formatted according to **RFC 3339** (no sub-second precision).

**unix** Returned timestamps will be formatted as the number of nanoseconds since the Unix Epoch (midnight on January 1<sup>st</sup> 1970 UTC).

### Implementation Detail

The endpoint passes back nanoseconds, specifically, because that is the form used both by InfluxDB, which is used to store the data being served, and Go's standard library. Clients may need to convert the value to match their own standard libraries - e.g. the `Date()` class in Javascript expects milliseconds.

The default behavior - when only e.g. *application/json* or *\*/\** is given - is to use **RFC 3339** formatting. It will, however, respect quality parameters. It is suggested that clients request timestamps they can handle specifically, rather than relying on this default behavior, as it **is subject to change** and is in fact **expected to invert in the next major release** as string-based time formats become deprecated.

### See also:

For more information on the “Accept” HTTP header, consult [its dedicated page on MDN](#).

<sup>2</sup> Either `deliveryServiceName` or `deliveryService` *must* be present, but if both are `deliveryServiceName` will be used and `deliveryService` will be ignored.

## Response Structure

**series** An object containing the actual data series and information necessary for working with it.

**columns** This is an array of names of the columns of the data contained in the “values” array - should always be `["time", "sum_count"]`

**count** The number of data points contained in the “values” array

**name** The name of the data set. Should always match `metric.ds.1min` where `metric` is the requested `metricType`

**values** The actual array of data points. Each represents a length of time specified by the `interval` query parameter

**time** The time at which the measurement was taken. This corresponds to the *beginning* of the interval. This time comes in the format of either an [RFC 3339](#)-formatted string, or a number containing the number of nanoseconds since the Unix Epoch depending on the “Accept” header sent by the client, according to the rules outlined in [Content Format](#).

**value** The value of the requested `metricType` at the time given by `time`. This will always be a floating point number, unless no data is available for the data interval, in which case it will be `null`

**source** A legacy field meant only for plugins that override this endpoint to name themselves. Should always be “TrafficStats”.

Deprecated since version 1.4: As this has no known purpose, developers are advised it will be removed in the future.

**summary** An object containing summary statistics describing the data series

**average** The arithmetic mean of the data’s values

**count** The number of measurements taken within the requested timespan. This is, in general, **not** the same as the `count` field of the `series` object, as it reflects the number of underlying, un-“bucketed” data points, and is therefore dependent on the implementation of Traffic Stats.

**fifthPercentile** Data points with values less than or equal to this number constitute the “bottom” 5% of the data set

**max** The maximum value that can be found in the requested data set

**min** The minimum value that can be found in the requested data set

**ninetyEighthPercentile** Data points with values greater than or equal to this number constitute the “top” 2% of the data set

**ninetyFifthPercentile** Data points with values greater than or equal to this number constitute the “top” 5% of the data set

**totalBytes** When the `metricType` requested is `kbps`, this will contain the total number of bytes transferred by the *Delivery Service* within the requested time window. Note that fractional amounts are possible, as the data transfer rate will almost certainly not be cleanly divided by the requested time range.

**totalTransactions** When the `metricType` requested is `not kbps`, this will contain the total number of transactions completed by the *Delivery Service* within the requested time window. Note that fractional amounts are possible, as the transaction rate will almost certainly not be cleanly divided by the requested time range.

**version** A legacy field that seems to have been meant to indicate the API version used. Will always be “1.2”

Deprecated since version 1.4: As this has no known purpose, developers are advised it will be removed in the future.

#### 740: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: zXJGjcYuu6HxWINVp8HA1gL31J3ukry5wCsTDNxtP/
    ↳waC6rSD8h10KJ9jEAtrZJ9owOSVPvKaA/2bRu/QeuCpQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 22 Jul 2019 17:57:14 GMT
Transfer-Encoding: chunked
```

(continues on next page)

(continued from previous page)

```
{ "response": {
  "series": {
    "columns": [
      "time",
      "sum_count"
    ],
    "count": 2,
    "name": "tps_total.ds.1min",
    "tags": {
      "cachegroup": "total"
    },
    "values": [
      [
        15638181000000000000,
        0
      ],
      [
        15638181600000000000,
        0
      ]
    ]
  },
  "source": "TrafficStats",
  "summary": {
    "average": 0,
    "count": 2,
    "fifthPercentile": 0,
    "max": 0,
    "min": 0,
    "ninetyEighthPercentile": 0,
    "ninetyFifthPercentile": 0,
    "totalBytes": null,
    "totalTransactions": 0
  },
  "version": "1.2"
}}
```



## deliveryservices

### GET

Retrieves *Delivery Services*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ, CDN:READ,  
TYPE:READ

**Response Type** Array

---

<sup>1</sup> Only those *Delivery Services* assigned to *Tenants* that are the requesting user's *Tenant* or children thereof will appear in the output of a GET request, and the same constraints are placed on the allowed values of the `tenantId` field of a POST request to create a new *Delivery Service*

## Request Structure

Table 308: Request Query Parameters

Name	Re-quired	Description
cdn	no	Show only the <i>Delivery Services</i> belonging to the <i>CDN</i> identified by this integral, unique identifier
id	no	Show only the <i>Delivery Service</i> that has this integral, unique identifier
logsEnabled	no	Show only the <i>Delivery Services</i> that have <i>Logs Enabled</i> set or not based on this boolean
profile	no	Return only <i>Delivery Services</i> using the <i>Profile</i> that has this <i>ID</i>
tenant	no	Show only the <i>Delivery Services</i> belonging to the <i>Tenant</i> identified by this integral, unique identifier
topology	no	Show only the <i>Delivery Services</i> assigned to the <i>Topology</i> identified by this unique name
type	no	Return only <i>Delivery Services</i> of the <i>Delivery Service Type</i> identified by this integral, unique identifier
accessibleTo	no	Return the <i>Delivery Services</i> accessible from a <i>Tenant</i> or it's children identified by this integral, unique identifier
service-Category	no	Show only the <i>Delivery Services</i> belonging to the <i>Service Category</i> that has this name
xmlId	no	Show only the <i>Delivery Service</i> that has this text-based, unique identifier
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n<sup>th</sup></i> page of results, where “n” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .
active	no	Show only the <i>Delivery Services</i> that have <i>Active</i> set or not based on this boolean (whether or not they are active)

### 741: Request Example

```
GET /api/4.0/deliveryservices?xmlId=demo2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.24.0
```

(continues on next page)

(continued from previous page)

```
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- active** A boolean that defines *Active*.
- anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*
- ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons
- cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs
- cdnName** Name of the *CDN* to which the *Delivery Service* belongs
- checkPath** A *Check Path*
- consistentHashRegex** A *Consistent Hashing Regular Expression*
- consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*
- deepCachingType** The *Deep Caching* setting for this *Delivery Service*
- displayName** The *Display Name*
- dnsBypassCname** A *DNS Bypass CNAME*
- dnsBypassIp** A *DNS Bypass IP*
- dnsBypassIp6** A *DNS Bypass IPv6*
- dnsBypassTtl** The *DNS Bypass TTL*
- dscp** A *DSCP* to be used within the *Delivery Service*
- ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*
- edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*
- exampleURLs** An array of *Example URLs*
- firstHeaderRewrite** A set of *First Header Rewrite Rules*
- fqPacingRate** The *Fair-Queuing Pacing Rate Bps*
- geoLimit** An integer that defines the *Geo Limit*
- geoLimitCountries** An array of strings defining the *Geo Limit Countries*
- geoLimitRedirectUrl** A *Geo Limit Redirect URL*
- geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in :rfc:3339 format

Changed in version 4.0: Prior to API version 4.0, this property used *Traffic Ops's Custom Date/Time Format*.

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
- setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
- type** The type of match performed using pattern.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

- profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated
- profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated
- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- tlsVersions** A list of explicitly supported *TLS Versions*  
New in version 4.0.
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

742: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 07 Jun 2021 22:52:20
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Jun 2021 21:52:20 GMT
Content-Length: 847

{ "response": [
  {
    "active": true,
    "anonymousBlockingEnabled": false,
    "ccrDnsTtl": null,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "checkPath": null,
    "consistentHashQueryParams": [],
    "consistentHashRegex": null,
    "deepCachingType": "NEVER",
    "displayName": "Demo 2",
    "dnsBypassCname": null,
    "dnsBypassIp": null,
    "dnsBypassIp6": null,
    "dnsBypassTtl": null,
    "dscp": 0,
    "ecsEnabled": false,
    "edgeHeaderRewrite": null,
    "exampleURLs": [
      "http://video.demo2.mycdn.ciab.test",
      "https://video.demo2.mycdn.ciab.test"
    ],
    "firstHeaderRewrite": null,
    "fqPacingRate": null,
    "geoLimit": 0,
    "geoLimitCountries": null,
    "geoLimitRedirectURL": null,
    "geoProvider": 0,
  }
]

```

(continues on next page)

(continued from previous page)

```
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": null,
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "2021-06-07T21:50:03.009954Z",
"logsEnabled": true,
"longDesc": "DNS Delivery Service for use with a Federation",
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo2\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"maxOriginConnections": 0,
"maxRequestHeaderBytes": 0,
"midHeaderRewrite": null,
"missLat": 42,
"missLong": -88,
"multiSiteOrigin": true,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 2,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"rangeSliceBlockSize": null,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "video",
"serviceCategory": null,
"signed": false,
"signingAlgorithm": null,
"sslKeyVersion": null,
"tenant": "root",
"tenantId": 1,
"tlsVersions": null,
```

(continues on next page)

(continued from previous page)

```

        "topology": "demo1-top",
        "trResponseHeaders": null,
        "trRequestHeaders": null,
        "type": "DNS",
        "typeId": 5,
        "xmlId": "demo2"
    }
}]

```

## POST

Allows users to create *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 1040, 1</sup>

**Permissions Required** DELIVERY-SERVICE:CREATE, DELIVERY-SERVICE:READ, CDN:READ, TYPE:READ

**Response Type** Array

## Request Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*



**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list, or an array of strings defining the *Geo Limit Countries*<sup>2</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>?</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

---

<sup>2</sup> These fields must be defined if and only if `geoLimit` is non-zero

- profileId** An optional *ID* of a *Profile* with which this *Delivery Service* shall be associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*
- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated - or null if there is to be no such category
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3. It can only be between (inclusive) 262144 (256KB) - 33554432 (32MB).
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- tlsVersions** An array of explicitly supported *TLS Versions*  
New in version 4.0.
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

743: Request Example

```
POST /api/4.0/deliveryservices HTTP/1.1
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 1602
Content-Type: application/json
Host: trafficops.infra.ciab.test

{
  "active": false,
  "anonymousBlockingEnabled": false,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "checkPath": null,
  "consistentHashRegex": null,
  "consistentHashQueryParams": [],
  "deepCachingType": "NEVER",
  "displayName": "test",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "ecsEnabled": true,
  "edgeHeaderRewrite": null,
  "firstHeaderRewrite": null,
  "fqPacingRate": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectUrl": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": false,
  "lastHeaderRewrite": null,
  "logsEnabled": true,
  "longDesc": "A Delivery Service created expressly for API_
→documentation examples",
  "maxDnsAnswers": null,
```

(continues on next page)

(continued from previous page)

```

"missLat": 0,
"missLong": 0,
"maxOriginConnections": 0,
"maxRequestHeaderBytes": 131072,
"midHeaderRewrite": null,
"multiSiteOrigin": false,
"orgServerFqdn": "http://origin.infra.ciab.test",
"originShield": null,
"profileId": null,
"protocol": 0,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"regexRemap": null,
"regionalGeoBlocking": false,
"routingName": "test",
"serviceCategory": null,
"signed": false,
"signingAlgorithm": null,
"rangeSliceBlockSize": null,
"sslKeyVersion": null,
"tenant": "root",
"tenantId": 1,
"tlsVersions": [
    "1.2",
    "1.3"
],
"topology": null,
"trRequestHeaders": null,
"trResponseHeaders": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "test"
}

```

## Response Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** An array of strings defining the *Geo Limit Countries*

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in :rfc:3339 format

Changed in version 4.0: Prior to API version 4.0, this property used *Traffic Ops's Custom Date/Time Format*.

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**matchList** The *Delivery Service*'s *Match List*

- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
- setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
- type** The type of match performed using pattern.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated
- signed** true if and only if `signingAlgorithm` is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if `rangeRequestHandling` is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- tlsVersions** An array of explicitly supported *TLS Versions*  
New in version 4.0.
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

#### 744: Response Example

```
HTTP/1.1 201 Created
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Location: /api/4.0/deliveryservices?id=6
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 07 Jun 2021 23:37:37
➔GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Jun 2021 22:37:37 GMT
Content-Length: 903
```

(continues on next page)

(continued from previous page)

```

{ "alerts": [
  {
    "text": "tlsVersions has no effect on 'HTTP' Delivery Services
→",
    "level": "warning"
  },
  {
    "text": "Delivery Service creation was successful",
    "level": "success"
  }
],
"response": [{
  "active": false,
  "anonymousBlockingEnabled": false,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "cdnName": null,
  "checkPath": null,
  "consistentHashQueryParams": [],
  "consistentHashRegex": null,
  "deepCachingType": "NEVER",
  "displayName": "test",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "ecsEnabled": true,
  "edgeHeaderRewrite": null,
  "exampleURLs": [
    "http://test.test.mycdn.ciab.test"
  ],
  "firstHeaderRewrite": null,
  "fqPacingRate": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectURL": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "id": 6,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": false,

```

(continues on next page)



(continued from previous page)

```

"lastHeaderRewrite": null,
"lastUpdated": "2021-06-07T22:37:37.187822Z",
"logsEnabled": true,
"longDesc": "A Delivery Service created expressly for API_
→documentation examples",
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.test\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"maxOriginConnections": 0,
"maxRequestHeaderBytes": 131072,
"midHeaderRewrite": null,
"missLat": 0,
"missLong": 0,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 0,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"rangeSliceBlockSize": null,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "test",
"serviceCategory": null,
"signed": false,
"signingAlgorithm": null,
"sslKeyVersion": null,
"tenant": "root",
"tenantId": 1,
"tlsVersions": [
  "1.2",
  "1.3"
],
"topology": null,
"trResponseHeaders": null,
"trRequestHeaders": null,
"type": "HTTP",

```

(continues on next page)

(continued from previous page)

```
"typeId": 1,
"xmlId": "test"
}}}
```

**deliveryservices/{{ID}}**

**PUT**

Allows users to edit an existing *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:UPDATE, DELIVERY-SERVICE:READ, CDN:READ, TYPE:READ

**Response Type** Array (should always have a length of exactly one on success)

## Request Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

---

**Note:** If the Delivery Service has SSL Keys, then cdnId is not allowed to change as that would invalidate the SSL Key

---

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

---

<sup>1</sup> Only those *Delivery Services* assigned to *Tenants* that are the requesting user’s *Tenant* or children thereof will appear in the output of a GET request, and the same constraints are placed on the allowed values of the tenantId field of a PUT request to update a new *Delivery Service*. Furthermore, the only *Delivery Services* a user may delete are those assigned to a *Tenant* that is either the same *Tenant* as the user’s *Tenant*, or a descendant thereof.

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** A string containing a comma-separated list, or an array of strings defining the *Geo Limit Countries*<sup>2</sup>

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*<sup>?</sup>

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

---

<sup>2</sup> These fields must be defined if and only if `geoLimit` is non-zero

**originShield** A *Origin Shield* string

**profileId** An optional *ID* of the *Profile* with which this *Delivery Service* will be associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

**routingName** The *Routing Name* of this *Delivery Service*

---

**Note:** If the *Delivery Service* has *SSL Keys*, then *routingName* is not allowed to change as that would invalidate the *SSL Key*

---

**signed** true if and only if *signingAlgorithm* is not null, false otherwise

**signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*

**rangeSliceBlockSize** An integer that defines the byte block size for the *ATS Slice Plugin*. It can only and must be set if *rangeRequestHandling* is set to 3. It can only be between (inclusive) 262144 (256KB) - 33554432 (32MB).

**sslKeyVersion** This integer indicates the *SSL Key Version*

**tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*

**tlsVersions** An array of explicitly supported *TLS Versions*

New in version 4.0.

**topology** The unique name of the *Topology* that this *Delivery Service* is assigned to

**trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*

**trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*

**typeId** The integral, unique identifier of the *Type* of this *Delivery Service*

**xmlId** This *Delivery Service*'s *xml\_id*

**Note:** While this field **must** be present, it is **not** allowed to change; this must be the same as the `xml_id` the *Delivery Service* already has. This should almost never be different from the *Delivery Service*'s `displayName`.

#### 745: Request Example

```
PUT /api/4.0/deliveryservices/6 HTTP/1.1
```

```
Host: trafficops.infra.ciab.test
```

```
User-Agent: python-requests/2.24.0
```

```
Accept-Encoding: gzip, deflate
```

```
Accept: */*
```

```
Connection: keep-alive
```

```
Cookie: mojolicious=...
```

```
Content-Length: 1585
```

```
Content-Type: application/json
```

```
{
  "active": false,
  "anonymousBlockingEnabled": false,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "checkPath": null,
  "consistentHashRegex": null,
  "consistentHashQueryParams": [],
  "deepCachingType": "NEVER",
  "displayName": "test",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "ecsEnabled": true,
  "edgeHeaderRewrite": null,
  "firstHeaderRewrite": null,
  "fqPacingRate": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectUrl": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": false,
```

(continues on next page)

(continued from previous page)

```

    "lastHeaderRewrite": null,
    "logsEnabled": true,
    "longDesc": "A Delivery Service created expressly for API_
↪documentation examples",
    "maxDnsAnswers": null,
    "missLat": 0,
    "missLong": 0,
    "maxOriginConnections": 0,
    "maxRequestHeaderBytes": 131072,
    "midHeaderRewrite": null,
    "multiSiteOrigin": false,
    "orgServerFqdn": "http://origin.infra.ciab.test",
    "originShield": null,
    "profileId": null,
    "protocol": 0,
    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "routingName": "test",
    "serviceCategory": null,
    "signed": false,
    "signingAlgorithm": null,
    "rangeSliceBlockSize": null,
    "sslKeyVersion": null,
    "tenant": "root",
    "tenantId": 1,
    "tlsVersions": null,
    "topology": null,
    "trRequestHeaders": null,
    "trResponseHeaders": null,
    "type": "HTTP",
    "typeId": 1,
    "xmlId": "test"
}

```

## Response Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** An array of strings defining the *Geo Limit Countries*

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in :rfc:3339 format

Changed in version 4.0: Prior to API version 4.0, this property used *Traffic Ops's Custom Date/Time Format*.

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using **pattern**.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**maxRequestHeaderBytes** The *Max Request Header Bytes*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*



- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- serviceCategory** The name of the *Service Category* with which the *Delivery Service* is associated
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- tlsVersions** An array of explicitly supported *TLS Versions*
- New in version 4.0.
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

#### 746: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 08 Jun 2021 00:34:04
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
    ↳tTncbRoJR+pyykVbEc6nWyoFnhlJzsbge9hVZfw+WK28rzSGECZ/
    ↳Q4zXTQtFjHWY5G+0Rk4w9GKrSFK3k+u5Ng==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Jun 2021 23:34:04 GMT
Content-Length: 840

{ "alerts": [
  {
    "text": "Delivery Service update was successful",
    "level": "success"
  }
],
"response": [{
  "active": false,
  "anonymousBlockingEnabled": false,
  "ccrDnsTtl": null,
  "cdnId": 2,
  "cdnName": null,
  "checkPath": null,
  "consistentHashQueryParams": [],
  "consistentHashRegex": null,
  "deepCachingType": "NEVER",
  "displayName": "test",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "ecsEnabled": true,
  "edgeHeaderRewrite": null,
  "exampleURLs": null,
  "firstHeaderRewrite": null,
  "fqPacingRate": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectURL": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "id": 6,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": false,
  "lastHeaderRewrite": null,
  "lastUpdated": "2021-06-07T23:34:04.831215Z",
  "logsEnabled": true,
```

(continues on next page)

(continued from previous page)

```

    "longDesc": "A Delivery Service created expressly for API_
    ↪documentation examples",
    "matchList": [
        {
            "type": "HOST_REGEX",
            "setNumber": 0,
            "pattern": ".*\\.test\\.\\.*"
        }
    ],
    "maxDnsAnswers": null,
    "maxOriginConnections": 0,
    "maxRequestHeaderBytes": 131072,
    "midHeaderRewrite": null,
    "missLat": 0,
    "missLong": 0,
    "multiSiteOrigin": false,
    "originShield": null,
    "orgServerFqdn": "http://origin.infra.ciab.test",
    "profileDescription": null,
    "profileId": null,
    "profileName": null,
    "protocol": 0,
    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "rangeSliceBlockSize": null,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "test",
    "serviceCategory": null,
    "signed": false,
    "signingAlgorithm": null,
    "sslKeyVersion": null,
    "tenant": "root",
    "tenantId": 1,
    "tlsVersions": null,
    "topology": null,
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "type": "HTTP",
    "typeId": 1,
    "xmlId": "test"
  }
}
```

## DELETE

Deletes the target *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 1040, 1</sup>

**Permissions Required** DELIVERY-SERVICE:DELETE, DELIVERY-SERVICE:READ, CDN:READ, TYPE:READ

**Response Type** undefined

## Request Structure

Table 309: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> to be deleted

### 747: Request Example

```
DELETE /api/4.0/deliveryservices/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 748: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: w9NlQpJJEl56r6iYq/
→fk8o5WfAXeUS5XR9yDHvKUgPO8lYEo8YyftaSF0MPFseeOk60dk6kQo+MLYTDIAhhRxw==
X-Server-Name: traffic_ops_golang/
Date: Tue, 20 Nov 2018 14:56:37 GMT
Content-Length: 57
```

(continues on next page)

(continued from previous page)

```
{ "alerts": [
  {
    "text": "ds was deleted.",
    "level": "success"
  }
]}
```

## deliveryservices/{ID}/capacity

See also:

*Health Protocol*

### GET

Retrieves the usage percentages of a servers associated with a *Delivery Service*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ

**Response Type** Object

### Request Structure

Table 310: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the <i>Delivery Service</i> of interest

### Response Structure

**availablePercent** The percent of servers assigned to this *Delivery Service* that is available - the allowed traffic level in terms of data per time period for all *cache servers* that remains unused

**unavailablePercent** The percent of servers assigned to this *Delivery Service* that is unavailable - the allowed traffic level in terms of data per time period for all *cache servers* that can't be used because the servers are deemed unhealthy

<sup>1</sup> Users will only be able to see capacity details for the *Delivery Services* their *Tenant* is allowed to see.

**utilizedPercent** The percent of servers assigned to this *Delivery Service* that is currently in use - the allowed traffic level in terms of data per time period that is currently devoted to servicing requests

**maintenancePercent** The percent of servers assigned to this *Delivery Service* that is unavailable due to server maintenance - the allowed traffic level in terms of data per time period that is unavailable because servers have intentionally been marked offline by administrators

#### 749: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 14:41:27 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→++dFR9V1c60CHGNwMjX6JSFEjHreXcL4QnhT03hiv04ByY379aLpL40rOzX2bPgJgpR94+f6jZ0+iDIyT
Content-Length: 134

{ "response": {
  "availablePercent": 99.9993696969697,
  "unavailablePercent": 0,
  "utilizedPercent": 0.0006303030303030303,
  "maintenancePercent": 0
}}
```

`deliveryservices/{{ID}}/health`

See also:

*Health Protocol*

## GET

Retrieves the health of all *Cache Groups* assigned to a particular *Delivery Service*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ, CACHE-GROUP:READ

**Response Type** Object

## Request Structure

Table 311: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the Delivery service for which <i>Cache Groups</i> will be displayed

## Response Structure

**cachegroups** An array of objects that represent the health of each *Cache Group* assigned to this *Delivery Service*

**name** A string that is the *name of the Cache Group* represented by this object

**offline** The number of OFFLINE *cache servers* within this *Cache Group*

**online** The number of ONLINE *cache servers* within this *Cache Group*

**totalOffline** Total number of OFFLINE *cache servers* assigned to this *Delivery Service*

**totalOnline** Total number of ONLINE *cache servers* assigned to this *Delivery Service*

## 750: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

<sup>1</sup> Users will only be able to see *Cache Group* health details for the *Delivery Services* their *Tenant* is allowed to see.

(continued from previous page)

```
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 14:43:43 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳ GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: ↳
    ↳ KpXViXeAgch58ueQqdyU8NuINBw1EUedE6Rv2ewcLUajJp6kowdbVynpwW7XiSvAyHdtClIOuT30khIm
Content-Length: 115

{ "response": {
  "totalOffline": 0,
  "totalOnline": 1,
  "cachegroups": [
    {
      "offline": 0,
      "name": "CDN_in_a_Box_Edge",
      "online": 1
    }
  ]
}}
```

## deliveryservices/{{ID}}/regexes

### GET

Retrieves routing regular expressions for a specific *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** Array

<sup>1</sup> Users will only be able to view and create regular expressions for the *Delivery Services* their *Tenant* is allowed to see.



## Request Structure

Table 312: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected

Table 313: Request Query Parameters

Name	Required	Description
id	no	Show only the <i>Delivery Service</i> regular expression that has this integral, unique identifier
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 751: Request Example

```
GET /api/4.0/deliveryservices/1/regexes HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- id** The integral, unique identifier of this regular expression
- pattern** The actual regular expression - \s are escaped
- setNumber** The order in which the regular expression is evaluated against requests
- type** The integral, unique identifier of the *Type* of this regular expression
- typeName** The *Type* of regular expression - determines that against which it will be evaluated

### 752: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: fW9Fde4WRpp2ShRAC41P9s/PhU71LI/
→SEzHgYjGqfzhk45wq0kpaWy76JvPfLpowY8eDTp8Y8TL5rNGEc+bM+A==
X-Server-Name: traffic_ops_golang/
Date: Tue, 27 Nov 2018 20:56:43 GMT
Content-Length: 100

{ "response": [
  {
    "id": 1,
    "type": 31,
    "typeName": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.demo1\\.\\.\\.*"
  }
]}
```

## POST

Creates a routing regular expression for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 1040, 1</sup>

**Permissions Required** DELIVERY-SERVICE:UPDATE, DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** Object

## Request Structure

Table 314: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected

**pattern** The actual regular expression

**Warning:** Be sure that \s are escaped, or the expression may not work as intended!

**setNumber** The order in which this regular expression should be checked

**type** The integral, unique identifier of a routing regular expression type

#### 753: Request Example

```
POST /api/4.0/deliveryservices/1/regexes HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 55
Content-Type: application/json

{
  "pattern": ".*\\.foo-bar\\.\\.\"",
  "type": 31,
  "setNumber": 1
}
```

## Response Structure

**id** The integral, unique identifier of this regular expression

**pattern** The actual regular expression - \s are escaped

**setNumber** The order in which the regular expression is evaluated against requests

**type** The integral, unique identifier of the type of this regular expression

**typeName** The type of regular expression - determines that against which it will be evaluated

#### 754: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: kS5dRzAhFKE7vfzHK7XVIwpMOjztksk9MU+qtj5YU/
  ↳1oxVHmqNbJ12Fe00IJsZJCXbYlnBS04sCI95Sz5wed1Q==
X-Server-Name: traffic_ops_golang/
Date: Wed, 28 Nov 2018 17:00:42 GMT
Content-Length: 188

{ "alerts": [
```

(continues on next page)

(continued from previous page)

```
{
  "text": "Delivery service regex creation was successful.",
  "level": "success"
},
],
"response": {
  "id": 2,
  "type": 31,
  "typeName": "HOST_REGEX",
  "setNumber": 1,
  "pattern": ".*\\.foo-bar\\.\\.\\.*"
}}
```

`deliveryservices/{{ID}}/regexes/{{rID}}`

## PUT

Updates a routing regular expression.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:UPDATE, DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** Object

## Request Structure

Table 315: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected
rID	The integral, unique identifier of the routing regular expression being inspected

**pattern** The actual regular expression

**Warning:** Be sure that \s are escaped, or the expression may not work as intended!

**setNumber** The order in which this regular expression should be checked

**type** The integral, unique identifier of a routing regular expression type

<sup>1</sup> Users will only be able to view, delete and update regular expressions for the *Delivery Services* their *Tenant* is allowed to see.

#### 755: Request Example

```
PUT /api/4.0/deliveryservices/1/regexes/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 55
Content-Type: application/json

{
  "pattern": ".*\\.foo-bar\\.\\.\\.\"",
  "type": 33,
  "setNumber": 1
}
```

### Response Structure

- id** The integral, unique identifier of this regular expression
- pattern** The actual regular expression - \s are escaped
- setNumber** The order in which the regular expression is evaluated against requests
- type** The integral, unique identifier of the type of this regular expression
- typeName** The type of regular expression - determines that against which it will be evaluated

#### 756: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: kS5dRzAhFKE7vfzHK7XVIwpMOjztk9MU+qtj5YU/
  ↳1oxVHmqNbJ12Fe00IJsZJCXbYlnBS04sCI95Sz5wed1Q==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 17:54:58 GMT
Content-Length: 188

{ "alerts": [
  {
```

(continues on next page)

(continued from previous page)

```
        "text": "Delivery service regex creation was successful.",
        "level": "success"
    }
],
"response": {
    "id": 2,
    "type": 33,
    "typeName": "PATH_REGEX",
    "setNumber": 1,
    "pattern": ".*\\.foo-bar\\.\\.\\.*"
}}
```

DELETE

Deletes a routing regular expression.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 1040, 1</sup>

**Permissions Required** DELIVERY-SERVICE:UPDATE, DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** undefined

Request Structure

Table 316: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being inspected
rID	The integral, unique identifier of the routing regular expression being inspected

#### 757: Request Example

```
DELETE /api/4.0/deliveryservices/1/regexes/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 758: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 8oEa78x7f/o39LIS98W6G+UqE6cX/
    ↳Iw4v3mMHvbAsliWHALuDYZRz3V0tA6jzfGQKpB040m8qaVG+zWRrBVoCmQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 18:44:00 GMT
Content-Length: 76

{ "alerts": [
  {
    "text": "deliveryservice_regex was deleted.",
    "level": "success"
  }
]}
```

## deliveryservices/{{ID}}/routing

### GET

Retrieves the aggregated routing percentages for a given *Delivery Service*. This is accomplished by polling stats from all online Traffic Routers via the */crs/stats* route.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

<sup>1</sup> Users will only be able to view routing details for the *Delivery Services* their *Tenant* is allowed to see.

**Permissions Required** DELIVERY-SERVICE:READ

**Response Type** Object

---

**Note:** Only HTTP or DNS *Delivery Services* can be requested.

---

## Request Structure

Table 317: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the <i>Delivery Service</i> of interest

759: Request Example

```
GET /api/4.0/deliveryservices/1/routing HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cz** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by a *Coverage Zone File*
- deepCz** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by a *Deep Coverage Zone File*
- dsr** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by sending the client to an overflow *Delivery Service*
- err** The percent of requests to online Traffic Routers for this *Delivery Service* that resulted in an error
- fed** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by sending the client to a federated CDN
- geo** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied using 3rd party geographic IP mapping
- miss** The percent of requests to online Traffic Routers for this *Delivery Service* that could not be satisfied
- regionalAlternate** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied by sending the client to the alternate, Regional Geo-blocking URL



**regionalDenied** The percent of requests to online Traffic Routers for this *Delivery Service* that were denied due to geographic location policy

**staticRoute** The percent of requests to online Traffic Routers for this *Delivery Service* that were satisfied with *Static DNS Entries*

#### 760: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Fri, 30 Nov 2018 15:08:07 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: UgPziRC/5u4+CfkZ9xm0EkEzjjJVu6cwBrFd/n3xH/
→ZmlkaXkQaa1y4+B7DyE46vxFLYE0OD0cQchyn7JkoQOg==
Content-Length: 132

{ "response": {
  "cz": 79,
  "deepCz": 0.50,
  "dsr": 0,
  "err": 0,
  "fed": 0.25,
  "geo": 20,
  "miss": 0.25,
  "regionalAlternate": 0,
  "regionalDenied": 0,
  "staticRoute": 0
}}
```

`deliveryservices/{{ID}}/safe`

**PUT**

Allows a user to edit metadata fields of a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

<sup>1</sup> Only those *Delivery Services* assigned to *Tenants* that are the requesting user's *Tenant* or children thereof may

**Permissions Required** DELIVERY-SERVICE-SAFE:UPDATE, DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** Array

## Request Structure

Table 318: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Delivery Service</i> being modified

**displayName** A string that is the *Display Name*

**infoUrl** An optional<sup>2</sup> string containing the *Info URL*

**longDesc** An optional<sup>2</sup> string containing the *Long Description* of this *Delivery Service*

761: Request Example

```
PUT /api/4.0/deliveryservices/1/safe HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 132
Content-Type: application/json

{
  "displayName": "test",
  "infoUrl": "this is not even a real URL",
  "longDesc": "this is a description of the delivery service"
}
```

## Response Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

---

be modified with this endpoint.

<sup>2</sup> If these fields are not present in the request body they are *implicitly set to null*.

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** An array of strings defining the *Geo Limit Countries*

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

**initialDispersion** The *Initial Dispersion*

**innerHeaderRewrite** A set of *Inner Header Rewrite Rules*

**ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*

**lastHeaderRewrite** A set of *Last Header Rewrite Rules*

**lastUpdated** The date and time at which this *Delivery Service* was last updated, in :rfc:3339 format

Changed in version 4.0: Prior to API version 4.0, this property used *Traffic Ops's Custom Date/Time Format*.

**logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*

**longDesc** The *Long Description* of this *Delivery Service*

**matchList** The *Delivery Service's Match List*

**pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)

**setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.

**type** The type of match performed using **pattern**.

**maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*

**maxOriginConnections** The *Max Origin Connections*

**midHeaderRewrite** A set of *Mid Header Rewrite Rules*

**missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*

**missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*

**multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*

**orgServerFqdn** The *Origin Server Base URL*

**originShield** A *Origin Shield* string

**profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated

**profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated

**profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated

**protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

**qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*

**rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*

**regexRemap** A *Regex Remap Expression*

**regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*

**remapText** *Raw Remap Text*

**signed** true if and only if `signingAlgorithm` is not null, false otherwise

**signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*

**rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if `rangeRequestHandling` is set to 3.

**sslKeyVersion** This integer indicates the *SSL Key Version*

**tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*

**tlsVersions** A list of explicitly supported *TLS Versions*

New in version 4.0.

**topology** The unique name of the *Topology* that this *Delivery Service* is assigned to

**trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*

**trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*

**type** The *Type* of this *Delivery Service*

**typeId** The integral, unique identifier of the *Type* of this *Delivery Service*

**xmlId** This *Delivery Service*'s *xml\_id*

## 762: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 08 Jun 2021 00:53:26.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: Ys/SfWWijsXCNXEqZ84oldfyXTgMe8UE/
→wWb53VU390H7kWOXF1BH5Hg7Y40nCgXoWEqcaBq5+WCZg0bYuJdAA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Jun 2021 23:53:26 GMT
Content-Length: 903

{ "alerts": [{
  "text": "Delivery Service safe update successful.",
  "level": "success"
}],
"response": [{
```

(continues on next page)

(continued from previous page)

```

"active": true,
"anonymousBlockingEnabled": false,
"ccrDnsTtl": null,
"cdnId": 2,
"cdnName": "CDN-in-a-Box",
"checkPath": null,
"consistentHashQueryParams": [],
"consistentHashRegex": null,
"deepCachingType": "NEVER",
"displayName": "test",
"dnsBypassCname": null,
"dnsBypassIp": null,
"dnsBypassIp6": null,
"dnsBypassTtl": null,
"dscp": 0,
"ecsEnabled": false,
"edgeHeaderRewrite": null,
"exampleURLs": [
    "http://video.demo2.mycdn.ciab.test",
    "https://video.demo2.mycdn.ciab.test"
],
"firstHeaderRewrite": null,
"fqPacingRate": null,
"geoLimit": 0,
"geoLimitCountries": null,
"geoLimitRedirectURL": null,
"geoProvider": 0,
"globalMaxMbps": null,
"globalMaxTps": null,
"httpBypassFqdn": null,
"id": 1,
"infoUrl": "this is not even a real URL",
"initialDispersion": 1,
"innerHeaderRewrite": null,
"ipv6RoutingEnabled": true,
"lastHeaderRewrite": null,
"lastUpdated": "2021-06-07T23:53:26.139899Z",
"logsEnabled": true,
"longDesc": "this is a description of the delivery service",
"matchList": [
    {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo2\\.\\.\\.*"
    }
],

```

(continues on next page)

(continued from previous page)

```

    "maxDnsAnswers": null,
    "maxOriginConnections": 0,
    "maxRequestHeaderBytes": 0,
    "midHeaderRewrite": null,
    "missLat": 42,
    "missLong": -88,
    "multiSiteOrigin": true,
    "originShield": null,
    "orgServerFqdn": "http://origin.infra.ciab.test",
    "profileDescription": null,
    "profileId": null,
    "profileName": null,
    "protocol": 2,
    "qstringIgnore": 0,
    "rangeRequestHandling": 0,
    "rangeSliceBlockSize": null,
    "regexRemap": null,
    "regionalGeoBlocking": false,
    "remapText": null,
    "routingName": "video",
    "serviceCategory": null,
    "signed": false,
    "signingAlgorithm": null,
    "sslKeyVersion": null,
    "tenant": "root",
    "tenantId": 1,
    "tlsVersions": null,
    "topology": "demo1-top",
    "trResponseHeaders": null,
    "trRequestHeaders": null,
    "type": "DNS",
    "typeId": 5,
    "xmlId": "demo2"
  ]}

```

`deliveryservices/{{ID}}/servers`

**Caution:** It's often much easier to use `deliveryservices/{{xml_id}}/servers` instead

GET

Retrieves properties of Edge-Tier servers assigned to a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** CACHE-GROUP:READ, CDN:READ, TYPE:READ, PROFILE:READ, DELIVERY-SERVICE:READ, SERVER:READ

**Response Type** Array

## Request Structure

Table 319: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the Delivery service for which servers will be displayed

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** An integral, unique identifier the CDN to which the server belongs

**cdnName** The name of the CDN to which the server belongs

**domainName** The domain name part of the FQDN of the server

**guid** Optionally represents an identifier used to uniquely identify the server

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS requests - 443 in most cases

**id** An integral, unique identifier for the server

**iloIpAddress** The IPv4 address of the lights-out-management port<sup>1</sup>

**iloIpGateway** The IPv4 gateway address of the lights-out-management port<sup>Page 1040, 1</sup>

**iloIpNetmask** The IPv4 subnet mask of the lights-out-management port<sup>Page 1040, 1</sup>

**iloPassword** The password of the of the lights-out-management user - displays as \*\*\*\*\* unless the requesting user has the 'admin' role<sup>Page 1040, 1</sup>

**iloUsername** The user name for lights-out-management<sup>Page 1040, 1</sup>

**interfaces** An array of interface and IP address information

---

<sup>1</sup> See the [Wikipedia article on Out-of-Band Management](#) for more information.



**max\_bandwidth** The maximum allowed bandwidth for this interface to be considered “healthy” by Traffic Monitor. This has no effect if *monitor* is not true. Values are in kb/s. The value *null* means “no limit”.

**monitor** A boolean indicating if Traffic Monitor should monitor this interface

**mtu** The MTU to configure for *interfaceName*

**See also:**

[The Wikipedia article on Maximum Transmission Unit](#)

**name** The network interface name used by the server.

**ipAddresses** An array of the IP address information for the interface

**address** The IPv4 or IPv6 address and subnet mask of the server - applicable for the interface name

**gateway** The IPv4 or IPv6 gateway address of the server - applicable for the interface name

**service\_address** A boolean determining if content will be routed to the IP address

**lastUpdated** The time and date at which this server was last updated, in *Traffic Ops's Custom Date/Time Format*

**mgmtIpAddress** The IPv4 address of the server's management port

**mgmtIpGateway** The IPv4 gateway of the server's management port

**mgmtIpNetmask** The IPv4 subnet mask of the server's management port

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status (will be empty if not offline)

**physLocation** The name of the *Physical Location* at which the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* at which the server resides

**profile** List of *Name* of the *Profiles* assigned to this server

**rack** A string indicating “rack” location

**routerHostName** The human-readable name of the router

**routerPortName** The human-readable name of the router port

**status** The Status of the server

See also:

*Health Protocol*

**statusId** An integral, unique identifier for the status of the server

See also:

*Health Protocol*

**tcpPort** The default port on which the main application listens for incoming TCP connections - 80 in most cases

**type** The name of the type of this server

**typeId** An integral, unique identifier for the type of this server

**updPending** true if the server has updates pending, false otherwise

### 763: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: MaIva080Sjysr4bCkuXFEMf3o6mOqga1aM4IHN/
    ↳tcP2aa1iXEmA5IrHB7DaqNX/2vGHLXvN+01FEAR/lRNqr1w==
X-Server-Name: traffic_ops_golang/
Date: Wed, 14 Nov 2018 21:28:23 GMT
Content-Length: 891

{ "response": [
  {
    "cachegroup": "CDN_in_a_Box_Edge",
    "cachegroupId": 7,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "domainName": "infra.ciab.test",
    "guid": null,
    "hostName": "edge",
    "httpsPort": 443,
    "id": 10,
    "iloIpAddress": "",
    "iloIpGateway": "",
    "iloIpNetmask": "",
    "iloPassword": "",
    "iloUsername": "",
```

(continues on next page)

(continued from previous page)

```

    "lastUpdated": "2018-11-14 21:08:44+00",
    "mgmtIpAddress": "",
    "mgmtIpGateway": "",
    "mgmtIpNetmask": "",
    "offlineReason": "",
    "physLocation": "Apachecon North America 2018",
    "physLocationId": 1,
    "profileNames": ["ATS_EDGE_TIER_CACHE"],
    "rack": "",
    "routerHostName": "",
    "routerPortName": "",
    "status": "REPORTED",
    "statusId": 3,
    "tcpPort": 80,
    "type": "EDGE",
    "typeId": 11,
    "updPending": false,
    "interfaces": [{
        "ipAddresses": [
            {
                "address": "172.16.239.100",
                "gateway": "172.16.239.1",
                "service_address": true
            },
            {
                "address": "fc01:9400:1000:8::100",
                "gateway": "fc01:9400:1000:8::1",
                "service_address": true
            }
        ],
        "max_bandwidth": 0,
        "monitor": true,
        "mtu": 1500,
        "name": "eth0"
    }]
}
]]

```

## deliveryservices/{{ID}}/servers/eligible

**Caution:** This endpoint may not work as advertised, and its use is therefore discouraged!

### GET

Retrieves properties of *Edge-tier cache servers* eligible for assignment to a particular *Delivery Service*. Eligibility is determined based on the following properties:

- The name of the server's *Type* must match one of the glob patterns `EDGE*`, `ORG*`
- The server and *Delivery Service* must belong to the same CDN
- If the *Delivery Service* has *Required Capabilities*, an *Edge-tier cache server* must have all of those defined capabilities

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DELIVERY-SERVICE:READ, SERVER:READ,  
CACHE-GROUP:READ, TYPE:READ, CDN:READ

**Response Type** Array

### Request Structure

Table 320: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the Delivery service for which servers will be displayed

### Response Structure

**cachegroup** A string which is the *Name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** An integral, unique identifier the CDN to which the server belongs

**cdnName** The name of the CDN to which the server belongs

**domainName** The domain name part of the FQDN of the server

**guid** Optionally represents an identifier used to uniquely identify the server

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS requests - 443 in most cases

**id** An integral, unique identifier for the server

**iloIpAddress** The IPv4 address of the lights-out-management port<sup>1</sup>

**iloIpGateway** The IPv4 gateway address of the lights-out-management port<sup>Page 1040, 1</sup>

**iloIpNetmask** The IPv4 subnet mask of the lights-out-management port<sup>Page 1040, 1</sup>

**iloPassword** The password of the of the lights-out-management user - displays as \*\*\*\*\* unless the requesting user has the ‘admin’ role)<sup>Page 1040, 1</sup>

**iloUsername** The user name for lights-out-management<sup>Page 1040, 1</sup>

**interfaces** An array of interface and IP address information

**max\_bandwidth** The maximum allowed bandwidth for this interface to be considered “healthy” by Traffic Monitor. This has no effect if *monitor* is not true. Values are in kb/s. The value *null* means “no limit”.

**monitor** A boolean indicating if Traffic Monitor should monitor this interface

**mtu** The MTU to configure for *interfaceName*

**See also:**

[The Wikipedia article on Maximum Transmission Unit](#)

**name** The network interface name used by the server.

**ipAddresses** An array of the IP address information for the interface

**address** The IPv4 or IPv6 address and subnet mask of the server - applicable for the interface *name*

**gateway** The IPv4 or IPv6 gateway address of the server - applicable for the interface *name*

**service\_address** A boolean determining if content will be routed to the IP address

**lastUpdated** The time and date at which this server was last updated, in *Traffic Ops’s Custom Date/Time Format*

**mgmtIpAddress** The IPv4 address of the server’s management port

**mgmtIpGateway** The IPv4 gateway of the server’s management port

**mgmtIpNetmask** The IPv4 subnet mask of the server’s management port

---

<sup>1</sup> See [the Wikipedia article on Out-of-Band Management](#) for more information.

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status (will be empty if not offline)

**physLocation** The name of the *Physical Location* at which the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* at which the server resides

**profile** The *Name* of the *Profile* assigned to this server

**profileDesc** A *Description* of the *Profile* assigned to this server

**profileId** The *ID* of the *Profile* assigned to this server

**rack** A string indicating “rack” location

**routerHostName** The human-readable name of the router

**routerPortName** The human-readable name of the router port

**status** The Status of the server

**See also:**

*Health Protocol*

**statusId** An integral, unique identifier for the status of the server

**See also:**

*Health Protocol*

**tcpPort** The default port on which the main application listens for incoming TCP connections - 80 in most cases

**type** The name of the *Type* of this server

**typeId** An integral, unique identifier for the *Type* of this server

**updPending** true if the server has updates pending, false otherwise

#### 764: Response Example

```
{ "response": [
  {
    "cachegroup": "CDN_in_a_Box_Edge",
    "cachegroupId": 7,
    "cdnId": 2,
    "cdnName": "CDN-in-a-Box",
    "domainName": "infra.ciab.test",
    "guid": null,
    "hostName": "edge",
    "httpsPort": 443,
    "id": 10,
    "iloIpAddress": "",
    "iloIpGateway": "",
    "iloIpNetmask": "",
```

(continues on next page)

(continued from previous page)

```

        "iloPassword": "",
        "iloUsername": "",
        "lastUpdated": "2018-10-30 16:01:12+00",
        "mgmtIpAddress": "",
        "mgmtIpGateway": "",
        "mgmtIpNetmask": "",
        "offlineReason": "",
        "physLocation": "Apachecon North America 2018",
        "physLocationId": 1,
        "profile": "ATS_EDGE_TIER_CACHE",
        "profileDesc": "Edge Cache - Apache Traffic Server",
        "profileId": 9,
        "rack": "",
        "routerHostName": "",
        "routerPortName": "",
        "status": "REPORTED",
        "statusId": 3,
        "tcpPort": 80,
        "type": "EDGE",
        "typeId": 11,
        "updPending": false,
        "interfaces": [{
            "ipAddresses": [
                {
                    "address": "172.16.239.100",
                    "gateway": "172.16.239.1",
                    "service_address": true
                },
                {
                    "address": "fc01:9400:1000:8::100",
                    "gateway": "fc01:9400:1000:8::1",
                    "service_address": true
                }
            ],
            "max_bandwidth": 0,
            "monitor": true,
            "mtu": 1500,
            "name": "eth0"
        }]
    }
}]

```

`deliveryservices/{{ID}}/urlkeys`

GET

See also:

`deliveryservices/xmlId/{{xmlid}}/urlkeys`

Retrieves URL signing keys for a *Delivery Service*.

**Caution:** This method will return the *Delivery Service*'s **PRIVATE** URL signing keys! Be wary of using this endpoint and **NEVER** share the output with anyone who would be unable to see it on their own.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DS-SECURITY-KEY:READ, DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

Table 321: Request Path Parameters

Name	Description
id	Filter for the <i>Delivery Service</i> identified by this integral, unique identifier



#### 765: Request Example

```
GET /api/4.0/deliveryservices/1/urlkeys HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**key<N>** The private URL signing key for this *Delivery Service* as a base-64-encoded string, where <N> is the “generation” of the key e.g. the first key will always be named “key0”. Up to 16 concurrent generations are retained at any time (<N> is always on the interval [0,15])

#### 766: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 16:34:56
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: cTc5OPE3hM+CiyQPCy36zD2tsQcFkvIqQ7/
    ↳D82WGMWHm+ACW3YbcKhgPnSQU6+Tuj4jya52Kx9+nw5+OonFvPQ==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 15:34:56 GMT
Content-Length: 533

{
  "response": {
    "key0": "...",
    "key1": "...",
    "key2": "...",
    "key3": "...",
    "key4": "...",
    "key5": "...",
    "key6": "...",
    "key7": "...",
    "key8": "...",
    "key9": "..."
```

(continues on next page)

(continued from previous page)

```
    "key10": "...",
    "key11": "...",
    "key12": "...",
    "key13": "...",
    "key14": "...",
    "key15": "...",
  }
}
```

DELETE

See also:

*deliveryservices/xmlId/{xmlid}/urlkeys*

Deletes URL signing keys for a *Delivery Service*.

- Auth. Required** Yes
- Roles Required** “admin” or “operations”
- Permissions Required** DS-SECURITY-KEY:DELETE, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE
- Response Type** Object

Request Structure

Table 322: Request Path Parameters

Name	Description
id	Filter for the <i>Delivery Service</i> identified by this integral, unique identifier

Response Structure

767: Response Example

```
{
  "alerts": [{
    "level": "success",
    "text": "Successfully deleted URL Sig keys from Traffic Vault"
  }]
}
```

## deliveryservices\_regexes

### GET

Retrieves routing regular expressions for all *Delivery Services*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ

**Response Type** Array

### Request Structure

No parameters available

### Response Structure

**dsName** The name of the *Delivery Service* represented by this object

**regexes** An array of objects that represent various routing regular expressions used by dsName

**pattern** The actual regular expression - \s are escaped

**setNumber** The order in which the regular expression is evaluated against requests

**type** The type of regular expression - determines that against which it will be evaluated

#### 768: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
➔GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +2MI+Q/NJqTizlMR/MhPAL+yu6/z/
➔Yqvo5fD08F593RM0mK6dX/Al4wARbEG+HqaJNgSCRPSiLVATusrmnnCMA==
X-Server-Name: traffic_ops_golang/
```

(continues on next page)

<sup>1</sup> If tenancy is used, then users (regardless of role) will only be able to see the routing regular expressions used by *Delivery Services* their tenant has permissions to see.

(continued from previous page)

```
Date: Tue, 27 Nov 2018 19:22:59 GMT
Content-Length: 110

{ "response": [
  {
    "regexes": [
      {
        "type": "HOST_REGEX",
        "setNumber": 0,
        "pattern": ".*\\.demo1\\.\\.*"
      }
    ],
    "dsName": "demo1"
  }
]}
```

## deliveryservices\_required\_capabilities

### GET

Gets all associations of *Server Capability* to *Delivery Services*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

Table 323: Request Query Parameters

Name	Re-quired	Description
deliv-ery-Servi-ceID	no	Filter <i>Server Capability</i> associations by <i>Delivery Service</i> integral, unique identifier
xm-IID	no	Filter <i>Server Capability</i> associations by <i>Delivery Service xml_id</i>
re-quired-Capa-bility	no	Filter <i>Server Capability</i> associations by <i>Server Capability</i> name
or-derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sor-tOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

### 769: Request Example

```
GET /api/4.0/deliveryservices_required_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**deliveryServiceID** The associated *Delivery Service*’s integral, unique identifier

**xmlID** The associated *Delivery Service*’s *xml\_id*

**lastUpdated** The date and time at which this association between the *Delivery Service* and the *Server Capability* was last updated, in *Traffic Ops’s Custom Date/Time Format*

**requiredCapability** The *Server Capability*’s name

## 770: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
    ↳jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
    ↳DbBwhMAOJoeqDImoDlrLnrVjQG04AooA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 396

{
  "response": [
    {
      "deliveryServiceID": 1,
      "lastUpdated": "2019-10-07 22:05:31+00",
      "requiredCapability": "ram",
      "xmlId": "example_ds-1"
    },
    {
      "deliveryServiceID": 2,
      "lastUpdated": "2019-10-07 22:05:31+00",
      "requiredCapability": "disk",
      "xmlId": "example_ds-2"
    }
  ]
}

```

## POST

Associates a *Server Capability* with a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object

**Note:** A *Server Capability* can only be made required on a *Delivery Service* if its associated Servers already have that *Server Capability* assigned.

## Request Structure

**deliveryServiceID** The integral, unique identifier of the *Delivery Service* to be associated

**requiredCapability** The name of the *Server Capability* to be associated

771: Request Example

```
POST /api/4.0/deliveryservices_required_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "deliveryServiceID": 1,
  "requiredCapability": "disk"
}
```

## Response Structure

**deliveryServiceID** The newly associated *Delivery Service*'s integral, unique identifier

**lastUpdated** The date and time at which this association between the *Delivery Service* and the *Server Capability* was last updated, in *Traffic Ops's Custom Date/Time Format*

**requiredCapability** The newly associated *Server Capability*'s name

772: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
→Cfh2X9i3G+MQ36eH95gukJq0BOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 287

{
  "alerts": [
    {
      "level": "success",
      "text": "deliveryservice.RequiredCapability was created."
    }
  ],
  "response": {
    "deliveryServiceID": 1,
    "lastUpdated": "2019-10-07 22:15:11+00",
    "requiredCapability": "disk"
  }
}
```

## DELETE

Dissociate a *Server Capability* from a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** undefined

## Request Structure

**deliveryServiceID** The integral, unique identifier of the *Delivery Service* from which a *Server Capability* will be dissociated

**requiredCapability** The name of the *Server Capability* to dissociate

773: Request Example

```
POST /api/4.0/deliveryservices_required_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)



(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "deliveryServiceID": 1,
  "requiredCapability": "disk"
}
```

## Response Structure

### 774: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
↳eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
↳Cfh2X9i3G+MQ36eH95gukJq0BOGbQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 127

{
  "alerts": [
    {
      "level": "success",
      "text": "deliveryservice.RequiredCapability was deleted."
    }
  ]
}
```

## deliveryservices/sslkeys/add

### See also:

In most cases it is preferable to allow Traffic Ops to generate the keys via *deliveryservices/sslkeys/generate*, rather than uploading them manually using this endpoint.

### POST

Allows user to upload an SSL certificate, csr, and private key for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:CREATE, DELIVERY-SERVICE:READ

**Response Type** Object (string)

### Request Structure

**cdn** The name of the CDN to which the *Delivery Service* belongs

**certificate** An object that contains the actual components of the SSL key

**crt** The certificate for the *Delivery Service* identified by key

**csr** The csr file for the *Delivery Service* identified by key

**key** The private key for the *Delivery Service* identified by key

**key** The *xml\_id* of the *Delivery Service* to which these keys will be assigned

**version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

#### 775: Request Example

```
POST /api/4.0/deliveryservices/sslkeys/add HTTP/1.1
Host: trafficops.infra.ciab.test
Content-Type: application/json

{
  "key": "ds-01",
  "version": "1",
  "certificate": {
    "key": "some_key",
    "csr": "some_csr",
    "crt": "some_crt"
```

(continues on next page)

(continued from previous page)

```
}
}
```

## Response Structure

### 776: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "response": "Successfully added ssl keys for ds-01"
}
```

## deliveryservices/sslkeys/generate

### POST

Generates an SSL certificate, csr, and private key for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:CREATE, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object (string)

## Request Structure

**city** An optional field which, if present, will represent the resident city of the generated SSL certificate

**country** An optional field which, if present, will represent the resident country of the generated SSL certificate

**hostname** The desired hostname of the *Delivery Service*

---

**Note:** In most cases, this must be the same as the *Delivery Service* URL’

---

**key** The *xml\_id* of the *Delivery Service* for which keys will be generated

**organization** An optional field which, if present, will represent the organization for which the SSL certificate was generated

**state** An optional field which, if present, will represent the resident state or province of the generated SSL certificate

**businessUnit** An optional field which, if present, will represent the business unit for which the SSL certificate was generated

**version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

### 777: Request Example

```
POST /api/4.0/deliveryservices/sslkeys/generate HTTP/1.1
Content-Type: application/json

{
  "key": "ds-01",
  "businessUnit": "CDN Engineering",
  "version": "3",
  "hostname": "tr.ds-01.ott.kabletown.com",
  "country": "US",
  "organization": "Kabletown",
  "city": "Denver",
  "state": "Colorado"
}
```

### Response Structure

## 778: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": "Successfully created ssl keys for ds-01" }
```

### deliveryservices/sslkeys/generate/acme

New in version 4.0.

#### POST

Generates an SSL certificate and private key using ACME protocol for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:UPDATE, ACME:READ,  
DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object (string)

#### Request Structure

**authType** The certificate provider correlating to an ACME account in *cdn.conf* or Let’s Encrypt.

**key** The *xml\_id* of the *Delivery Service* for which keys will be generated<sup>1</sup>

**deliveryservice** The *xml\_id* of the *Delivery Service* for which keys will be generated<sup>Page 1227, 1</sup>

**version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key

**hostname** The desired hostname of the *Delivery Service*

---

**Note:** In most cases, this must be the same as the *Example URLs*.

---

**cdn** The name of the CDN of the *Delivery Service* for which the certs will be generated

---

<sup>1</sup> Either the key or the deliveryservice field must be provided. If both are provided, then they must match.

#### 779: Request Example

```
POST /api/4.0/deliveryservices/sslkeys/generate/acme HTTP/1.1
Content-Type: application/json

{
  "authType": "Lets Encrypt",
  "key": "ds-01",
  "deliveryservice": "ds-01",
  "version": "3",
  "hostname": "tr.ds-01.ott.kabletown.com",
  "cdn": "test-cdn"
}
```

### Response Structure

#### 780: Response Example

```
{ "alerts": [{
  "level": "success",
  "text": "Beginning async ACME call for demo1 using Lets Encrypt.
→ This may take a few minutes. Status updates can be found here: /api/
→ 4.0/async_status/1"
}]}
```

### deliveryservices/sslkeys/generate/letsencrypt

Deprecated since version ATCv6.

#### POST

Generates an SSL certificate and private key using Let's Encrypt for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:CREATE, DELIVERY-SERVICE:READ

**Response Type** Object (string)

## Request Structure

- key** The *xml\_id* of the *Delivery Service* for which keys will be generated<sup>1</sup>
- deliveryservice** The *xml\_id* of the *Delivery Service* for which keys will be generated<sup>Page 1227, 1</sup>
- version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key
- hostname** The desired hostname of the *Delivery Service*

---

**Note:** In most cases, this must be the same as the *Example URLs*.

---

- cdn** The name of the CDN of the *Delivery Service* for which the certs will be generated

### 781: Request Example

```
POST /api/4.0/deliveryservices/sslkeys/generate/letsencrypt HTTP/1.1
Content-Type: application/json

{
  "key": "ds-01",
  "deliveryservice": "ds-01",
  "version": "3",
  "hostname": "tr.ds-01.ott.kabletown.com",
  "cdn": "test-cdn"
}
```

## Response Structure

### 782: Response Example

```
{ "alerts": [{
  "level": "warning",
  "text": "This endpoint is deprecated, please use /deliveryservices/
→sslkeys/generate/acme instead."
}, {
  "level": "success",
  "text": "Beginning async ACME call for demo1 using Lets Encrypt.
→This may take a few minutes. Status updates can be found here: /api/
→4.0/async_status/1"
}]}
```

---

<sup>1</sup> Either the key or the deliveryservice field must be provided. If both are provided, then they must match.

`deliveryservices/{xml_id}/servers`

POST

Assigns *cache servers* to a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:UPDATE, SERVER:UPDATE,  
DELIVERY-SERVICE:READ, SERVER:READ

**Response Type** Object

## Request Structure

Table 324: Request Path Parameters

Name	Description
xml_id	The ‘xml_id’ of the <i>Delivery Service</i> whose server assignments are being edited

**serverNames** An array of hostname of *cache servers* to assign to this *Delivery Service*

783: Request Example

```
POST /api/4.0/deliveryservices/test/servers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 24
Content-Type: application/json

{ "serverNames": [ "edge" ] }
```

## Response Structure

**xml\_id** The *xml\_id* of the *Delivery Service* to which the servers in **serverNames** have been assigned

**serverNames** An array of hostnames of *cache servers* assigned to *Delivery Service* identified by **xml\_id**

---

<sup>1</sup> Users can only assign servers to *Delivery Services* that are visible to their *Tenant*.



## 784: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳zTpLrWiLM4xRsm8mlBQFB5KzT478AjloSyXHgtyWhebCv1YIwWltnmkjr0HFgc3GMGZODt+fyzk0Yy5Zl/
    ↳yBtJw==
X-Server-Name: traffic_ops_golang/
Date: Tue, 20 Nov 2018 15:21:50 GMT
Content-Length: 52

{ "response": {
  "serverNames": [
    "edge"
  ],
  "xmlId": "test"
}}
```

`deliveryservices/{xml_id}/urisignkeys`

### DELETE

Deletes URISigning objects for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** admin<sup>1</sup>

**Permissions Required** DS-SECURITY-KEY:DELETE, DS-SECURITY-KEY:READ, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** undefined

<sup>1</sup> URI Signing Keys can only be created, viewed, deleted, or modified on *Delivery Services* that either match the requesting user's *Tenant* or are descendants thereof.

## Request Structure

Table 325: Request Path Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Response Structure

TBD

### GET

Retrieves one or more URISigning objects for a delivery service.

**Auth. Required** Yes

**Roles Required** admin Page 1227, 1

**Permissions Required** DS-SECURITY-KEY:READ

**Response Type** undefined

## Request Structure

Table 326: Request Route Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Response Structure

**Issuer** a string describing the issuer of the URI signing object. Multiple URISigning objects may be returned in a response, see example

**renewal\_kid** a string naming the jwt key used for renewals

**keys** json array of jwt symmetric keys

**alg** this parameter repeats for each jwt key in the array and specifies the jwa encryption algorithm to use with this key, [RFC 7518](#)

**kid** this parameter repeats for each jwt key in the array and specifies the unique id for the key as defined in [RFC 7516](#)

**kty** this parameter repeats for each jwt key in the array and specifies the key type as defined in [RFC 7516](#)

**k** this parameter repeats for each jwt key in the array and specifies the base64 encoded symmetric key see [RFC 7516](#)

#### 785: Response Example

```
{ "Kabletown URI Authority": {
  "renewal_kid": "Second Key",
  "keys": [
    {
      "alg": "HS256",
      "kid": "First Key",
      "kty": "oct",
      "k": "Kh_RkUMj-fzbD37qBnDf_3e_RvQ3RP9PaSmVEpE24AM"
    },
    {
      "alg": "HS256",
      "kid": "Second Key",
      "kty": "oct",
      "k": "fZBpDBNbK2GqhwoB_DGBAsBxqQZVix04rIoLJ7p_RlE"
    }
  ]
}}
```

## POST

Assigns URISigning objects to a delivery service.

**Auth. Required** Yes

**Roles Required** admin Page 1227, 1

**Permissions Required** DS-SECURITY-KEY:CREATE

**Response Type** undefined

## Request Structure

Table 327: Request Path Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Request Structure

**Issuer** a string describing the issuer of the URI signing object. Multiple URISigning objects may be returned in a response, see example

**renewal\_kid** a string naming the jwt key used for renewals

**keys** json array of jwt symmetric keys

**alg** this parameter repeats for each jwt key in the array and specifies the jwa encryption algorithm to use with this key, [RFC 7518](#)

**kid** this parameter repeats for each jwt key in the array and specifies the unique id for the key as defined in [RFC 7516](#)

**kty** this parameter repeats for each jwt key in the array and specifies the key type as defined in [RFC 7516](#)

**k** this parameter repeats for each jwt key in the array and specifies the base64 encoded symmetric key see [RFC 7516](#)

### 786: Request Example

```
{ "Kabletown URI Authority": {
  "renewal_kid": "Second Key",
  "keys": [
    {
      "alg": "HS256",
      "kid": "First Key",
      "kty": "oct",
      "k": "Kh_RkUMj-fzbD37qBnDf_3e_RvQ3RP9PaSmVEpE24AM"
    },
    {
      "alg": "HS256",
      "kid": "Second Key",
      "kty": "oct",
      "k": "fZBpDBNbK2GqhwoB_DGBAsBxqQZVix04rIoLJ7p_RlE"
    }
  ]
}}
```

## PUT

updates URISigning objects on a delivery service.

**Auth. Required** Yes

**Roles Required** admin Page 1227, 1

**Permissions Required** DS-SECURITY-KEY:UPDATE

**Response Type** undefined

## Request Structure

Table 328: Request Path Parameters

Name	Required	Description
xml_id	yes	xml_id of the desired delivery service

## Request Structure

**Issuer** a string describing the issuer of the URI signing object. Multiple URISigning objects may be returned in a response, see example

**renewal\_kid** a string naming the jwt key used for renewals

**keys** json array of jwt symmetric keys

**alg** this parameter repeats for each jwt key in the array and specifies the jwa encryption algorithm to use with this key, [RFC 7518](#)

**kid** this parameter repeats for each jwt key in the array and specifies the unique id for the key as defined in [RFC 7516](#)

**kty** this parameter repeats for each jwt key in the array and specifies the key type as defined in [RFC 7516](#)

**k** this parameter repeats for each jwt key in the array and specifies the base64 encoded symmetric key see [RFC 7516](#)

### 787: Request Example

```
{ "Kabletown URI Authority": {
  "renewal_kid": "Second Key",
  "keys": [
    {
      "alg": "HS256",
      "kid": "First Key",
      "kty": "oct",
      "k": "Kh_RkUMj-fzbD37qBnDf_3e_RvQ3RP9PaSmVEpE24AM"
    },
    {
      "alg": "HS256",
      "kid": "Second Key",
      "kty": "oct",
      "k": "fZBpDBNbk2GqhwoB_DGBAsBxqQZVix04rIoLJ7p_RlE"
    }
  ]
}}
```

`deliveryservices/xmlId/{XMLID}/sslkeys`

**GET**

Retrieves SSL keys for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DS-SECURITY-KEY:READ, DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

Table 329: Request Path Parameters

Name	Description
XMLID	The ‘xml_id’ of the desired <i>Delivery Service</i>

Table 330: Request Query Parameters

Name	Re-quired	Description
ver-sion	no	The version number of the SSL keys to retrieve
de-code	no	If <code>true</code> , the returned keys will be decoded - if <code>false</code> , they will not be decoded

**Caution:** There’s almost certainly no good reason to request the private key! Even when “base 64-encoded” do not let **ANYONE** see this who would be unable to request it themselves!

## Response Structure

**businessUnit** An optional field which, if present, contains the business unit entered by the user when generating the SSL certificate<sup>1</sup>

**certificate** An object containing the actual generated key, certificate, and signature of the SSL keys

---

<sup>1</sup> These optional fields will be present in the response if and only if they were specified during key generation; they are optional during key generation and thus cannot be guaranteed to exist or not exist.

- crt** Base 64-encoded (or not if the decode query parameter was given and `true`) certificate for the *Delivery Service* identified by `deliveryservice`
- csr** Base 64-encoded (or not if the decode query parameter was given and `true`) csr file for the *Delivery Service* identified by `deliveryservice`
- key** Base 64-encoded (or not if the decode query parameter was given and `true`) private key for the *Delivery Service* identified by `deliveryservice`

**Caution:** There’s almost certainly no good reason to request the private key! Even when “base 64-encoded” do not let **ANYONE** see this who would be unable to request it themselves!

- cdn** The CDN of the *Delivery Service* for which the certs were generated
- city** An optional field which, if present, contains the city entered by the user when generating the SSL certificate<sup>Page 1246, 1</sup>
- country** An optional field which, if present, contains the country entered by the user when generating the SSL certificate<sup>Page 1246, 1</sup>
- deliveryservice** The ‘xml\_id’ of the *Delivery Service* for which the certificate was generated
- hostname** The hostname generated by Traffic Ops that is used as the common name when generating the certificate - this will be a FQDN for DNS *Delivery Services* and a wildcard URL for HTTP *Delivery Services*
- organization** An optional field which, if present, contains the organization entered by the user when generating certificate<sup>Page 1246, 1</sup>
- state** An optional field which, if present, contains the state entered by the user when generating certificate<sup>Page 1246, 1</sup>
- version** An integer that defines the “version” of the key - which may be thought of as the sequential generation; that is, the higher the number the more recent the key
- expiration** The expiration date of the certificate for the *Delivery Service* in **RFC 3339** format
- sans** The SANs (Subject Alternate Names) from the SSL certificate.  
New in version 4.0.

#### 788: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
{ "response": {
  "certificate": {
    "crt": "crt",
    "key": "key",
    "csr": "csr"
  },
  "deliveryservice": "my-ds",
  "cdn": "qa",
  "businessUnit": "CDN_Eng",
  "city": "Denver",
  "organization": "KableTown",
  "hostname": "foober.com",
  "country": "US",
  "state": "Colorado",
  "version": "1",
  "expiration": "2020-08-18T13:53:06Z",
  "sans": [ "*.foober.com", "*.foober2.com" ]
}}
```

## DELETE

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:DELETE, DELIVERY-  
SERVICE:READ, DS-SECURITY-KEY:READ, DELIVERY-  
SERVICE:UPDATE

**Response Type** Object (string)

## Request Structure

Table 331: Request Path Parameters

Name	Required	Description
xmlId	yes	The <i>xml_id</i> of the desired <i>Delivery Service</i>

Table 332: Request Query Parameters

Name	Required	Description
version	no	The version number of the SSL keys that shall be retrieved



## Response Structure

### 789: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 18 Mar 2020 17:36:10
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Pj+zCoOXg19nGNxcSkjib2iDjG062Y3RcEEV+0YnwbGIsLcpa0BKZleY/
    ↳qJOKT5DkSoX2qQkckUxUqdDxjVorQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 18 Mar 2020 16:36:10 GMT
Content-Length: 79

{
  "response": "Successfully deleted ssl keys for demo1"
}
```

`deliveryservices/xmlId/{xmlid}/urlkeys`

GET

See also:

*deliveryservices/{ID}/urlkeys*

Retrieves URL signing keys for a *Delivery Service*.

**Caution:** This method will return the *Delivery Service*'s **PRIVATE** URL signing keys! Be wary of using this endpoint and **NEVER** share the output with anyone who would be unable to see it on their own.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DS-SECURITY-KEY:READ, DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

Table 333: Request Path Parameters

Name	Description
xmlid	The ‘xml_id’ of the desired <i>Delivery Service</i>

## Response Structure

**key<N>** The private URL signing key for this *Delivery Service* as a base-64-encoded string, where <N> is the “generation” of the key e.g. the first key will always be named "key0". Up to 16 concurrent generations are retained at any time (<N> is always on the interval [0,15])

790: Response Example

```
{ "response": {
  "key9": "ZvVQNYpPVQWQV8tjQnUl6osm4y7xK4zD",
  "key6": "JhGdpw5X9o8TqHfgezCm0bqb9SQPASWL",
  "key8": "ySXdp1T8IeDEE1OCMftzZb9EIw_20wwq",
  "key0": "D4AYzJ1AE2nYisA9MxMtY03TPDCHji9C",
  "key3": "W90YHlGc_kYlYw5_I0LrkvV9J0zSIneI",
  "key12": "ZbtMb3mrKqfS8hnx9_xWBIP_OPWlUpzc",
  "key2": "0qgEoD07sUsugIQemZbwmMt0tNCwB1sf",
  "key4": "aFJ2Gb7atmxVB8uv7T9S6OaDml3ycpGf",
  "key1": "wnWNR1mCz104C7EFPtcqHd0xUMQyNFhA",
  "key11": "k6HMz1BH1x6htKkypRFfWQhAndQqe50e",
  "key10": "zYONfdD7fGYKj4kLvIj4U0918csuZ00d",
  "key15": "3360cGaIip_layZMc_0hI2teJbazxTQh",
  "key5": "SIwv3G0hWN7EE9wSwPFj18qE4M07sFxN",
  "key13": "SqQKBR6LqEOzp8AewZUCVtBcW_8YFc1g",
  "key14": "DtXsu8nsw04YhT0kNoKBhu2G3P9WRpQJ",
  "key7": "cmKoIIxXGAxUMdCsWvnGLoIMGmNiuT5I"
}}
```

## DELETE

### See also:

*deliveryservices/{{ID}}/urlkeys*

Deletes URL signing keys for a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:DELETE, DS-SECURITY-KEY:READ, DELIVERY-SERVICE:READ, SERVICE:UPDATE, DELIVERY-SERVICE:UPDATE

**Response Type** Object

## Request Structure

Table 334: Request Path Parameters

Name	Description
xmlid	The ‘xml_id’ of the desired <i>Delivery Service</i>

## Response Structure

791: Response Example

```
{
  "alerts": [{
    "level": "success",
    "text": "Successfully deleted URL Sig keys from Traffic Vault"
  }]
}
```

deliveryservices/xmlId/{{xml\_id}}/urlkeys/copyFromXmlId/  
{{copyFrom\_xml\_id}}

### POST

Allows a user to copy URL signing keys from a specified *Delivery Service* to another *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:READ, DS-SECURITY-KEY:CREATE, DELIVERY-SERVICE:READ, SERVICE:UPDATE, DELIVERY-SERVICE:UPDATE

**Response Type** Object (string)

## Request Structure

Table 335: Request Path Parameters

Name	Description
xml_id	The <i>xml_id</i> of the <i>Delivery Service</i> to which keys will be copied
copyFrom_xml_id	The <i>xml_id</i> of the <i>Delivery Service</i> from which keys will be copied

## Response Structure

792: Response Example

```
{
  "response": "Successfully copied and stored keys"
}
```

**deliveryservices/xmlId/{xml\_id}/urlkeys/generate**

**POST**

Generates URL signing keys for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:CREATE, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object (string)

## Request Structure

Table 336: Request Path Parameters

Name	Description
xml_id	The <i>xml_id</i> of the desired <i>Delivery Service</i>

## Response Structure

793: Response Example

```
{
  "response": "Successfully generated and stored keys"
}
```

### deliveryserviceserver

#### GET

Retrieve information about the assignment of servers to *Delivery Services*

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** SERVER:READ, DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

Table 337: Request Query Parameters

Name	Re- quired	De- fault	Description
cdn	no	None	Limit the results to delivery service servers for the given CDN name
page	no	0	The page number for use in pagination - 0 means “no pagination”
limit	no	20	Limits the results to a maximum of this number - if pagination is used, this defines the number of results per page
or- derby	no	“deliv- erySer- vice”	Choose the ordering of the results - the value must either be the name of one of the fields of the objects in the response array or be empty to skip ordering altogether

794: Request Example

```
GET /api/4.0/deliveryserviceserver?page=1&limit=2&orderby=lastUpdated.
→HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

<sup>1</sup> While no roles are required, this endpoint *does* respect tenancy permissions (pending [GitHub Issue #2978](#)).

## Response Structure

Unlike most API endpoints, this will return a JSON response body containing both a “response” object as well as other, top-level fields (besides the optional “alerts” field). For this reason, this section contains a “response” key, which normally is implicit.

See also:

### *Response Structure*

- limit** The maximum size of the response array, also indicative of the number of results per page using the pagination requested by the query parameters (if any) - this should be the same as the `limit` query parameter (if given)
- orderby** A string that names the field by which the elements of the response array are ordered - should be the same as the `orderby` request query parameter (if given)
- response** An array of objects, each of which represents a server’s *Delivery Service* assignment
  - deliveryService** The integral, unique identifier of the *Delivery Service* to which the server identified by `server` is assigned
  - lastUpdated** The date and time at which the server’s assignment to a *Delivery Service* was last updated
  - server** The integral, unique identifier of a server which is assigned to the *Delivery Service* identified by `deliveryService`
- size** The page number - if pagination was requested in the query parameters, else 0 to indicate no pagination - of the results represented by the response array. This is named “size” for legacy reasons

### 795: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: J7sK8PohQWyTpTrMjJrWdlJwPj+Zyep/
    ↳xutM25uVosL6cHgi30nXa6VMYOC5Y3vd9r5KLES8rTgR+qUQcZcJ/A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 01 Nov 2018 14:27:45 GMT
Content-Length: 129
```

(continues on next page)

(continued from previous page)

```
{ "orderBy": "lastUpdated",
  "response": [
    {
      "server": 8,
      "deliveryService": 1,
      "lastUpdated": "2018-11-01 14:10:38+00"
    }
  ],
  "size": 1,
  "limit": 2
}
```

## POST

Assign a set of one or more servers to a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>2</sup>

**Permissions Required** DELIVERY-SERVICE:READ, SERVER:READ,  
SERVER:UPDATE, DELIVERY-SERVICE:UPDATE

**Response Type** Object

## Request Structure

**dsId** The integral, unique identifier of the *Delivery Service* to which the servers identified in the `servers` array will be assigned

**replace** If true, any existing assignments for a server identified in the `servers` array will be overwritten by this request

**servers** An array of integral, unique identifiers for servers which are to be assigned to the *Delivery Service* identified by `deliveryService`

### 796: Request Example

```
POST /api/4.0/deliveryserviceserver HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

<sup>2</sup> Users with the “admin” or “operations” roles will be able to modify ALL server-to-Delivery-Service assignments, whereas all other users can only assign servers to the *Delivery Services* their Tenant has permissions to edit.

(continued from previous page)

```
Content-Length: 46
Content-Type: application/x-www-form-urlencoded

dsId=1&replace=true&servers=12
```

## Response Structure

- dsId** The integral, unique identifier of the *Delivery Service* to which the servers identified by the elements of the `servers` array have been assigned
- replace** If true, any existing assignments for a server identified in the `servers` array have been overwritten by this request
- servers** An array of integral, unique identifiers for servers which have been assigned to the *Delivery Service* identified by `deliveryService`

### 797: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳D+HhGhoxzaxvka9vZIStoa0ZUpX23nz7zZnMbpFHNRO3MawyEaSb3GVUHQyCv6sDgwhpZZjRggDmctGCw
X-Server-Name: traffic_ops_golang/
Date: Thu, 01 Nov 2018 14:12:49 GMT
Content-Length: 123

{ "alerts": [
  {
    "text": "server assignments complete",
    "level": "success"
  }
],
"response": {
  "dsId": 1,
  "replace": false,
  "servers": [ 12 ]
}}
```



deliveryserviceserver/{DSID}/{serverID}

## DELETE

Removes a *cache server* from a *Delivery Service*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE, SERVER:READ, SERVER:UPDATE

**Response Type** undefined

## Request Structure

Table 338: Request Path Parameters

Name	Required	Description
dsId	yes	An integral, unique identifier for a <i>Delivery Service</i>
serverID	yes	An integral, unique identifier for a server

**Note:** The server identified by `serverID` must be a *cache server*, or the assignment will fail.

## Response Structure

798: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: rGD2sOMHYF0sga1zuTytyLHCUkkc3ZwQRKvZ/
    ↳HuPz0bOP4WztKTOVXB4uhs3iJqBg9zRB2TucMx0NHN+3/yShQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 01 Nov 2018 14:24:34 GMT
```

(continues on next page)

<sup>1</sup> Users with the “admin” or “operations” roles will be able to delete *any Delivery Service*, whereas other users will only be able to delete *Delivery Services* that their tenant has permissions to delete.

(continued from previous page)

Content-Length: 80

```
{ "alerts": [
  {
    "text": "Server unlinked from delivery service.",
    "level": "success"
  }
]}
```

## divisions

### GET

Returns a JSON representation of all configured *Divisions*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** DIVISION:READ

**Response Type** Array

## Request Structure

Table 339: Request Query Parameters

Name	Description
id	Filter for <i>Divisions</i> having this integral, unique identifier
name	Filter for <i>Divisions</i> with this name
orderby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops's Custom Date/Time Format*

**name** The Division name

### 799: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳SLKi9RHa67sGoSz62IDcQsk7KZjTXKfonqMoCUFPXGcNUdhBssvUjc1G7KkWK8X1Ny16geMx2BN8Hm/
    ↳3dQ75GA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 19:44:03 GMT
Content-Length: 139

{ "response": [
  {
    "id": 1,
    "lastUpdated": "2018-11-29 18:38:28+00",
    "name": "Quebec"
  },
  {
    "id": 2,
    "lastUpdated": "2018-11-29 18:38:28+00",
    "name": "USA"
  }
]}
```

### POST

Creates a new Division.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DIVISION:CREATE, DIVISION:READ

**Response Type** Object

### Request Structure

**name** The name of the new Division

800: Request Example

```
POST /api/4.0/divisions HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 16
Content-Type: application/json

{"name": "test"}
```

### Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops's Custom Date/Time Format*

**name** The Division name

801: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +pJm4c30+JTaSXNt+LP+u240Ba/
  ↳SsvSSDOQ4rDc6hcyZ0FIL+iY/WWrMHhpLuLRGKGy88bM4YPCMaxGn3FZ9yQ=
```

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 19:52:06 GMT
Content-Length: 136

{ "alerts": [
  {
    "text": "division was created.",
    "level": "success"
  }
],
"response": {
  "id": 3,
  "lastUpdated": "2018-11-29 19:52:06+00",
  "name": "test"
}}
```

`divisions/{ID}`

**PUT**

Updates a specific Division

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DIVISION:UPDATE, DIVISION:READ

## Request Structure

Table 340: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the requested Division

**name** The new name of the Division

802: Request Example

```
PUT /api/4.0/divisions/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 17
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
{"name": "quest"}
```

## Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops's Custom Date/Time Format*

**name** The Division name

### 803: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: yBd8MzBR/Qbc/
    ↳xTs44WEIFRTrqeMKZwUe2ufpm6JH6frh1UjFmYRs3/
    ↳B7E5FTruFWRTuvEIlx5EpDmp3f9LjzA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 20:10:36 GMT
Content-Length: 137

{ "alerts": [
  {
    "text": "division was updated.",
    "level": "success"
  }
],
"response": {
  "id": 3,
  "lastUpdated": "2018-11-29 20:10:36+00",
  "name": "quest"
}}
```

## DELETE

Deletes a specific Division

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DIVISION:DELETE, DIVISION:READ

## Request Structure

Table 341: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the requested Division

804: Request Example

```
DELETE /api/4.0/divisions/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 2
Content-Type: application/json

{}
```

## Response Structure

**id** An integral, unique identifier for this Division

**lastUpdated** The date and time at which this Division was last modified, in *Traffic Ops’s Custom Date/Time Format*

**name** The Division name

805: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: yBd8MzBR/Qbc/
→xTs44WEIFRTrqeMKZwUe2ufpm6JH6frh1UjFmYRs3/
→B7E5FTruFWRTuvEIlx5EpDmp3f9LjzA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 29 Nov 2018 20:10:36 GMT
Content-Length: 83

{ "alerts": [
  {
    "text": "division was deleted.",
    "level": "success"
  }
]}

```

## federation\_resolvers

### GET

Retrieves *Federation* Resolvers.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** FEDERATION-RESOLVER:READ, TYPE:READ

**Response Type** Array



## Request Structure

Table 342: Request Query Parameters

Name	Re-quired	Description
id	no	Return only the Federation Resolver identified by this integral, unique identifier
ipAddress	no	Return only the Federation Resolver(s) that has/have this IP Address
type	no	Return only the Federation Resolvers of this <i>Type</i>
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 806: Request Example

```
GET /api/4.0/federation_resolvers?type=RESOLVE6 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.63.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**id** The integral, unique identifier of the resolver

**ipAddress** The IP address or CIDR-notation subnet of the resolver - may be IPv4 or IPv6

**lastUpdated** The date and time at which this resolver was last updated, in *Traffic Ops’s Custom Date/Time Format*

**type** The *Type* of the resolver

### 807: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→4TLkULAOAuap47H+hpwyf2lHjDbHbSNQHLmj7BCTHtps2CQxCuq7mwctbwqmPdmAjLOUXAIRsHmvSuAp4
X-Server-Name: traffic_ops_golang/
Date: Wed, 06 Nov 2019 00:03:56 GMT
Content-Length: 101

{ "response": [
  {
    "id": 1,
    "ipAddress": "::1/1",
    "lastUpdated": "2019-11-06 00:00:40+00",
    "type": "RESOLVE6"
  }
]}
```

## POST

Creates a new federation resolver.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION-RESOLVER:CREATE, TYPE:READ

**Response Type** Object

## Request Structure

**ipAddress** The IP address of the resolver - may be IPv4 or IPv6

**typeId** The integral, unique identifier of the *Type* of resolver being created

**Caution:** This field should only ever be an identifier for one of the *Types* “RESOLVE4” or “RESOLVE6”, but there is **no protection for this built into Traffic Ops** and therefore **any valid *Type* identifier will be silently accepted by Traffic Ops** and so care should be taken to ensure that these *Types* are properly identified. If any *Type* besides “RESOLVE4” or “RESOLVE6” is identified, the resulting resolver *will not work*.

**See also:**

*types* is the endpoint that can be used to determine the identifier for various *Types*

808: Request Example

```
POST /api/4.0/federation_resolvers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.63.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 36
Content-Type: application/json

{
  "ipAddress": "::1/1",
  "typeId": 37
}
```

## Response Structure

**id** The integral, unique identifier of the resolver

**ipAddress** The IP address or CIDR-notation subnet of the resolver - may be IPv4 or IPv6

**type** The *Type* of the resolver

**typeId** The integral, unique identifier of the *Type* of the resolver

809: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳e9D8JNrQb64xpuDwoBwbISSWUKDGCL2l37NuDXsXsPYof2EqmeHondD8NzxDSwWNJ8d9B9DXpZDbRUtgdl
X-Server-Name: traffic_ops_golang/
Date: Wed, 06 Nov 2019 00:00:40 GMT
Content-Length: 153

{ "alerts": [
```

(continues on next page)

(continued from previous page)

```
{
  "text": "Federation Resolver created [ IP = ::1/1 ] with id: 1
↪",
  "level": "success"
},
],
"response": {
  "id": 1,
  "ipAddress": "::1/1",
  "type": "RESOLVE6",
  "typeId": 37
}}
```

## DELETE

Deletes a federation resolver.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION-RESOLVER:DELETE, TYPE:READ

**Response Type** Object

## Request Structure

Table 343: Request Query Parameters

Name	Required	Description
id	yes	Integral, unique identifier for the federation resolver to be deleted

#### 810: Request Example

```
DELETE /api/4.0/federation_resolvers?id=4 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

### Response Structure

**id** The integral, unique identifier of the resolver

**ipAddress** The IP address or CIDR-notation subnet of the resolver - may be IPv4 or IPv6

**type** The *Type* of the resolver

#### 811: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; HttpOnly
Whole-Content-Sha512:
  ↳2v4LYQdRVhaFJVd86Iv1BWVYzNPSlzpQ222bUB7Zz+Ss8A48FNyHZjPlq5a+a4g9KAQCTUIytWnIQk+L1
X-Server-Name: traffic_ops_golang/
Date: Fri, 08 Nov 2019 23:19:01 GMT
Content-Length: 161

{ "alerts": [
  {
    "text": "Federation resolver deleted [ IP = 1.2.6.4/22 ] with
    ↳id: 4",
    "level": "success"
  }
],
"response": {
  "id": 4,
  "ipAddress": "1.2.6.4/22",
```

(continues on next page)

(continued from previous page)

```
"type": "RESOLVE6"
}}
```

## federations

### GET

Retrieves a list of *Federation* mappings (i.e. *Federation* Resolvers) for the current user.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** FEDERATION-RESOLVER:READ, DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

No parameters available.

## Response Structure

**deliveryService** The xml\_id that uniquely identifies the *Delivery Service* that uses the federation mappings in mappings

**mappings** An array of objects that represent the mapping of a *Federation*’s CNAME to one or more Resolvers

**cname** The actual CNAME used by the *Federation*

**resolve4** An array of IPv4 addresses (or subnets in CIDR notation) capable of resolving the *Federation*’s CNAME

**resolve6** An array of IPv6 addresses (or subnets in CIDR notation) capable of resolving the *Federation*’s CNAME

**ttl** The TTL of the CNAME in hours

### 812: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→d6Llm5qNc2sfgVH9IimW7hA4wvtBUq6EzUmpJf805kB0k6v2WysNgFEWK4hBXNdAYkr8hYuKPrwDy3tCx
X-Server-Name: traffic_ops_golang/
Date: Mon, 03 Dec 2018 17:19:13 GMT
Content-Length: 136

{ "response": [
  {
    "mappings": [
      {
        "ttl": 300,
        "cname": "blah.blah.",
        "resolve4": [
          "0.0.0.0/32"
        ],
        "resolve6": [
          ":/128"
        ]
      }
    ],
    "deliveryService": "demo1"
  }
]}
```

## POST

Allows a user to create *Federation* Resolvers for *Delivery Services*, providing the *Delivery Service* is within a CDN that has some associated *Federation*.

**Warning:** Confusingly, this method of this endpoint does **not** create a new *Federation*; to do that, the *cdns/{name}/federations* endpoint must be used. Furthermore, the *Federation* must properly be assigned to a *Delivery Service* using the *federations/{ID}/deliveryservices* and assigned to the user creating Resolvers using *federations/{ID}/users*.

### See also:

The *federations/{ID}/federation\_resolvers* endpoint duplicates this functionality.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** FEDERATION-RESOLVER:CREATE, FEDERATION-RESOLVER:READ, DELIVERY-SERVICE:READ

**Response Type** Object (string)

## Request Structure

The request payload is an array of objects that describe Delivery Service *Federation* Resolver mappings. Each object in the array must be in the following format.

**deliveryService** The *xml\_id* of the *Delivery Service* which will use the *Federation* Resolvers specified in mappings

**mappings** An object containing two arrays of IP addresses (or subnets in CIDR notation) to use as *Federation* Resolvers

**resolve4** An array of IPv4 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service's Federation*

**resolve6** An array of IPv6 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service's Federation*

### 813: Request Example

```
POST /api/4.0/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 118
Content-Type: application/json

[{"deliveryService": "demo1",
  "mappings": {
    "resolve4": ["127.0.0.1", "0.0.0.0/32"],
    "resolve6": [ "::1", "5efa::ff00/128"]
  }
}]
```



## Response Structure

### 814: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: B7TSU0YZPRPyi3mVy+CuxiXR5k/
    ↳d0s07w4i6kYzpWS+YL79juEfkusQfedayG/kMA809XbjkWRjcBAdxOVrdTQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 23 Oct 2019 22:28:02 GMT
Content-Length: 152

{ "alerts": [
  {
    "text": "admin successfully created federation resolvers.",
    "level": "success"
  }
],
"response": "admin successfully created federation resolvers."
}
```

## DELETE

Deletes **all** *Federation* Resolvers associated with the logged-in user's *Federations*.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** FEDERATION-RESOLVER:DELETE

**Response Type** Object (string)

## Request Structure

No parameters available

### 815: Request Example

```
DELETE /api/4.0/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 816: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳fd7P45mIiHuYqZZW6+8K+YjY1Pe504Aaw4J4Zp9AhrqLX72ERytTqWtAp1msutzNSRUdUSC72+odNPtpv:
X-Server-Name: traffic_ops_golang/
Date: Wed, 23 Oct 2019 23:34:53 GMT
Content-Length: 184

{ "alerts": [
  {
    "text": "admin successfully deleted all federation resolvers:
    ↳[ 8.8.8.8 ]",
    "level": "success"
  }
],
"response": "admin successfully deleted all federation resolvers: [ 8.
    ↳8.8.8 ]"
}
```

## PUT

Replaces **all** *Federations* Resolvers associated with a user's *Delivery Service*(s) with those defined inside the request payload.

**Auth. Required** Yes

**Roles Required** “admin”, “Federation”, “operations”, “Portal”, or “Steering”

**Permissions Required** FEDERATION-RESOLVER:DELETE, FEDERATION-RESOLVER:CREATE, DELIVERY-SERVICE:READ

**Response Type** Object (string)

## Request Structure

The request payload is an array of objects that describe Delivery Service *Federation* Resolver mappings. Each object in the array must be in the following format.

**deliveryService** The *xml\_id* of the *Delivery Service* which will use the *Federation* Resolvers specified in mappings

**mappings** An object containing two arrays of IP addresses (or subnets in CIDR notation) to use as *Federation* Resolvers

**resolve4** An array of IPv4 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service*'s *Federation*

**resolve6** An array of IPv6 addresses (or subnets in CIDR notation) that can resolve the *Delivery Service*'s *Federation*

### 817: Request Example

```
PUT /api/4.0/federations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 95
Content-Type: application/json

[{"mappings": {
  "resolve4": ["8.8.8.8"],
  "resolve6": []
},
"deliveryService": "demo1"
}]
```

## Response Structure

### 818: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dQ5AvQULhc254zQwgUpBl1/CHbLr/
→clKtkbs0Ju9f1BM4xIfbb03puFNN9zaEaZ1iz0lBvHFp/PgfUqisD3QHA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 23 Oct 2019 23:22:03 GMT
Content-Length: 258
Content-Type: application/json

{ "alerts": [
  {
    "text": "admin successfully deleted all federation resolvers:
→[ 8.8.8.8 ]",
    "level": "success"
  },
  {
    "text": "admin successfully created federation resolvers.",
    "level": "success"
  }
],
"response": "admin successfully created federation resolvers."
}
```

## federations/all

### GET

Retrieves a list of *Federation* mappings (also called *Federation* Resolvers) for the current user.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION-RESOLVER:READ, DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

No parameters available.

### 819: Request Example

```
GET /api/4.0/federations/all HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**deliveryService** The *xml\_id* of the delivery service.

**mappings** An array of objects that represent the mapping of a *Federation*'s CNAME to one or more Resolvers

**cname** The actual CNAME used by the *Federation*

**ttd** The TTL of the CNAME in hours

### 820: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Sun, 23 Feb 2020 21:38:06.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳UQBlGVPJytYMkv0V42EAIoJUnXjBTCXnOGp0berxte6TtnX63LTAKFfD2LejBVYXkKtnCdkBbs+SzhA0H
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 20:38:06 GMT
Content-Length: 138

{
  "response": [
    {
      "mappings": [
        {
          "ttl": 60,
```

(continues on next page)

(continued from previous page)

```

        "cname": "img1.mcdn.ciab.test."
      },
      {
        "ttl": 60,
        "cname": "img2.mycdn.ciab.test."
      }
    ],
    "deliveryService": "demo1"
  },
  {
    "mappings": [
      {
        "ttl": 60,
        "cname": "static.mycdn.ciab.test."
      }
    ],
    "deliveryService": "demo2"
  }
]
}

```

## federations/{{ID}}/deliveryservices

### GET

Retrieves *Delivery Services* assigned to a *Federation*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** FEDERATION:READ, DELIVERY-SERVICE:READ

**Response Type** Array

### Request Structure

Table 344: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation to be inspected

Table 345: Request Query Parameters

Name	Required	Description
dsID	no	Show only the <i>Delivery Service</i> identified by this integral, unique identifier
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

#### 821: Request Example

```
GET /api/4.0/federations/1/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cdn** The CDN to which this *Delivery Service* Belongs
- id** The integral, unique identifier for the *Delivery Service*
- type** The routing type used by this *Delivery Service*
- xmlId** The ‘xml\_id’ which uniquely identifies this *Delivery Service*

#### 822: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:44:13 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:44:13 GMT;
→path=/; HttpOnly
```

(continues on next page)

(continued from previous page)

```
vary: Accept-Encoding
whole-content-sha512: 7Y9Q/
→qHeXfbjJduvucRCR85wf4VRfyYhlK59sNRkzIJuwnsMhFcEfYfNqrvELwfexOum/
→VEX2f/1oa+I/edGfw==
content-length: 74

{ "response": [
  {
    "xmlId": "demo1",
    "cdn": "CDN-in-a-Box",
    "type": "HTTP",
    "id": 1
  }
]}
```

## POST

Assigns one or more *Delivery Services* to a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION:UPDATE, DELIVERY-  
SERVICE:UPDATE, FEDERATION:READ, DELIVERY-  
SERVICE:READ

**Response Type** Object

## Request Structure

Table 346: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation to be inspected

**dsIds** An array of integral, unique identifiers for *Delivery Services* which will be assigned to this federation

**replace** An optional boolean (default: `false`) which, if `true`, will cause any conflicting assignments already in place to be overridden by this request

---

**Note:** If `replace` is not given (and/or not `true`), then any conflicts with existing assignments will cause the entire operation to fail.

---



### 823: Request Example

```
POST /api/4.0/federations/1/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 32
Content-Type: application/json

{
  "dsIds": [1],
  "replace": true
}
```

## Response Structure

**dsIds** An array of integral, unique identifiers for *Delivery Services* which are now assigned to this federation

**replace** An optional boolean (default: false) which, if true, means any conflicting assignments already in place were overridden by this request

### 824: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
content-type: application/json
set-cookie: mojolicious=...; Path=/; HttpOnly
whole-content-sha512:
→rVd0nx8G3bRI8ub1zw6FTdmwQ7jer4zoqzOZf5tC1ckrR0HEIOH1Azdcmvv0FVE5I0omcHVnrYbzab7tU
x-server-name: traffic_ops_golang/
content-length: 137
date: Wed, 05 Dec 2018 00:34:06 GMT

{ "alerts": [
  {
    "text": "1 delivery service(s) were assigned to the federation.
→1",
    "level": "success"
  }
],
```

(continues on next page)

(continued from previous page)

```
"response": {
  "dsIds": [
    1
  ],
  "replace": true
}}
```

`federations/{{ID}}/deliveryservices/{{dsID}}`

DELETE

Removes a *Delivery Service* from a federation. A *Delivery Service* cannot be removed from a federation if it is the only *Delivery Service* assigned to said federation

- Auth. Required** Yes
- Roles Required** “admin”
- Permissions Required** FEDERATION:UPDATE, DELIVERY-SERVICE:UPDATE, FEDERATION:READ, DELIVERY-SERVICE:READ
- Response Type** undefined

Request Structure

Table 347: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the federation from which the <i>Delivery Service</i> identified by dsID will be removed
dsID	The integral, unique identifier of the <i>Delivery Service</i> which will be removed from the federation identified by ID

#### 825: Request Example

```
DELETE /api/4.0/federations/1/deliveryservices/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

#### 826: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 01:06:51 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 05:06:51 GMT;
    ↳path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: NqAZuZYlF1UW0aazbj/
    ↳j4gWX7ye0kGGakRRFEkK6ShxqXvCxEOdCTyu75qiLPN2wSgr3FGQnp2Sq345sE7In9g==
content-length: 98

{ "alerts": [
  {
    "level": "success",
    "text": "federation deliveryservice was deleted."
  }
]}
```

**federations/{{ID}}/federation\_resolvers**

**GET**

Retrieves federation resolvers assigned to a federation.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** FEDERATION:READ,  
RESOLVER:READ

FEDERATION-

**Response Type** Array

## Request Structure

Table 348: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation for which resolvers will be retrieved

### 827: Request Example

```
GET /api/4.0/federations/1/federation_resolvers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**id** The integral, unique identifier of this federation resolver

**ipAddress** The IP address of the federation resolver - may be IPv4 or IPv6

**type** The type of resolver - one of:

**RESOLVE4** This resolver is for IPv4 addresses (and **ipAddress** is IPv4)

**RESOLVE6** This resolver is for IPv6 addresses (and **ipAddress** is IPv6)

### 828: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
  ↳Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:49:50 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:49:50 GMT;
  ↳path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: csC18kE3YjiILHP1wmJg7V4h/
  ↳XWY8HUMKyPuZWnde2g7HJ4gTY51HfjCSqhyKvIJQ8R17uEqshF3Ey6xJMOX4A=
(continues on next page)
```

(continued from previous page)

```
content-length: 63

{ "response": [
  {
    "ipAddress": "0.0.0.0",
    "type": "RESOLVE4",
    "id": 1
  }
]}
```

POST

Assigns one or more resolvers to a federation.

- Auth. Required** Yes
- Roles Required** “admin”
- Permissions Required** FEDERATION:UPDATE, FEDERATION:READ, FEDERATION-RESOLVER:READ
- Response Type** Object

Request Structure

Table 349: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the federation for which resolvers will be retrieved

- fedResolverIds** An array of integral, unique identifiers for federation resolvers
- replace** An optional boolean (default: false) which, if true, will cause any conflicting assignments already in place to be overridden by this request

**Note:** If replace is not given (and/or not true), then any conflicts with existing assignments will cause the entire operation to fail.

829: Request Example

```
POST /api/4.0/federations/1/federation_resolvers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

(continued from previous page)

```
Content-Length: 41
Content-Type: application/json

{
  "fedResolverIds": [1],
  "replace": true
}
```

## Response Structure

**fedResolverIds** An array of integral, unique identifiers for federation resolvers

**replace** An optionally-present boolean (default: false) which, if true, any conflicting assignments already in place were overridden by this request

### 830: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:47:47 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:47:47 GMT;
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: +JDcRByS3H06pMg3Gzkvn0w7/
→v5oRu19e+RxyFIOKJKNH0kZILyQBS+PJpxDeCgwI19+0poW5dyHPPR9SwbNCA==
content-length: 148

{ "alerts": [
  {
    "level": "success",
    "text": "1 resolver(s) were assigned to the test.quest.
→federation"
  }
],
"response": {
  "replace": true,
  "fedResolverIds": [
    1
  ]
}
```

(continues on next page)

(continued from previous page)

```
}}
```

**federations/{{ID}}/users**

**GET**

Retrieves users assigned to a federation.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** FEDERATION:READ, USER:READ

**Response Type** Array

## Request Structure

Table 350: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the federation for which users will be retrieved

Table 351: Request Query Parameters

Name	Re-quired	Description
userId	no	Show only the user that has this integral, unique identifier
role	no	Show only the users that have this role
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

## Response Structure

**company** The company to which the user belongs

**email** The user's email address

**fullName** The user's full name

**id** An integral, unique identifier for the user

**role** The user's highest role

**username** The user's short "username"

### 831: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:31:34 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:31:34 GMT;
    ↳path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512: eQQoF2xlbK2I2oTja7zrt/
    ↳FlkLzCgwpU2zb2+rmIjHbHJ3MnmsSczSamIAAyTzs5gDaqcuUX1G35ZB8d7Bj82g==
content-length: 101

{ "response": [
  {
    "fullName": null,
    "email": null,
    "id": 2,
    "role": "admin",
    "company": null,
    "username": "admin"
  }
]}
```



## POST

Assigns one or more users to a federation.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** FEDERATION:UPDATE, USER:READ, FEDERATION:READ

**Response Type** Object

## Request Structure

**userIds** An array of integral, unique identifiers for users which will be assigned to this federation

**replace** An optional boolean (default: `false`) which, if `true`, will cause any conflicting assignments already in place to be overridden by this request

---

**Note:** If `replace` is not given (and/or not `true`), then any conflicts with existing assignments will cause the entire operation to fail.

---

### 832: Request Example

```
POST /api/4.0/federations/1/users HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 34
Content-Type: application/json

{
  "userIds": [2],
  "replace": true
}
```

## Response Structure

**userIds** An array of integral, unique identifiers for users which have been assigned to this federation

**replace** An optional boolean (default: false) which, if true, caused any conflicting assignments already in place to be overridden by this request

### 833: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
→Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 00:29:19 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 04:29:19 GMT;
→path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512:
→MvPmgOAs58aSOgVh+iEilflgOexbaexg+qE2IPrQZX0H4iSX4JvEys9adbGE9a9yaLj9uUMxg77N6ZyDh
content-length: 137

{ "alerts": [
  {
    "level": "success",
    "text": "1 user(s) were assigned to the test.quest. federation"
```

(continues on next page)

(continued from previous page)

```
    }
  ],
  "response": {
    "userIds": [
      2
    ],
    "replace": true
  }
}
```

`federations/{ID}/users/{userID}`

**DELETE**

Removes a user from a federation.

- Auth. Required** Yes
- Roles Required** “admin”
- Permissions Required** FEDERATION:UPDATE, FEDERATION:READ, USER:READ
- Response Type** undefined

**Request Structure**

Table 352: Request Path Parameters

Name	Description
ID	An integral, unique identifier for the federation from which the user identified by userID will be removed
userID	An integral, unique identifier for the user who will be removed from the federation identified by ID

#### 834: Request Structure

```
DELETE /api/4.0/federations/1/users/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

#### 835: Response Example

```
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-headers: Origin, X-Requested-With, Content-Type,
  ↳Accept
access-control-allow-methods: POST,GET,OPTIONS,PUT,DELETE
access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-type: application/json
date: Wed, 05 Dec 2018 01:14:04 GMT
X-Server-Name: traffic_ops_golang/
set-cookie: mojolicious=...; expires=Wed, 05 Dec 2018 05:14:04 GMT;
  ↳path=/; HttpOnly
vary: Accept-Encoding
whole-content-sha512:
  ↳xdF6l7jdd2t8au6lh4pFtDqYxTfehZke2aDBuytL7I74hK9KCT7ssLuYbfvD8ejdqqF3+jiBiFk7neQ8c
content-length: 93

{ "alerts": [
  {
    "level": "success",
    "text": "Removed user [ admin ] from federation [ foo.bar. ]"
  }
]}
```

**isos**

**POST**

Generates an ISO from the requested ISO source.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ISO:GENERATE, ISO:READ

**Response Type** undefined - ISO image as a streaming download

## Request Structure

**dhcp** A string that specifies whether the generated system image will use DHCP IP address leasing; one of:

**yes** DHCP will be used, and other network configuration keys need not be present in the request (and are ignored if they are)

**no** DHCP will not be used, and the desired network configuration **must** be specified manually in the request body

**disk** An optional string that names the block device (under /dev/) used for the boot media, e.g. “sda”

**domainName** The domain part of the system image’s Fully Qualified Domain Name (FQDN)

**hostName** The host name part of the system image’s FQDN

**interfaceMtu** A number that specifies the Maximum Transmission Unit (MTU) for the system image’s network interface card - the only valid values of which I’m aware are 1500 or 9000, and this should almost always just be 1500

**interfaceName** An optional string naming the network interface to be used by the generated system image e.g. “bond0”, “eth0”, etc. If the special name “bond0” is used, an LACP binding configuration will be created and included in the system image

**See also:**

[The Link Aggregation Wikipedia page.](#)

**ip6Address** An optional string containing the IPv6 address of the generated system image

**ip6Gateway** An optional string specifying the IPv6 address of the generated system image’s network gateway - this will be ignored if **ipGateway** is specified

**ipAddress** An optional<sup>1</sup> string containing the IP address of the generated system image

**ipGateway** An optional<sup>1</sup> string specifying the IP address of the generated system image’s network gateway

**ipNetmask** An optional<sup>1</sup> string specifying the subnet mask of the generated system image

**osversionDir** The name of the directory containing the ISO source

---

<sup>1</sup> This optional key is required if and only if **dhcp** is “no”.

See also:

*osversions*

**rootPass** The password used by the generated system image's root user

### 836: Request Example

```
POST /api/4.0/isos HTTP/1.1
Host: some.trafficops.host
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 334
Content-Type: application/json

{
  "osversionDir": "centos72",
  "hostName": "test",
  "domainName": "quest",
  "rootPass": "twelve",
  "dhcp": "no",
  "interfaceMtu": 1500,
  "ipAddress": "1.3.3.7",
  "ipNetmask": "255.255.255.255",
  "ipGateway": "8.0.0.8",
  "ip6Address": "1::3:3:7",
  "ip6Gateway": "8::8",
  "interfaceName": "eth0",
  "disk": "hda"
}
```

## Response Structure

ISO image as a streaming download.

### 837: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Connection: keep-alive
Content-Disposition: attachment; filename="test-centos72_centos72-
→netinstall.iso"
Content-Encoding: gzip
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/download
Date: Wed, 05 Feb 2020 21:59:15 GMT
Set-Cookie: mojolicious=...; Path=/; Expires=Wed, 05 Feb 2020 22:59:11.
→GMT; Max-Age=3600; HttpOnly
Transfer-Encoding: chunked
Whole-Content-sha512: sLSVQGrLCQ4hGQhv2reragQHWNi2aKMcz2c/
→HMAH45tLcZ1LenPyOzWRcRfHUNbV4PEEK0oiTfwE2HlA+WtRIQ==
X-Server-Name: traffic_ops_golang/
```

## jobs

### GET

Retrieve *Content Invalidation Jobs*.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** JOB:READ, DELIVERY-SERVICE:READ<sup>Page 1293, 1</sup>

**Response Type** Array

<sup>1</sup> When viewing *Content Invalidation Jobs*, only those jobs that operate on a *Delivery Service* visible to the requesting user's *Tenant* will be returned. Likewise, creating a new *Content Invalidation Jobs* requires that the target *Delivery Service* is modifiable by the requesting user's *Tenant*. However, when modifying or deleting an existing *Content Invalidation Jobs*, the operation can be completed if and only if the requesting user's *Tenant* is the same as the job's *Delivery Service*'s *Tenant* or a descendant thereof, **and** if the requesting user's *Tenant* is the same as the *Tenant* of the *user who initially created the job* or a descendant thereof.

## Request Structure

Table 353: Request Query Parameters

Name	Re-quired	Description
assetUrl	no	Return only <i>Content Invalidation Jobs</i> with this <i>Asset URL</i>
cdn	no	Return only <i>Content Invalidation Jobs</i> for <i>Delivery Services</i> within the CDN with this name
createdBy	no	Return only <i>Content Invalidation Jobs</i> that were created by the user with this username
delivery-Service	no	Return only <i>Content Invalidation Jobs</i> that operate on the <i>Delivery Service</i> with this <i>xml_id</i>
dsId	no	Return only <i>Content Invalidation Jobs</i> pending on the <i>Delivery Service</i> identified by this integral, unique identifier
id	no	Return only the single <i>Content Invalidation Job</i> with this <i>ID</i>
maxReval-Duration-Days	no	Return only <i>Content Invalidation Jobs</i> with a <i>Start Time</i> that is within the window defined by the <i>maxRevalDurationDays Parameter</i> in <i>The GLOBAL Profile</i>
userId	no	Return only <i>Content Invalidation Jobs</i> created by the user identified by this integral, unique identifier

### 838: Request Example

```
GET /api/4.0/jobs?id=1&dsId=1&userId=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.20.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**assetUrl** A regular expression - matching URLs will be operated upon according to keyword

**createdBy** The username of the user who initiated the job

**deliveryService** The *xml\_id* of the *Delivery Service* on which this job operates

**id** An integral, unique identifier for this job

**keyword** A keyword that represents the operation being performed by the job:

**PURGE** This job will prevent caching of URLs matching the *assetUrl* until it is removed (or its Time to Live expires)



**parameters** A string containing key/value pairs representing parameters associated with the job - currently only uses Time to Live e.g. "TTL:48h"

**startTime** The date and time at which the job began, in a non-standard format

#### 839: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 12 Nov 2021 19:30:36 GMT
Content-Length: 206

{ "response": [{
  "id": 1,
  "assetUrl": "http://origin.infra.ciab.test/.+",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "ttlHours": 72,
  "invalidationType": "REFETCH",
  "startTime": "2021-11-09T01:02:03Z"
}]}
```

## POST

Creates a new *Content Invalidation Jobs*.

**Caution:** Creating a *Content Invalidation Job* immediately triggers a CDN-wide revalidation update. In the case that the global *Parameter* `use_reval_pending` has a value of exactly "0", this will instead trigger a CDN-wide "Queue Updates". This means that *Content Invalidation Jobs* become active **immediately** at their `startTime` - unlike most other configuration changes they do not wait for a *Snapshot* or a "Queue Updates". Furthermore, if the global *Parameter* `use_reval_pending` is "0", this will cause all pending configuration changes to propagate to all *cache servers* in the CDN. Take care when using this endpoint.

**Auth. Required** Yes

**Roles Required** "operations" or "admin"<sup>Page 1293, 1</sup>

**Permissions Required** JOB:CREATE, JOB:READ, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE<sup>Page 1293, 1</sup>

**Response Type** Object

## Request Structure

**deliveryService** The *Delivery Service*

**invalidationType** The *Invalidation Type*

**regex** The *Regular Expression*

**startTime** The *Start Time*

**ttl** The *TTL*

### 840: Request Example

```
POST /api/4.0/jobs HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Transfer-Encoding: chunked
Content-Type: application/json

{
  "deliveryService": "demo1",
  "invalidationType": "REFRESH",
  "regex": "/.+",
  "startTime": "2021-11-09T01:02:03Z",
  "ttlHours": 72
}
```

## Response Structure

**assetUrl** The *Asset URL*

**createdBy** The *Created By*

**deliveryService** The *Delivery Service*

**id** The *ID*.

**invalidationType** The *Invalidation Type*

**ttlHours** The *TTL*

**startTime** The *Start Time*

### 841: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
Location: https://localhost:6443/api/4.0/jobs?id=1
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Mon, 08 Nov 2021 15:44:46 GMT
Content-Length: 265

{
  "alerts": [
    {
      "text": "Invalidation (REFRESH) request created for http://
→origin.infra.ciab.test/., start:2021-11-09 01:02:03 +0000 UTC end.
→2021-11-12 01:02:03 +0000 UTC",
      "level": "success"
    }
  ],
  "response": {
    "id": 1,
    "assetUrl": "http://origin.infra.ciab.test/.",
    "createdBy": "admin",
    "deliveryService": "demo1",
    "ttlHours": 72,
    "invalidationType": "REFRESH",
    "startTime": "2021-11-09T01:02:03Z"
  }
}
```

## PUT

Replaces an existing *Content Invalidation Job* with a new one provided in the request. This method of editing a *Content Invalidation Job* does not prevent the requesting user from changing fields that normally only have one value. Use with care.

**Caution:** Modifying a *Content Invalidation Job* immediately triggers a CDN-wide revalidation update. In the case that the global *Parameter* `use_reval_pending` has a value of exactly `"0"`, this will instead trigger a CDN-wide “Queue Updates”. This means that *Content Invalidation Jobs* become active **immediately** at their `startTime` - unlike most other configuration changes they do not wait for a *Snapshot* or a “Queue Updates”. Furthermore, if the global *Parameter* `use_reval_pending` is `"0"`, this will cause all pending configuration changes to propagate to all *cache servers* in the CDN. Take care when using this endpoint.

**Auth. Required** Yes

**Roles Required** “operations” or “admin”<sup>Page 1293, 1</sup>

**Permissions Required** JOB:UPDATE, DELIVERY-SERVICE:UPDATE,  
JOB:READ, DELIVERY-SERVICE:READ<sup>Page 1293, 1</sup>

**Response Type** Object

## Request Structure

Table 354: Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Content Invalidation Job</i> being modified

**assetUrl** The *Asset URL* - the scheme and authority parts of the regular expression cannot be changed

**createdBy** The *Created By*<sup>2</sup>

**deliveryService** The *Delivery Service*<sup>?</sup>

**id** The *ID*<sup>?</sup>

**invalidationType** The *Invalidation Type*

**ttlHours** The *TTL*

**startTime** The *Start Time*

### 842: Request Example

```
PUT /api/4.0/jobs?id=1 HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 191

{
  "assetUrl": "http://origin.infra.ciab.test/.+",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "id": 1,
  "invalidationType": "REFETCH",
  "startTime": "2021-11-09T01:02:03Z",
  "ttlHours": 72
}
```

<sup>2</sup> This field must exist, but it must *not* be different than the same field of the existing job (i.e. as seen in a *GET* response). That is, this cannot be changed.

## Response Structure

**assetUrl** The *Asset URL*  
**createdBy** The *Created By*  
**deliveryService** The *Delivery Service*  
**id** The *ID*  
**invalidationType** The *Invalidation Type*  
**ttlHours** The *TTL*  
**startTime** The *Start Time*

### 843: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Mon, 08 Nov 2021 16:43:35 GMT
Content-Length: 266

{ "alerts": [
  {
    "text": "Invalidation request created for http://origin.infra.
    ciab.test/., start: 2021-11-09 01:02:03 +0000 UTC end: 2021-11-12
    01:02:03 +0000 UTC invalidation type: REFETCH",
    "level": "success"
  }
],
"response": {
  "assetUrl": "http://origin.infra.ciab.test/.",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "id": 1,
  "invalidationType": "REFETCH",
  "startTime": "2021-11-09T01:02:03Z",
  "ttlHours": 72
}}
```

## DELETE

Deletes a *Content Invalidation Job*.

**Tip:** *Content Invalidation Jobs* that have passed their TTL are not automatically deleted - for record-keeping purposes - so use this to clean up old jobs that are no longer useful.

**Caution:** Deleting a *Content Invalidation Job* immediately triggers a CDN-wide revalidation update. In the case that the global *Parameter* `use_reval_pending` has a value of exactly "0", this will instead trigger a CDN-wide "Queue Updates". This means that *Content Invalidation Jobs* become active **immediately** at their `startTime` - unlike most other configuration changes they do not wait for a *Snapshot* or a "Queue Updates". Furthermore, if the global *Parameter* `use_reval_pending` is "0", this will cause all pending configuration changes to propagate to all *cache servers* in the CDN. Take care when using this endpoint.

**Auth. Required** Yes

**Roles Required** "operations" or "admin"<sup>Page 1293, 1</sup>

**Permissions Required** JOB:DELETE, JOB:READ, DELIVERY-SERVICE:UPDATE, DELIVERY-SERVICE:READ<sup>Page 1293, 1</sup>

**Response Type** Object

## Request Structure

Table 355: Query Parameters

Name	Re-quired	Description
id	yes	The integral, unique identifier of the <i>Content Invalidation Job</i> being modified

### 844: Request Example

```
DELETE /api/4.0/jobs?id=1 HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

**assetUrl** The *Asset URL* of the deleted *Content Invalidation Job*

**createdBy** The *Created By* of the deleted *Content Invalidation Job*

**deliveryService** The *Delivery Service* of the deleted *Content Invalidation Job*

**id** The *ID*. of the deleted *Content Invalidation Job*

**invalidationType** The *Invalidation Type* of the deleted *Content Invalidation Job*

**ttlHours** The *TTL* of the deleted *Content Invalidation Job*

**startTime** The *Start Time* of the deleted *Content Invalidation Job*

### 845: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Mon, 08 Nov 2021 16:54:32 GMT
Content-Length: 230

{ "alerts": [
  {
    "text": "Content invalidation job was deleted",
    "level": "success"
  }
],
"response": {
  "assetUrl": "http://origin.infra.ciab.test/.+",
  "createdBy": "admin",
  "deliveryService": "demo1",
  "id": 1,
  "invalidationType": "REFETCH",
  "startTime": "2021-11-09T01:02:03Z",
  "ttlHours": 72
}}
```

## letsencrypt/autorenew

### POST

Generates an SSL certificate and private key using Let's Encrypt for a *Delivery Service*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:CREATE, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object

### Request Structure

No parameters available

### Response Structure



#### 846: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "alerts": [
  {
    "text": "This endpoint is deprecated, please use letsencrypt/
↳autorenew instead",
    "level": "warning"
  },
  {
    "text": "Beginning async call to renew certificates. This may
↳take a few minutes.",
    "level": "success"
  }
]}
```

### letsencrypt/dnsrecords

#### GET

Gets DNS challenge records for Let’s Encrypt DNS challenge for a specified FQDN.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DS-SECURITY-KEY:READ

**Response Type** Array

### Request Structure

Table 356: Request Query Parameters

Name	Required	Description
fqdn	no	Return only DNS challenge records for the specified FQDN

#### 847: Request Example

```
GET /api/4.0/letsencrypt/dnsrecord?fqdn=_acme-challenge.demo1.example.
↳com. HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

**fqdn** The FQDN for the TXT record location to complete the DNS challenge

**record** The record provided by Let's Encrypt to complete the DNS challenge

848: Response Example

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": [
  {
    "fqdn": "_acme-challenge.demo1.example.com.",
    "record": "testRecord"
  }
]}
```

### logs

---

**Note:** This endpoint's responses will contain a cookie (`last_seen_log`) that is used by *logs/newcount* to determine the time of last access. Be sure your client uses cookies properly if you intend to use *logs/newcount* in concert with this endpoint!

---

### GET

Fetches a list of changes that have been made to the Traffic Control system

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** LOG:READ

**Response Type** Array

## Request Structure

Table 357: Request Query Parameters

Name	Re-quired	Description
days	no	An integer number of days of change logs to return
limit	no	The number of records to which to limit the response, by default there is no limit applied
off-set	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.
user-name	no	A name to which to limit the response too

New in version ATCv6: The username, page, offset query parameters were added to this in endpoint across across all API versions in ATC version 6.0.0.

### 849: Request Example

```
GET /api/4.0/logs?days=1&limit=2&username=admin HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**id** Integral, unique identifier for the Log entry

**lastUpdated** Date and time at which the change was made, in *Traffic Ops’s Custom Date/Time Format*

**level** Log categories for each entry, e.g. ‘UICHANGE’, ‘OPER’, ‘APICHANGE’

**message** Log detail about what occurred

**ticketNum** Optional field to cross reference with any bug tracking systems

**user** Name of the user who made the change

### 850: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept
```

(continues on next page)

(continued from previous page)

```

Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 15:11:38 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: last_seen_log="2018-11-15% 15:11:38"; path=/; Max-
    ↳Age=604800
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
    ↳40dV+aZaZ3b6F30y6YHVbV3H2a3ekZrdoxICupwaxQnj62pwYfb7YCM7Qhe30AItmB77Tbg9INy27ymaz3
Content-Length: 357

{ "response": [
  {
    "ticketNum": null,
    "level": "APICHANGE",
    "lastUpdated": "2018-11-14 21:40:06.493975+00",
    "user": "admin",
    "id": 444,
    "message": "User [ test ] unlinked from deliveryservice [ 1 |
    ↳demo1 ]."
  },
  {
    "ticketNum": null,
    "level": "APICHANGE",
    "lastUpdated": "2018-11-14 21:37:30.707571+00",
    "user": "admin",
    "id": 443,
    "message": "1 delivery services were assigned to test"
  }
],
  "summary": {
    "count": 2
  }
}

```

## Summary Fields

The summary object returned by this method of this endpoint uses only the count *standard property*.

### logs/newcount

#### GET

Gets the number of new changes made to the Traffic Control system - “new” being defined as the last time the client requested either *logs*

**Note:** This endpoint’s functionality is implemented by the *logs* endpoint’s response setting cookies for the client to use when requesting `_this_` endpoint. Take care that your client respects cookies!

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** LOG:READ

**Response Type** Object

## Request Structure

No parameters available

## Response Structure

**newLogcount** The integer number of new changes

851: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 15 Nov 2018 15:17:35 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
➔GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Vary: Accept-Encoding
Whole-Content-Sha512: Ugdqe8GXKS0ExphwbDX/
↪Gli+2vBpubttbpfYMBJaCP7adox3MzmVRi2RxTDL5kwPewrcL1C088zGITskh0sc9g==
Content-Length: 30

{ "response": {
  "newLogcount": 4
}}
```

## OC/CI/configuration

### PUT

Triggers an asynchronous task to update the configuration for the uCDN by adding the request to a queue to be reviewed later. This returns a 202 Accepted status and an endpoint to be used for status updates.

**Note:** Users with the ICN:UCDN-OVERRIDE permission will need to provide a “ucdn” query parameter to bypass the need for uCDN information in the JWT and allow them to view all CDNI information.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDNI:UPDATE

**Response Type** Object

## Request Structure

This requires authorization using a JWT provided by the dCDN to identify the uCDN. This token must include the following claims:

Table 358: Required JWT claims

Name	Description
iss	Issuer claim as a string key for the uCDN
aud	Audience claim as a string key for the dCDN
exp	Expiration claim as the expiration date as a Unix epoch timestamp (in seconds)

**type** A string of the type of metadata to follow. See [RFC 8006](#) for possible values. Only a selection of these are supported.

**host** A string of the domain that the requested updates will change.

**metadata** An array of generic metadata objects that conform to [RFC 8006](#).

**generic-metadata-type** A string of the type of metadata to follow conforming to **RFC 8006**.

**generic-metadata-value** An array of generic metadata value objects conforming to **RFC 8006** and SVA specifications.

852: Example /OC/CI/configuration Request

```
PUT /api/4.0/oc/ci/configuration HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 181
Content-Type: application/json

{
  "type": "MI.HostMetadata",
  "host": "example.com",
  "metadata": [
    {
      "generic-metadata-type": "MI.RequestedCapacityLimits",
      "generic-metadata-value": {
        "requested-limits": [
          {
            "limit-type": "egress",
            "limit-value": 20000,
            "footprints": [
              {
                "footprint-type": "ipv4cidr",
                "footprint-value": [
                  "127.0.0.1",
                  "127.0.0.2"
                ]
              }
            ]
          }
        ]
      }
    }
  ]
}
```

## Response Structure

853: Response Example

```
HTTP/1.1 202 Accepted
Content-Type: application/json

{ "alerts": [
  {
    "text": "CDNi configuration update request received. Status_
    ↪updates can be found here: /api/4.0/async_status/1",
    "level": "success"
  }
]}
```

OC/CI/configuration/{{host}}

### PUT

Triggers an asynchronous task to update the configuration for the uCDN and the specified host by adding the request to a queue to be reviewed later. This returns a 202 Accepted status and an endpoint to be used for status updates.

**Note:** Users with the ICDN:UCDN-OVERRIDE permission will need to provide a “ucdn” query parameter to bypass the need for uCDN information in the JWT and allow them to view all CDNI information.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDNI:UPDATE

**Response Type** Object

## Request Structure

This requires authorization using a JWT provided by the dCDN to identify the uCDN. This token must include the following claims:

Table 359: Required JWT claims

Name	Description
iss	Issuer claim as a string key for the uCDN
aud	Audience claim as a string key for the dCDN
exp	Expiration claim as the expiration date as a Unix epoch timestamp (in seconds)



Table 360: Request Path Parameters

Name	Description
host	The text identifier for the host domain to be updated with the new configuration.

**type** A string of the type of metadata to follow. See [RFC 8006](#) for possible values. Only a selection of these are supported.

**host-metadata** An array of generic metadata objects that conform to [RFC 8006](#).

**generic-metadata-type** A string of the type of metadata to follow conforming to [RFC 8006](#).

**generic-metadata-value** An array of generic metadata value objects conforming to [RFC 8006](#) and SVA specifications.

#### 854: Example /OC/CI/configuration Request

```
PUT /api/4.0/oc/ci/configuration/example.com HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 181
Content-Type: application/json

{
  "type": "MI.HostMetadata",
  "host-metadata": [
    {
      "generic-metadata-type": "MI.RequestedCapacityLimits",
      "generic-metadata-value": {
        "requested-limits": [
          {
            "limit-type": "egress",
            "limit-value": 20000,
            "footprints": [
              {
                "footprint-type": "ipv4cidr",
                "footprint-value": [
                  "127.0.0.1",
                  "127.0.0.2"
                ]
              }
            ]
          }
        ]
      }
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
]
}
```

## Response Structure

855: Response Example

```
HTTP/1.1 202 Accepted
Content-Type: application/json

{ "alerts": [
  {
    "text": "CDNi configuration update request received. Status_
    ↪updates can be found here: /api/4.0/async_status/1",
    "level": "success"
  }
]}
```

OC/CI/configuration/request/{{id}}/{{approved}}

### PUT

Triggers an asynchronous task to update the configuration for the uCDN and the specified host by adding the request to a queue to be reviewed later. This returns a 202 Accepted status and an endpoint to be used for status updates.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** CDNI-ADMIN:READ, CDNI-ADMIN:UPDATE

**Response Type** Object

## Request Structure

Table 361: Request Path Parameters

Name	Description
id	The integral identifier for the configuration update request to be approved or denied.
ap-proved	A boolean for whether to approve a configuration change request or not.

## Response Structure

### 856: Response Example For Approved Change

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": "Successfully updated configuration." }
```

### 857: Response Example For Denied Change

```
HTTP/1.1 200 OK
Content-Type: application/json

{ "response": "Successfully denied configuration update request." }
```

## OC/CI/configuration/requests

### GET

Returns the requested updates for CDNI configurations. An optional `id` parameter will return only information for a specific request.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** CDNI-ADMIN:READ

**Response Type** Array

## Request Structure

Table 362: Request Query Parameters

Name	Re-quired	Description
id	no	Return only the configuration requests identified by this integral, unique identifier

## Response Structure

- id** An integral, unique identifier for the requested configuration updates.
- ucdn** The name of the uCDN to which the requested changes apply.
- data** An array of generic FCI base objects.
- host** The domain to which the requested changes apply.
- requestType** A string of the type of configuration update request.
- asyncStatusId** An integral, unique identifier for the associated asynchronous status.
- generic-metadata-type** A string of the type of metadata to follow conforming to [RFC 8006](#).
- generic-metadata-value** An array of generic metadata value objects conforming to [RFC 8006](#) and SVA specifications.
- footprints** An array of footprints impacted by this generic base object.

---

**Note:** These are meant to be generic and therefore there is not much information in these documents. For further information please see [RFC 8006](#), [RFC 8007](#), [RFC 8008](#), and the SVA documents titled *Footprint and Capabilities Interface: Open Caching API*, *Open Caching API Implementation Guidelines*, *Configuration Interface: Part 1 Specification - Overview & Architecture*, *Configuration Interface: Part 2 Specification – CDNi Metadata Model Extensions*, and *Configuration Interface: Part 3 Specification – Publishing Layer APIs*.

---

858: Example /OC/CI/configuration/requests Response

```
{
  "response": [
    {
      "id": 1,
      "ucdn": "ucdn1",
      "data": [
        {
          "generic-metadata-type": "MI.
↪RequestedCapacityLimits",
          "generic-metadata-value": {
            "requested-limits": [
              {
                "limit-type": "egress",
                "limit-value": 232323,
                "footprints": [
                  {
                    "footprint-type": "ipv4cidr",
                    "footprint-value": [
```

(continues on next page)

(continued from previous page)

```

        "127.0.0.1",
        "127.0.0.2"
      ],
    },
    {
      "footprint-type": "countrycode",
      "footprint-value": [
        "us"
      ]
    }
  ]
}
],
"host": "example.com",
"requestType": "hostConfigUpdate",
"asyncStatusId": 0
}
]
}

```

## OC/FCI/advertisement

### GET

Returns the complete footprint and capabilities information structure the dCDN wants to expose to a given uCDN.

**Note:** Users with the ICDN:UCDN-OVERRIDE permission will need to provide a “ucdn” query parameter to bypass the need for uCDN information in the JWT and allow them to view all CDNI information.

**Auth. Required** No

**Roles Required** “admin” or “operations”

**Permissions Required** CDNI:READ

**Response Type** Array

## Request Structure

This requires authorization using a JWT provided by the dCDN to identify the uCDN. This token must include the following claims:

Table 363: Required JWT claims

Name	Description
iss	Issuer claim as a string key for the uCDN
aud	Audience claim as a string key for the dCDN
exp	Expiration claim as the expiration date as a Unix epoch timestamp (in seconds)

## Response Structure

**capabilities** An array of generic FCI base objects.

**capability-type** A string of the type of base object.

**capability-value** An array of the value for the base object.

**footprints** An array of footprints impacted by this generic base object.

---

**Note:** These are meant to be generic and therefore there is not much information in these documents. For further information please see [RFC 8006](#), [RFC 8007](#), [RFC 8008](#), and the SVA documents titled *Footprint and Capabilities Interface: Open Caching API*, *Open Caching API Implementation Guidelines*, *Configuration Interface: Part 1 Specification - Overview & Architecture*, *Configuration Interface: Part 2 Specification – CDNi Metadata Model Extensions*, and *Configuration Interface: Part 3 Specification – Publishing Layer APIs*.

---

859: Example /OC/FCI/advertisement Response

```
{
  "capabilities": [
    {
      "capability-type": "FCI.CapacityLimits",
      "capability-value": [
        {
          "limits": [
            {
              "id": "host_limit_requests_requests",
              "scope": {
                "type": "testScope",
                "value": [
                  "test.com"
                ]
              }
            }
          ]
        }
      ]
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```

    },
    "limit-type": "requests",
    "maximum-hard": 20,
    "maximum-soft": 15,
    "telemetry-source": {
      "id": "request_metrics",
      "metric": "requests"
    }
  },
  {
    "id": "total_limit_egress_capacity",
    "limit-type": "egress",
    "maximum-hard": 202020,
    "maximum-soft": 500,
    "telemetry-source": {
      "id": "capacity_metrics",
      "metric": "capacity"
    }
  }
]
},
"footprints": [
  {
    "footprint-type": "countrycode",
    "footprint-value": [
      "us"
    ]
  }
]
},
{
  "capability-type": "FCI.Telemetry",
  "capability-value": {
    "sources": [
      {
        "id": "capacity_metrics",
        "type": "generic",
        "metrics": [
          {
            "name": "capacity",
            "time-granularity": 0,
            "data-percentile": 50,
            "latency": 0
          }
        ]
      }
    ]
  }
},

```

(continues on next page)

(continued from previous page)

```

        "configuration": {
            "url": "example.com/telemetry1"
        }
    ],
    },
    "footprints": [
        {
            "footprint-type": "countrycode",
            "footprint-value": [
                "us"
            ]
        }
    ]
}

```

## origins

### GET

Gets all requested *Origins*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** ORIGIN:READ, DELIVERY-SERVICE:READ

**Response Type** Array



## Request Structure

Table 364: Request Query Parameters

Name	Re- quired	Description
cacheGroup	no	Return only <i>Origins</i> within the <i>Cache Group</i> that has this <i>ID</i>
co- ordi- nate	no	Return only <i>Origins</i> located at the geographic coordinates identified by this integral, unique identifier
de- liv- ery- ser- vice	no	Return only <i>Origins</i> that belong to the <i>Delivery Service</i> identified by this integral, unique identifier
id	no	Return only the <i>Origin</i> that has this integral, unique identifier
name	no	Return only <i>Origins</i> by this name
pro- fileId	no	Return only <i>Origins</i> which use the <i>Profile</i> that has this <i>ID</i>
pri- mary	no	If <code>true</code> , return only <i>Origins</i> which are the the primary <i>Origin</i> of the <i>Delivery Service</i> to which they belong - if <code>false</code> return only <i>Origins</i> which are <i>not</i> the primary <i>Origin</i> of the <i>Delivery Service</i> to which they belong
ten- ant	no	Return only <i>Origins</i> belonging to the tenant identified by this integral, unique identifier
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
off- set	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
page	no	Return the $n^{\text{th}}$ page of results, where “ $n$ ” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

**Note:** Several fields of origin definitions which are filterable by Query Parameters are allowed to be `null`. `null` values in these fields will be filtered *out* appropriately by such Query Parameters, but do note that `null` is not a valid value accepted by any of these Query Parameters, and attempting to pass it will result in an error.

### 860: Request Example

```
GET /api/4.0/origins?name=demo1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cachegroup** A string that is the *name of the Cache Group* to which the *Origin* belongs
- cachegroupId** An integer that is the *ID of the Cache Group* to which the *Origin* belongs
- coordinate** The name of a coordinate pair that defines the origin's geographic location
- coordinateId** An integral, unique identifier for the coordinate pair that defines the *Origin's* geographic location
- deliveryService** A string that is the *xml\_id* of the *Delivery Service* to which the *Origin* belongs
- deliveryServiceId** An integral, unique identifier for the *Delivery Service* to which the *Origin* belongs
- fqdn** The FQDN of the *Origin*
- id** An integral, unique identifier for this *Origin*
- ip6Address** The IPv6 address of the *Origin*
- ipAddress** The IPv4 address of the *Origin*
- isPrimary** A boolean value which, when `true` specifies this *Origin* as the 'primary' *Origin* served by `deliveryService`
- lastUpdated** The date and time at which this *Origin* was last modified
- name** The name of the *Origin*
- port** The TCP port on which the *Origin* listens
- profile** The *Name* of the *Profile* used by this *Origin*
- profileId** The *ID* of the *Profile* used by this *Origin*
- protocol** The protocol used by this origin - will be one of 'http' or 'https'
- tenant** The name of the *Tenant* that owns this *Origin*
- tenantId** An integral, unique identifier for the *Tenant* that owns this *Origin*

### 861: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳sm8DpvdvrfdSVLtmXTdfjsZbTlbc+pI40Gy0aj00XIURTPfFXuv/
    ↳4LgHb6A3r92iybBRHvFrH6qdB2g97U2sBg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 15:43:41 GMT
Content-Length: 376
```

```
{ "response": [
  {
    "cachegroup": null,
    "cachegroupId": null,
    "coordinate": null,
    "coordinateId": null,
    "deliveryService": "demo1",
    "deliveryServiceId": 1,
    "fqdn": "origin.infra.ciab.test",
    "id": 1,
    "ip6Address": null,
    "ipAddress": null,
    "isPrimary": true,
    "lastUpdated": "2018-12-10 19:11:32+00",
    "name": "demo1",
    "port": null,
    "profile": null,
    "profileId": null,
    "protocol": "http",
    "tenant": "root",
    "tenantId": 1
  }
]}
```

## POST

Creates a new origin definition.

**Warning:** At the time of this writing it is possible to create and/or modify origin definitions assigned to STEERING and CLIENT\_STEERING *Delivery Services* - despite that an origin has no meaning in those contexts. In these cases, the API responses may give incorrect output - see [GitHub Issue #3107](#) for details and updates.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ORIGIN:CREATE, ORIGIN:READ, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object

## Request Structure

**cachegroupId** An optional, integer which, if present, should be the *Cache Group ID* that identifies a *Cache Group* to which the new *Origin* shall belong

**coordinateId** An optional, integral, unique identifier of a coordinate pair that shall define the *Origin*’s geographic location

**deliveryServiceId** The integral, unique identifier of the *Delivery Service* to which the new *Origin* shall belong

**fqdn** The FQDN of the *Origin*

**ip6Address** An optional string containing the IPv6 address of the *Origin*

**ipAddress** An optional string containing the IPv4 address of the *Origin*

**isPrimary** An optional boolean which, if `true` will set this *Origin* as the ‘primary’ *Origin* served by the *Delivery Service* identified by `deliveryServiceID`

**Note:** Though not specifying this field in this request will leave it as `null` in the output, Traffic Ops will silently coerce that to its default value: `false`.

**name** A human-friendly name of the *Origin*

**port** An optional port number on which the *Origin* listens for incoming TCP connections

**profileId** An optional *ID* of a *Profile* that shall be used by this *Origin*

**protocol** The protocol used by the origin - must be one of ‘http’ or ‘https’

**tenantId** An optional<sup>1</sup>, integral, unique identifier for the *Tenant* which shall own the new *Origin*

#### 862: Request Example

```
POST /api/4.0/origins HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 114
Content-Type: application/json

{
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "name": "example",
  "port": 80,
  "protocol": "http",
  "tenantId": 1
}
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the *Origin* belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the *Origin* belongs

**coordinate** The name of a coordinate pair that defines the origin's geographic location

**coordinateId** An integral, unique identifier for the coordinate pair that defines the *Origin*'s geographic location

**deliveryService** The 'xml\_id' of the *Delivery Service* to which the *Origin* belongs

**deliveryServiceId** An integral, unique identifier for the *Delivery Service* to which the *Origin* belongs

**fqdn** The FQDN of the *Origin*

**id** An integral, unique identifier for this *Origin*

**ip6Address** The IPv6 address of the *Origin*

**ipAddress** The IPv4 address of the *Origin*

**isPrimary** A boolean value which, when true specifies this *Origin* as the 'primary' *Origin* served by *deliveryService*

<sup>1</sup> The tenantId field is required if and only if tenancy is enabled within Traffic Ops.

**lastUpdated** The date and time at which this *Origin* was last modified

**name** The name of the *Origin*

**port** The TCP port on which the *Origin* listens

**profile** The *Name* of the *Profile* used by this *Origin*

**profileId** The *ID* the *Profile* used by this *Origin*

**protocol** The protocol used by this origin - will be one of 'http' or 'https'

**tenant** The name of the *Tenant* that owns this *Origin*

**tenantId** An integral, unique identifier for the *Tenant* that owns this *Origin*

### 863: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳z4gp0MaqYu+gSR0RhKT2e0bVBuVDVx1rdteRaN5kRL9uJ3hNzUCi4dSKIt0rgNgOEDt6x/
    ↳iTyrnVhr/TSHYtmA==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 15:14:27 GMT
Content-Length: 418

{ "alerts": [
  {
    "text": "origin was created.",
    "level": "success"
  }
],
"response": {
  "cachegroup": null,
  "cachegroupId": null,
  "coordinate": null,
  "coordinateId": null,
  "deliveryService": null,
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "id": 2,
  "ip6Address": null,
  "ipAddress": null,
```

(continues on next page)

(continued from previous page)

```
"isPrimary": null,
"lastUpdated": "2018-12-11 15:14:27+00",
"name": "example",
"port": 80,
"profile": null,
"profileId": null,
"protocol": "http",
"tenant": null,
"tenantId": 1
}}
```

## PUT

Updates an *Origin* definition.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ORIGIN:UPDATE, ORIGIN:READ, DELIVERY-SERVICE:READ, DELIVERY-SERVICE:UPDATE

**Response Type** Object

## Request Structure

Table 365: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the <i>Origin</i> definition being edited

**cachegroupId** An optional, integer which, if present, should be the *Cache Group ID* that identifies a *Cache Group* to which the new *Origin* shall belong

**coordinateId** An optional, integral, unique identifier of a coordinate pair that shall define the *Origin*’s geographic location

**deliveryServiceId** The integral, unique identifier of the *Delivery Service* to which the *Origin* shall belong

**fqdn** The FQDN of the *Origin*

**ip6Address** An optional string containing the IPv6 address of the *Origin*

**ipAddress** An optional string containing the IPv4 address of the *Origin*

**isPrimary** An optional boolean which, if true will set this *Origin* as the ‘primary’ origin served by the *Delivery Service* identified by `deliveryServiceID`

**name** A human-friendly name of the *Origin*

**port** An optional port number on which the *Origin* listens for incoming TCP connections

**profileId** An optional *ID* of the *Profile* that shall be used by this *Origin*

**protocol** The protocol used by the *Origin* - must be one of ‘http’ or ‘https’

**tenantId** An optional<sup>Page 1325, 1</sup>, integral, unique identifier for the *Tenant* which shall own the new *Origin*

#### 864: Request Example

```
PUT /api/4.0/origins?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 135
Content-Type: application/json

{
  "deliveryServiceId": 2,
  "fqdn": "example.com",
  "isprimary": true,
  "name": "example",
  "port": 443,
  "protocol": "https",
  "tenantId": 1
}
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the *Origin* belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the *Origin* belongs

**coordinate** The name of a coordinate pair that defines the origin’s geographic location

**coordinateId** An integral, unique identifier for the coordinate pair that defines the *Origin*’s geographic location

**deliveryService** The ‘xml\_id’ of the *Delivery Service* to which the *Origin* belongs

**deliveryServiceId** An integral, unique identifier for the *Delivery Service* to which the *Origin* belongs

**fqdn** The FQDN of the *Origin*

**id** An integral, unique identifier for this *Origin*



**ip6Address** The IPv6 address of the *Origin*

**ipAddress** The IPv4 address of the *Origin*

**isPrimary** A boolean value which, when `true` specifies this *Origin* as the ‘primary’ *Origin* served by `deliveryService`

**lastUpdated** The date and time at which this *Origin* was last modified

**name** The name of the *Origin*

**port** The TCP port on which the *Origin* listens

**profile** The *Name* of the *Profile* used by this *Origin*

**profileId** The *ID* the *Profile* used by this *Origin*

**protocol** The protocol used by this origin - will be one of ‘http’ or ‘https’

**tenant** The name of the *Tenant* that owns this *Origin*

**tenantId** An integral, unique identifier for the *Tenant* that owns this *Origin*

#### 865: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Zx7j0a7UAQxRtDenYodvGQSoooPj4m0yY0AIEUpbdelmYMiNdPYtW82BCmMesFXkmP74nV4HbTUyDHVMu.
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 15:40:53 GMT
Content-Length: 420

{ "alerts": [
  {
    "text": "origin was updated.",
    "level": "success"
  }
],
"response": {
  "cachegroup": null,
  "cachegroupId": null,
  "coordinate": null,
  "coordinateId": null,
  "deliveryService": null,
  "deliveryServiceId": 2,
```

(continues on next page)

(continued from previous page)

```

    "fqdn": "example.com",
    "id": 2,
    "ip6Address": null,
    "ipAddress": null,
    "isPrimary": true,
    "lastUpdated": "2018-12-11 15:40:53+00",
    "name": "example",
    "port": 443,
    "profile": null,
    "profileId": null,
    "protocol": "https",
    "tenant": null,
    "tenantId": 1
  }}

```

## DELETE

Deletes an *Origin* definition.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** ORIGIN:DELETE, DELIVERY-SERVICE:UPDATE

**Response Type** undefined

## Request Structure

Table 366: Request Query Parameters

Name	Required	Description
id	yes	The integral, unique identifier of the <i>Origin</i> definition being deleted

#### 866: Request Example

```
DELETE /api/4.0/origins?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 867: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: fLaY4/
    ↳nh0yR38xq5weBKYg02+aQV6Z1ZroOq9UqUCHLMMrH1NMyhOHx+EphPq7JxkjmGY04WCt6VvDyjGWcgfQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 17:04:14 GMT
Content-Length: 61

{ "alerts": [
  {
    "text": "origin was deleted.",
    "level": "success"
  }
]}
```

### osversions

See also:

*Generate ISO*

### GET

Gets all available OS versions for ISO generation, as well as the name of the directory where the “kickstarter” files are found.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** ISO:READ

**Response Type** Object

### Request Structure

No parameters available.

### Response Structure

This endpoint has no constant keys in its response. Instead, each key in the response object is the name of an OS, and the value is a string that names the directory where the ISO source can be found. These directories sit under `/var/www/files/` on the Traffic Ops host machine by default, or at the location defined by the `kickstart.files.location` *Parameter* of the Traffic Ops server’s *Profile*, if it is defined.

#### 868: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳RxbRY2DZ+1Y0dTzzUETEZ3wtLBiD2BwXMVuaZjhe4a4cwgCZKRBWxZ6Qy5YYujFe1+UBiTG4sML/
  ↳Amn27F4AVg==
X-Server-Name: traffic_ops_golang/
Date: Fri, 30 Nov 2018 19:14:36 GMT
```

(continues on next page)

(continued from previous page)

```
Content-Length: 38

{ "response": {
  "CentOS 7.2": "centos72"
}}
```

## Configuration File

The data returned from the endpoint comes directly from a configuration file. By default, the file is located at `/var/www/files/osversions.json`. The **directory** of the file can be changed by creating a specific *Parameter* named `kickstart.files.location` in configuration file `mkisofs`.

The format of the file is a JSON object as described in *Response Structure*.

869: Example osversions.json file

```
{
  "CentOS 7.2": "centos72"
}
```

The legacy Perl Traffic Ops used a Perl configuration file located by default at `/var/www/files/osversions.cfg`. A Perl script is provided to convert the legacy configuration file to the new JSON format. The script is located within the Traffic Control repository at `traffic_ops/app/bin/osversions-convert.pl`.

870: Example usage of conversion script

```
./osversions-convert.pl < /var/www/files/osversions.cfg > /var/www/
↪files/osversions.json
```

## parameterprofile

### POST

Create one or more *Parameter/Profile* assignments.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:UPDATE, PROFILE:READ, PARAMETER:READ

**Response Type** Object

## Request Structure

- paramId** The *ID* of the *Parameter* to be assigned to the *Profiles* identified within the `profileIds` array
- profileIds** An array of *Profile IDs* to which the *Parameter* identified by `paramId` shall be assigned
- replace** An optional boolean (default: `false`) which, if `true`, will cause any conflicting *Profile/Parameter* assignments to be overridden.

### 871: Request Example

```
POST /api/4.0/parameterprofile HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 38
Content-Type: application/json

{
  "paramId": 4,
  "profileIds": [18]
}
```

## Response Structure

- paramId** The *ID* of the *Parameter* which has been assigned to the *Profiles* identified within the `profileIds` array
- profileIds** An array of *Profile IDs* to which the *Parameter* identified by `paramId` has been assigned
- replace** An optional boolean (default: `false`) which, if `true`, caused any conflicting *Profile/Parameter* assignments to be overridden.

### 872: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳N2ahnhEnfZ0UqnjylN6Vu3HaOZk340YuiuyiqkhTbk0pENp+kwBPYu4Z/
  ↳sqBALoCfXSQaWlJzaeXw4uOD5heWw==
```

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:18:23 GMT
Content-Length: 147

{ "alerts": [
  {
    "text": "2 parameters were assigned to the 18 profile",
    "level": "success"
  }
],
"response": {
  "profileId": 18,
  "paramIds": [
    2,
    3
  ],
  "replace": false
}}
```

## parameters

### GET

Gets all *Parameters* configured in Traffic Ops

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PARAMETER:READ

**Response Type** Array

## Request Structure

Table 367: Request Query Parameters

Name	Re-quired	Description
con-fig-File	no	Filter <i>Parameters</i> by <i>Config File</i>
id	no	Filters <i>Parameters</i> by <i>ID</i>
name	no	Filter <i>Parameters</i> by <i>Name</i>
value	no	Filter <i>Parameters</i> by <i>Value</i>
or-derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
off-set	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

873: Request Example

```
GET /api/4.0/parameters?configFile=records.config&name=location HTTP/1.
→1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**configFile** The *Parameter’s Config File*

**id** The *Parameter’s ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops’s Custom Date/Time Format*

**name** *Name* of the *Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter’s Value*



## 874: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
    ↳jcBFmFZM7CsrsIWtfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
    ↳DbBwhMAOJoeqDImoDlrLnrVjQGO4AooA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 18:23:39 GMT
Content-Length: 212

{ "response": [
  {
    "configFile": "records.config",
    "id": 29,
    "lastUpdated": "2018-12-05 17:51:02+00",
    "name": "location",
    "profiles": [
      "ATS_EDGE_TIER_CACHE",
      "ATS_MID_TIER_CACHE"
    ],
    "secure": false,
    "value": "/etc/trafficserver/"
  }
]}
```

## POST

Creates one or more new *Parameters*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PARAMETER:CREATE, PARAMETER:READ

**Response Type** Array

## Request Structure

The request body may be in one of two formats, a single *Parameter* object or an array of *Parameter* objects. Each *Parameter* object shall have the following keys:

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 875: Request Example - Single Object Format

```
POST /api/4.0/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 84
Content-Type: application/json

{
  "name": "test",
  "value": "quest",
  "configFile": "records.config",
  "secure": false
}
```

### 876: Request Example - Array Format

```
POST /api/4.0/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 180
Content-Type: application/json

[
  {
    "name": "test",
    "value": "quest",
    "configFile": "records.config",
    "secure": false
  },
  {
    "name": "foo",
    "value": "bar",
```

(continues on next page)

(continued from previous page)

```
"configFile": "records.config",
"secure": false
}]
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name* of the *Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 877: Response Example - Single Object Format

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳eQrl48zWids0kDpfCYmmtYMpegjnFxf0VvlBYxxLSfp7P7p6oWX4uiC+/
    ↳Cfh2X9i3G+MQ36eH95gukJqOB0GbQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:18:21 GMT
Content-Length: 212

{ "alerts": [
  {
    "text": "param was created.",
    "level": "success"
  }
],
"response": {
  "configFile": "records.config",
  "id": 124,
```

(continues on next page)

(continued from previous page)

```
"lastUpdated": "2018-12-05 19:18:21+00",
"name": "test",
"profiles": null,
"secure": false,
"value": "quest"
}}
```

parameters/{ID}

PUT

Replaces a *Parameter*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PARAMETER:UPDATE, PARAMETER:READ

**Response Type** Object

## Request Structure

Table 368: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Parameter</i> which will be deleted

**configFile** The *Parameter*’s *Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter*’s *Value*

### 878: Request Example

```
PUT /api/4.0/parameters/124 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 81
Content-Type: application/json

{
  "name": "foo",
```

(continues on next page)

(continued from previous page)

```
"value": "bar",
"configFile": "records.config",
"secure": false
}
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name of the Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 879: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: DMxS2gKceFVKRtezON/
→vsnrC+zI8onASSHaGv5i3wwvUvyt9KEe72gxQd6ZgVcSq3K8ZpkH6g3UI/WtEfdp5vA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 20:21:07 GMT
Content-Length: 209

{ "alerts": [
  {
    "text": "param was updated.",
    "level": "success"
  }
],
"response": {
  "configFile": "records.config",
  "id": 125,
```

(continues on next page)

(continued from previous page)

```
"lastUpdated": "2018-12-05 20:21:07+00",
"name": "foo",
"profiles": null,
"secure": false,
"value": "bar"
}}
```

## DELETE

Deletes the specified *Parameter*. If, however, the *Parameter* is associated with one or more *Profiles*, deletion will fail.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PARAMETER:DELETE, PARAMETER:READ

**Response TYPe** undefined

## Request Structure

Table 369: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Parameter</i> which will be deleted

### 880: Request Example

```
DELETE /api/4.0/parameters/124 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 881: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: hJjQq2Seg7sqWt+jKgp6gwRxUtoVU34PFoc9wEaweXdaIBTn/
→BscoUuyw2/n+V8GZPqpeQcihZE50/0oQhdtHw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 19:20:30 GMT
Content-Length: 60

{ "alerts": [
  {
    "text": "param was deleted.",
    "level": "success"
  }
]}

```

## phys\_locations

### GET

Retrieves *Physical Locations*

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PHYSICAL-LOCATION:READ

**Response Type** Array

## Request Structure

Table 370: Request Query Parameters

Name	Description
id	Filter by integral, unique identifier
re-gion	Filter by integral, unique identifier of containing <i>Region</i>
name	Filter by name
or-derby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

882: Request Example

```
GET /api/4.0/phys_locations?name=CDN_in_a_Box HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- address** The physical location’s street address
- city** The name of the city in which the physical location lies
- comments** Any and all human-readable comments
- email** The email address of the physical location’s poc
- id** An integral, unique identifier for the physical location
- lastUpdated** The date and time at which the physical location was last updated, in *Traffic Ops’s Custom Date/Time Format*
- name** The name of the physical location
- phone** A phone number where the the physical location’s poc might be reached
- poc** The name of a “point of contact” for the physical location
- region** The name of the region within which the physical location lies



**regionId** An integral, unique identifier for the region within which the physical location lies

**shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

#### 883: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳0g4b3W1AwXytCnBo8TReQQij2v9oHA17MG9KuwMig5V4sFcMM5qP8dgPsFTunFr00DPI20c7BpUbZsvJts
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 22:19:52 GMT
Content-Length: 275

{ "response": [
  {
    "address": "1600 Pennsylvania Avenue NW",
    "city": "Washington",
    "comments": "",
    "email": "",
    "id": 2,
    "lastUpdated": "2018-12-05 17:50:58+00",
    "name": "CDN_in_a_Box",
    "phone": "",
    "poc": "",
    "regionId": 1,
    "region": "Washington, D.C",
    "shortName": "ciab",
    "state": "DC",
    "zip": "20500"
  }
]}
```

### POST

Creates a new physical location

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PHYSICAL-LOCATION:CREATE, PHYSICAL-LOCATION:READ

**Response Type** Object

### Request Structure

**address** The physical location’s street address

**city** The name of the city in which the physical location lies

**comments** An optional string for containing any and all human-readable comments

**email** An optional string containing email address of the physical location’s poc

**name** An optional name of the physical location

**phone** An optional string containing the phone number where the the physical location’s poc might be reached

**poc** The name of a “point of contact” for the physical location

**region** An optional string naming the region that contains this physical location<sup>1</sup>

**regionId** An integral, unique identifier for the region within which the physical location lies<sup>1</sup>

**shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

#### 884: Request Example

```
POST /api/4.0/phys_locations HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

---

<sup>1</sup> The only “region” key that actually matters in the request body is `regionId`; `region` is not validated and has no effect - particularly not the effect of re-naming the region - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the region. Note that if `region` is not present in the request body it will be `null` in the response, but again further requests will show the true region name.

(continued from previous page)

```
Content-Length: 326
Content-Type: application/json

{
  "address": "Buckingham Palace",
  "city": "London",
  "comments": "Buckingham Palace",
  "email": "steve.kingstone@royal.gsx.gov.uk",
  "name": "Great_Britain",
  "phone": "0-843-816-6276",
  "poc": "Her Majesty The Queen Elizabeth Alexandra Mary Windsor II",
  "regionId": 3,
  "shortName": "uk",
  "state": "Westminster",
  "zip": "SW1A 1AA"
}
```

## Response Structure

- address** The physical location's street address
- city** The name of the city in which the physical location lies
- comments** Any and all human-readable comments
- email** The email address of the physical location's poc
- id** An integral, unique identifier for the physical location
- lastUpdated** The date and time at which the physical location was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of the physical location
- phone** A phone number where the the physical location's poc might be reached
- poc** The name of a "point of contact" for the physical location
- region** The name of the region within which the physical location lies
- regionId** An integral, unique identifier for the region within which the physical location lies
- shortName** An abbreviation of the name
- state** An abbreviation of the name of the state or province within which this physical location lies
- zip** The zip code of the physical location

## 885: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: GZ/
    ↳BC+AgGpOQNfd9oiZy19jtsD8MP0deyi7PVdz+9YSiLYP44gmn5K+Xi1yS0l59yjHf70+C1loVQPSlIeP9
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 00:14:47 GMT
Content-Length: 443

{ "alerts": [
  {
    "text": "physLocation was created.",
    "level": "success"
  }
],
"response": {
  "address": "Buckingham Palace",
  "city": "London",
  "comments": "Buckingham Palace",
  "email": "steve.kingstone@royal.gsx.gov.uk",
  "id": 3,
  "lastUpdated": "2018-12-06 00:14:47+00",
  "name": "Great_Britain",
  "phone": "0-843-816-6276",
  "poc": "Her Majesty The Queen Elizabeth Alexandra Mary Windsor II",
  "regionId": 3,
  "region": null,
  "shortName": "uk",
  "state": "Westminster",
  "zip": "SW1A 1AA"
}}
```

`phys_locations/{{ID}}`

**PUT**

Updates a *Physical Location*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PHYSICAL-LOCATION:UPDATE, PHYSICAL-LOCATION:READ

**Response Type** Array

## Request Structure

Table 371: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Physical Location</i> being modified

**address** The physical location’s street address

**city** The name of the city in which the physical location lies

**comments** An optional string for containing any and all human-readable comments

**email** An optional string containing email address of the physical location’s poc

**name** An optional name of the physical location

**phone** An optional string containing the phone number where the the physical location’s poc might be reached

**poc** The name of a “point of contact” for the physical location

**region** An optional string naming the region that contains this physical location<sup>1</sup>

**regionId** An integral, unique identifier for the region within which the physical location lies<sup>1</sup>

**shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

<sup>1</sup> The only “region” key that actually matters in the request body is `regionId`; `region` is not validated and has no effect - particularly not the effect of re-naming the region - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the region. Note that if `region` is not present in the request body it will be `null` in the response, but again further requests will show the true region name.

### 886: Request Structure

```
PUT /api/4.0/phys_locations/2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 268
Content-Type: application/json

{
  "address": "1600 Pennsylvania Avenue NW",
  "city": "Washington",
  "comments": "The White House",
  "email": "the@white.house",
  "name": "CDN_in_a_Box",
  "phone": "1-202-456-1414",
  "poc": "Donald J. Trump",
  "regionId": 2,
  "shortName": "ciab",
  "state": "DC",
  "zip": "20500"
}
```

### Response Structure

- address** The physical location's street address
- city** The name of the city in which the physical location lies
- comments** Any and all human-readable comments
- email** The email address of the physical location's poc
- id** An integral, unique identifier for the physical location
- lastUpdated** The date and time at which the physical location was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of the physical location
- phone** A phone number where the the physical location's poc might be reached
- poc** The name of a "point of contact" for the physical location
- region** The name of the region within which the physical location lies
- regionId** An integral, unique identifier for the region within which the physical location lies
- shortName** An abbreviation of the name

**state** An abbreviation of the name of the state or province within which this physical location lies

**zip** The zip code of the physical location

#### 887: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳qnMe60qxjSU8H1njlh00HWNr20YnVlOCufqCTdMBcdC1322jk2ICFQsQQ3Xu0OR0WSb7h70HCfXqDC1/
    ↳jA1xjA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 23:39:17 GMT
Content-Length: 385

{ "alerts": [
  {
    "text": "physLocation was updated.",
    "level": "success"
  }
],
"response": {
  "address": "1600 Pennsylvania Avenue NW",
  "city": "Washington",
  "comments": "The White House",
  "email": "the@white.house",
  "id": 2,
  "lastUpdated": "2018-12-05 23:39:17+00",
  "name": "CDN_in_a_Box",
  "phone": "1-202-456-1414",
  "poc": "Donald J. Trump",
  "regionId": 2,
  "region": null,
  "shortName": "ciab",
  "state": "DC",
  "zip": "20500"
}}
```

## DELETE

Deletes a *Physical Location*

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PHYSICAL-LOCATION:DELETE, PHYSICAL-LOCATION:READ

**Response Type** undefined

## Request Structure

Table 372: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the <i>Physical Location</i> being deleted

### 888: Request Example

```
DELETE /api/4.0/phys_locations/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 889: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: KeW/tEmICwpCGC8F0YMTqHdeR9J6W6Z3w/
→U+H0SbeCGyaEheCIhIsWlngT3dyfH1tiu8UyzaPB6QrJyXdybBkw==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 00:28:48 GMT
Content-Length: 67
```

(continues on next page)



(continued from previous page)

```
{ "alerts": [
  {
    "text": "physLocation was deleted.",
    "level": "success"
  }
]}
```

## ping

Checks whether Traffic Ops is online.

## GET

**Auth. Required** No

**Response Type** undefined

## Request Structure

No parameters available.

### 890: Request Example

```
GET /api/4.0/ping HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**ping** Returns an object containing only the "ping" property, which always has the value "pong".

### 891: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST, GET, OPTIONS, PUT, DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Whole-Content-Sha512: 0HqcLcYHCB4AFYGFzcAsP2h+PCMlYxk/
↪TqMajcy3fWCzY730Tv32k5trUaJLeSBbRx2FUi7z/sTAuzikdg0E4g==
X-Server-Name: traffic_ops_golang/
Date: Sun, 23 Feb 2020 20:52:01 GMT
Content-Length: 40
Content-Type: application/x-gzip

{
  "ping": "pong"
}
```

### plugins

**See also:**

*Go Plugins*

### GET

Retrieves the list of configured enabled Traffic Ops plugins.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PLUGIN:READ

**Response Type** Array

### Request Structure

### 892: Request Example

```
GET /api/4.0/plugins HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**description** A short description of the plugin.

**name** The name of the plugin.

**version** A (hopefully) semantic version number describing the version of the plugin.

### 893: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Tue, 11 Dec 2018 20:51:48 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: n73jg9XR4V5Cwqq56Rf3wuIi99k3mM5u2NAjcZ/
→gQBu8jvAFymDlnZqKeJ+wTl1lvjIsHpXCOVXV7+5UGakLgA==
Transfer-Encoding: chunked

{ "response": [
  {
    "name": "helloworld",
    "version": "1.0.0",
    "description": "configuration plugin to run at startup"
  }
]
}]}
```

### profileparameter

See also:

*profileparameters*.

### POST

Create one or more *Profile/Parameter* assignments.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:READ,      PARAMETER:READ,      PRO-  
FILE:UPDATE

**Response Type** Object

### Request Structure

**profileId** The *ID* of the *Profile* to which the *Parameters* identified within the *parameterIds* array will be assigned

**paramIds** An array of *Parameter IDs* which shall be assigned to the *Profile* identified by *profileId*

**replace** An optional boolean (default: false) which, if true, will cause any conflicting *Profile/Parameter* assignments to be overridden.

894: Request Example

```
POST /api/4.0/profileparameter HTTP/1.1
```

```
Host: trafficops.infra.ciab.test
```

```
User-Agent: curl/7.47.0
```

```
Accept: */*
```

```
Cookie: mojolicious=...
```

```
Content-Length: 38
```

```
Content-Type: application/json
```

```
{
  "profileId": 18,
  "paramIds": [2, 3]
}
```

## Response Structure

- profileId** The *ID* of the *Profile* to which the *Parameters* identified within the *parameterIds* array are assigned
- paramIds** An array of *Parameter IDs* which have been assigned to the *Profile* identified by *profileId*
- replace** An optional boolean (default: false) which, if true, indicates that any conflicting *Profile/Parameter* assignments have been overridden.

### 895: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳N2ahnhEnfZ0UqnjylN6Vu3HaOZk340YuiuyiqkhTbk0pENp+kwBPYu4Z/
    ↳sqBAloCfXSQaWlJzaeXw4uOD5heWw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:18:23 GMT
Content-Length: 147

{ "alerts": [
  {
    "text": "2 parameters were assigned to the 18 profile",
    "level": "success"
  }
],
"response": {
  "profileId": 18,
  "paramIds": [
    2,
    3
  ],
  "replace": false
}}
```

## profileparameters

### GET

Retrieves all *Parameter/Profile* assignments.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PROFILE:READ, PARAMETER:READ

**Response Type** Array

### Request Structure

Table 373: Request Query Parameters

Name	Re- quired	Description
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### Response Structure

**lastUpdated** The date and time at which this *Profile/Parameter* association was last modified, in *Traffic Ops’s Custom Date/Time Format*

**parameter** The *ID* of a *Parameter* assigned to profile

**profile** The *Name* of the *Profile* to which the *Parameter* identified by parameter is assigned

#### 896: Response Structure

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→+bnMkRgdx4bJoGGlr3mZl539obj3aQAP8e65FAXgywdRAUfXZCFM6VNDn7wScXBmvF2SFXo9F+MhuSwrtI
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:09:13 GMT
Transfer-Encoding: chunked

{ "response": [
  {
    "lastUpdated": "2018-12-05 17:50:49+00",
    "profile": "GLOBAL",
    "parameter": 4
  },
  {
    "lastUpdated": "2018-12-05 17:50:49+00",
    "profile": "GLOBAL",
    "parameter": 5
  }
]}
```

**Note:** The response example for this endpoint has been truncated to only the first two elements of the resulting array, as the output was hundreds of lines long.

## POST

Associate a *Parameter* to a *Profile*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Response Type** Object

**Permissions Required** PROFILE:READ,      PARAMETER:READ,      PRO-  
FILE:UPDATE

## Request Structure

This endpoint accepts two formats for the request payload:

**Single Object Format** For assigning a single *Parameter* to a single *Profile*

**Array Format** For making multiple assignments of *Parameters* to *Profiles* simultaneously

### Single Object Format

**parameterId** The *ID* of a *Parameter* to assign to some *Profile*

**profileId** The *ID* of the *Profile* to which the *Parameter* identified by parameterId will be assigned

897: Request Example - Single Object Format

```
POST /api/4.0/profileparameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 36
Content-Type: application/json

{
  "profileId": 18,
  "parameterId": 1
}
```

### Array Format

**parameterId** The *ID* of a *Parameter* to assign to some *Profile*

**profileId** The *ID* of the *Profile* to which the *Parameter* identified by parameterId will be assigned

898: Request Example - Array Format

```
POST /api/4.0/profileparameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 88
Content-Type: application/json

[ {
```

(continues on next page)



(continued from previous page)

```
"profileId": 18,
"parameterId": 2
},
{
  "profileId": 18,
  "parameterId": 3
}]
```

## Response Structure

**lastUpdated** The date and time at which the *Profile/Parameter* assignment was last modified, in *Traffic Ops's Custom Date/Time Format*

**parameter** *Name* of the *Parameter* which is assigned to profile

**parameterId** The *ID* of the assigned *Parameter*

**profile** *Name* of the *Profile* to which the *Parameter* is assigned

**profileId** The *ID* of the *Profile* to which the *Parameter* identified by parameterId is assigned

### 899: Response Example - Single Object Format

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→eDmIwlzX44fZdxLRPHMNa8aoGAK5fQv9Y70A2eeQHfEkliU4evwcsQ4WeHcH0l3/
→wPTGlpyC0gwLo8LQqUxWQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 13:50:11 GMT
Content-Length: 166

{ "alerts": [
  {
    "text": "profileParameter was created.",
    "level": "success"
  }
],
"response": {
```

(continues on next page)

(continued from previous page)

```
"lastUpdated": null,
"profile": null,
"profileId": 18,
"parameter": null,
"parameterId": 1
}}
```

profileparameters/{profileID}/{parameterID}

## DELETE

Deletes a *Profile/Parameter* association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:UPDATE, PROFILE:READ, PARAMETER:READ

**Response Type** undefined

## Request Structure

Table 374: Request Path Parameters

Name	Description
profileID	The <i>ID</i> of the <i>Profile</i> from which a <i>Parameter</i> shall be removed
parameterID	The <i>ID</i> of the <i>Parameter</i> which shall be removed from the <i>Profile</i> identified by profileID

#### 900: Request Example

```
DELETE /api/4.0/profileparameters/18/129 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 901: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳JQuBqHyT9MnNwO9NSIDVQhkRtXdeAJc95W1pF2dwQeoBFmf0Y8knXm3/0/
    ↳rbJDEoUC7DhUQN1aoYIsqqmz4qQQ==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 15:00:15 GMT
Content-Length: 71

{ "alerts": [
  {
    "text": "profileParameter was deleted.",
    "level": "success"
  }
]}
```

## profiles

### GET

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PROFILE:READ

**Response Type** Array

## Request Structure

Table 375: Request Query Parameters

Name	Re-quired	Description
cdn	no	Used to filter <i>Profiles</i> by the integral, unique identifier of the CDN to which they belong
id	no	Filters <i>Profiles</i> by <i>ID</i>
name	no	Filters <i>Profiles</i> by <i>Name</i>
param	no	Used to filter <i>Profiles</i> by the <i>ID</i> of a <i>Parameter</i> associated with them

### 902: Request Example

```
GET /api/4.0/profiles?name=ATS_EDGE_TIER_CACHE HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- cdn** The integral, unique identifier of the *CDN* to which this *Profile* belongs
- cdnName** The name of the *CDN* to which this *Profile* belongs
- description** The *Profile*'s *Description*
- id** The *Profile*'s *ID*
- lastUpdated** The date and time at which this *Profile* was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The *Profile*'s *Name*
- routingDisabled** The *Profile*'s *Routing Disabled* setting
- type** The *Profile*'s *Type*

### 903: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: QEpKM/
→DwHBRvue9K7XKrpwKFKhW6yCMQ2vSQgxE7dWFGJaqC4KOU092bsJU/
→5fjI9qlB+1uMT2kz6mFb1Wzp/w==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 20:40:31 GMT
Content-Length: 220

{ "response": [
  {
    "id": 9,
    "lastUpdated": "2018-12-05 17:51:00+00",
    "name": "ATS_EDGE_TIER_CACHE",
    "description": "Edge Cache - Apache Traffic Server",
    "cdnName": "CDN-in-a-Box",
    "cdn": 2,
    "routingDisabled": false,
    "type": "ATS_PROFILE"
  }
]}
```

## POST

Creates a new *Profile*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:CREATE, PROFILE:READ

**Response Type** Object

## Request Structure

**cdn** The integral, unique identifier of the *CDN* to which this *Profile* shall belong

**description** The *Profile*’s *Description*

**name** The *Profile*’s *Name*

**routingDisabled** The *Profile*’s *Routing Disabled* setting

**type** The *Profile*’s *Type*

904: Request Example

```
POST /api/4.0/profiles HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)

(continued from previous page)

```
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 125
Content-Type: application/json

{
  "name": "test",
  "description": "A test profile for API examples",
  "cdn": 2,
  "type": "UNK_PROFILE",
  "routingDisabled": true
}
```

## Response Structure

- cdn** The integral, unique identifier of the *CDN* to which this *Profile* belongs
- cdnName** The name of the *CDN* to which this *Profile* belongs
- description** The *Profile*'s *Description*
- id** The *Profile*'s *ID*
- lastUpdated** The date and time at which this *Profile* was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The *Profile*'s *Name*
- routingDisabled** The *Profile*'s *Routing Disabled* setting
- type** The *Profile*'s *Type*

### 905: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→UGV3PCnYBY0J3siICR0f9VVRNdUK1+9zsDDP6T9yt6t+AoHckHe6bvz0li9to/
→fGhC2zz5l9Nc1ro4taJUDD8g==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 21:24:49 GMT
```

(continues on next page)

(continued from previous page)

Content-Length: 251

```
{ "alerts": [
  {
    "text": "profile was created.",
    "level": "success"
  }
],
"response": {
  "id": 16,
  "lastUpdated": "2018-12-07 21:24:49+00",
  "name": "test",
  "description": "A test profile for API examples",
  "cdnName": null,
  "cdn": 2,
  "routingDisabled": true,
  "type": "UNK_PROFILE"
}}
```

profiles/{{ID}}

PUT

Replaces the specified *Profile* with the one in the request payload

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:UPDATE, PROFILE:READ

**Response Type** Object

## Request Structure

Table 376: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> being modified

**cdn** The integral, unique identifier of the *CDN* to which this *Profile* will belong

**description** The *Profile*’s new *Description*

**name** The *Profile*’s new *Name*

**routingDisabled** The *Profile*’s new *Routing Disabled* setting

**type** The *Profile*'s new *Type*

**Warning:** Changing this will likely break something, be **VERY** careful when modifying this value

#### 906: Request Example

```
PUT /api/4.0/profiles/16 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 125
Content-Type: application/json

{
  "name": "test",
  "description": "A test profile for API examples",
  "cdn": 2,
  "type": "UNK_PROFILE",
  "routingDisabled": true
}
```

## Response Structure

**cdn** The integral, unique identifier of the *CDN* to which this *Profile* belongs

**cdnName** The name of the *CDN* to which this *Profile* belongs

**description** The *Profile*'s *Description*

**id** The *Profile*'s *ID*

**lastUpdated** The date and time at which this *Profile* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** The *Profile*'s *Name*

**routingDisabled** The *Profile*'s *Routing Disabled* setting

**type** The *Profile*'s *Type*

#### 907: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↪Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
```

(continues on next page)



(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Pnf+G9G3/
→+edt4b8PVsyGZHsNzaFEgphaGSminjRlRmMpWtuLAA20WZDUo3nX0Q081c2GCuFuEh9uMF2Vjeppg==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 21:45:06 GMT
Content-Length: 251

{ "alerts": [
  {
    "text": "profile was updated.",
    "level": "success"
  }
],
"response": {
  "id": 16,
  "lastUpdated": "2018-12-07 21:45:06+00",
  "name": "test",
  "description": "A test profile for API examples",
  "cdnName": null,
  "cdn": 2,
  "routingDisabled": true,
  "type": "UNK_PROFILE"
}}
```

## DELETE

Allows user to delete a *Profile*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:DELETE, PROFILE:READ

**Response Type** undefined

## Request Structure

Table 377: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> being deleted

### 908: Request Example

```
DELETE /api/4.0/profiles/16 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 909: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: HNmJkZaNW9yil08/
    ↳3TnqZ5FllH6Rp+jgp3KI46FZdojLYcu+8jEhDL1lokoirdrHyU4R1c3hjCI0urN7PVvWDA==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Dec 2018 21:55:33 GMT
Content-Length: 62

{ "alerts": [
  {
    "text": "profile was deleted.",
    "level": "success"
  }
]}
```

`profiles/{{ID}}/export`

GET

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PROFILE:READ, PARAMETER:READ

**Response Type** Object

## Request Structure

Table 378: Request Path Parameters

Parameter	Description
id	The <i>ID</i> of the <i>Profile</i> to be exported

910: Request Example

```
GET /api/4.0/profiles/3/export HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**profile** The exported *Profile*

**cdn** The name of the *CDN* to which this *Profile* belongs

**description** The *Profile*'s *Description*

**name** The *Profile*'s *Name*

**type** The *Profile*'s *Type*

**parameters** An array of *Parameters* in use by this *Profile*

**config\_file** The *Parameter*'s *Config File*

**name** *Name* of the *Parameter*

**value** The *Parameter*'s *Value*

## 911: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Disposition: attachment; filename="GLOBAL.json"
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳mzP7DVxFAGhICxqagwDyBDRea7oBZPMAx7NCDeOBVCRqlcCFFe7XL3JP58b80aaVOW/
    ↳2ZGfg/jpYF70cdDfzQA==
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 Sep 2019 20:14:42 GMT
Transfer-Encoding: gzip

{ "profile": {
  "name": "GLOBAL",
  "description": "Global Traffic Ops profile",
  "cdn": "ALL",
  "type": "UNK_PROFILE"
},
"parameters": [
  {
    "config_file": "global",
    "name": "tm.instance_name",
    "value": "Traffic Ops CDN"
  },
  {
    "config_file": "global",
    "name": "tm.toolname",
    "value": "Traffic Ops"
  }
]}

```

## profiles/{{ID}}/parameters

### GET

Retrieves all *Parameters* assigned to the *Profile*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PROFILE:READ, PARAMETER:READ

**Response Type** Array

### Request Structure

Table 379: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> for which <i>Parameters</i> will be listed

#### 912: Request Example

```
GET /api/4.0/parameters/profile/GLOBAL HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name* of the *Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

913: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: NudgZXUNyKNpmSFf856KEjyy+Pin/
    ↳bFhG9NoRBDaxYbRKt2T5fF5Ze7sUNZfFI5n/ZZsgbx6Tsgtfd7oM6j+eg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 21:08:56 GMT
Content-Length: 542

{ "response": [
  {
    "configFile": "global",
    "id": 4,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.instance_name",
    "secure": false,
    "value": "Traffic Ops CDN"
  },
  {
    "configFile": "global",
    "id": 5,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.toolname",
    "secure": false,
    "value": "Traffic Ops"
  },
  {
    "configFile": "regex_revalidate.config",
    "id": 7,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "maxRevalDurationDays",
    "secure": false,
    "value": "90"
  }
]}

```

## POST

Associates *Parameters* to a *Profile*. If the *Parameter* does not exist, creates it and associates it to the *Profile*. If the *Parameter* already exists, associates it to the *Profile*. If the *Parameter* is already associated with the *Profile*, keep the association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:UPDATE, PROFILE:READ, PARAMETER:READ

**Response Type** Object

## Request Structure

Table 380: Request Path Parameters

Name	Description
ID	The <i>ID</i> of the <i>Profile</i> to which <i>Parameters</i> will be assigned

This endpoint accepts two formats for the request payload:

**Single Object Format** For assigning a single *Parameter* to a single *Profile*

**Parameter Array Format** For making multiple assignments of *Parameters* to *Profiles* simultaneously

**Warning:** Most API endpoints dealing with *Parameters* treat *Secure* as a boolean value, whereas this endpoint takes the legacy approach of treating it as an integer. Be careful when passing data back and forth, as boolean values will **not** be accepted by this endpoint!

## Single Parameter Format

**configFile** The *Parameter*’s *Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter*’s *Value*

914: Response Example - Single Parameter Format

```
POST /api/4.0/profiles/18/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
```

(continues on next page)

(continued from previous page)

```
Accept: */*
Cookie: mojolicious=...
Content-Length: 99
Content-Type: application/json

{
  "name": "test",
  "configFile": "quest",
  "value": "A test parameter for API examples",
  "secure": 0
}
```

## Parameter Array Format

**configFile** The *Parameter's Config File*

**name** *Name of the Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 915: Request Example - Parameter Array Format

```
POST /api/4.0/profiles/18/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 212
Content-Type: application/json

[
  {
    "name": "test",
    "configFile": "quest",
    "value": "A test parameter for API examples",
    "secure": 0
  },
  {
    "name": "foo",
    "configFile": "bar",
    "value": "Another test parameter for API examples",
    "secure": 0
  }
]
```



## Response Structure

**parameters** An array of objects representing the *Parameters* which have been assigned

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

**profileId** The *ID* of the *Profile* to which the *Parameter(s)* have been assigned

**profileName** *Name* of the *Profile* to which the *Parameter(s)* have been assigned

### 916: Response Example - Single Parameter Format

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: R2QUyCaNvKvVv/PNVNmEd/ma5h/iP1fMJlqhv+x2jE/
    ↳zNpHJ1KVXt6s3btB8nnHv6IDF/gt5kIzQ0mbW5U8bpg==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 14:45:28 GMT
Content-Length: 253

{ "alerts": [
  {
    "text": "Assign parameters successfully to profile test",
    "level": "success"
  }
],
"response": {
  "parameters": [
    {
      "configFile": "quest",
      "name": "test",
      "secure": 0,
      "value": "A test parameter for API examples",
      "id": 126
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
],
"profileId": 18,
"profileName": "test"
}}
```

---

**Note:** The format of the request does not affect the format of the response. `parameters` will be an array either way.

---

## profiles/import

### POST

Imports a *Profile* that was exported via *profiles/{{ID}}/export*

---

**Note:** On import of the *Profile Parameters* if a *Parameter* already exists with the same *Name*, *Config File* and *Value* it will link that to the *Profile* instead of creating it.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:CREATE, PARAMETER:CREATE, PROFILE:READ, PARAMETER:READ

**Response Type** Object

## Request Structure

**profile** The exported *Profile*

**cdn** The name of the *CDN* to which this *Profile* belongs

**description** The *Profile*’s *Description*

**name** The *Profile*’s *Name*

**type** The *Profile*’s *Type*

**parameters** An array of *Parameters* in use by this *Profile*

**config\_file** The *Parameter*’s *Config File*

**name** *Name* of the *Parameter*

**value** The *Parameter*’s *Value*

917: Request Example

```
POST /api/4.0/profiles/import HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
Content-Type: application/json

{ "profile": {
  "name": "GLOBAL",
  "description": "Global Traffic Ops profile",
  "cdn": "ALL",
  "type": "UNK_PROFILE"
},
"parameters": [
  {
    "config_file": "global",
    "name": "tm.instance_name",
    "value": "Traffic Ops CDN"
  },
  {
    "config_file": "global",
    "name": "tm.toolname",
    "value": "Traffic Ops"
  }
]
}
```

## Response Structure

**cdn** The name of the *CDN* to which this *Profile* belongs

**description** The *Profile's Description*

**name** The *Profile's Name*

**type** The *Profile's Type*

**id** The *Profile's ID*

918: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→mzP7DVxFAGhICxqagwDyBDRea7oBZPMAx7NCDe0BVCRqlcCFFe7XL3JP58b80aaVOW/
→2ZGfg/jpYF70cdDfzQA==
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 Sep 2019 20:14:42 GMT
Transfer-Encoding: gzip

{ "alerts": [
  {
    "level": "success",
    "text": "Profile imported [ Global ] with 2 new and 0 existing.
→parameters"
  }
],
"response": {
  "cdn": "ALL",
  "name": "Global",
  "id": 18,
  "type": "UNK_PROFILE",
  "description": "Global Traffic Ops profile"
}}
```

`profiles/name/{{name}}/copy/{{copy}}`

## POST

Copy *Profile* to a new *Profile*. The new *Profile*'s *Name* must not already exist.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:CREATE, PROFILE:READ, PARAME-  
TER:READ

**Response Type** Object

## Request Structure

Table 381: Request Path Parameters

Name	Description
name	The <i>Name</i> of the new <i>Profile</i>
copy	The <i>Name</i> of <i>Profile</i> from which the copy will be made

### 919: Request Example

```
POST /api/4.0/profiles/name/GLOBAL_copy/copy/GLOBAL HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.62.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**description** The new *Profile*'s *Description*

**id** The *ID* of the new *Profile*

**idCopyFrom** The *ID* of the *Profile* from which the copy was made

**name** The *Name* of the new *Profile*

**profileCopyFrom** The *Name* of the *Profile* from which the copy was made

### 920: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Fri, 07 Dec 2018 22:03:54 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→r6V9viEZui1WCns0AUGEx1MtxjjXiU8SZV0tSQjeq7ZJDL15s8fMmjJdR/
→HRWduHn7Ax6GzYhoKwnIjMyc7ZWg==
Content-Length: 252
```

(continues on next page)

(continued from previous page)

```
{ "alerts": [
  {
    "level": "success",
    "text": "Created new profile [ GLOBAL_copy ] from existing_
→profile [ GLOBAL ]"
  }
],
"response": {
  "idCopyFrom": 1,
  "name": "GLOBAL_copy",
  "profileCopyFrom": "GLOBAL",
  "id": 17,
  "description": "Global Traffic Ops profile, DO NOT DELETE"
}}
```

`profiles/name/{{name}}/parameters`

GET

Retrieves all *Parameters* associated with a given *Profile*

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** PROFILE:READ, PARAMETER:READ

**Response Type** None

## Request Structure

Table 382: Request Path Parameters

Name	Description
name	The <i>Name</i> of the <i>Profile</i> for which <i>Parameters</i> will be listed

921: Request Example

```
GET /api/4.0/profiles/name/GLOBAL/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**configFile** The *Parameter's Config File*

**id** The *Parameter's ID*

**lastUpdated** The date and time at which this *Parameter* was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** *Name of the Parameter*

**profiles** An array of *Profile Names* that use this *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 922: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: NudgZXUNyKNpmSFf856KEjyy+Pin/
→bFhG9NoRBDaxYbRKt2T5fF5Ze7sUNZfFI5n/ZZsgbx6Tsgtfd7oM6j+eg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 05 Dec 2018 21:52:08 GMT
Content-Length: 542

{ "response": [
  {
    "configFile": "global",
    "id": 4,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.instance_name",
    "secure": false,
    "value": "Traffic Ops CDN"
  },
  {
    "configFile": "global",
    "id": 5,
    "lastUpdated": "2018-12-05 17:50:49+00",
    "name": "tm.toolname",
    "secure": false,
    "value": "Traffic Ops"
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "configFile": "regex_revalidate.config",
      "id": 7,
      "lastUpdated": "2018-12-05 17:50:49+00",
      "name": "maxRevalDurationDays",
      "secure": false,
      "value": "90"
    }
  ]
}
```

## POST

Associates *Parameters* to a *Profile*. If the *Parameter* does not exist, creates it and associates it to the *Profile*. If the *Parameter* already exists, associates it to the *Profile*. If the *Parameter* is already associated with the *Profile*, keep the association.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** PROFILE:UPDATE, PROFILE:READ, PARAMETER:READ

**Response Type** Object

## Request Structure

Table 383: Request Path Parameters

Name	Description
name	The <i>Name</i> of the <i>Profile</i> to which <i>Parameters</i> will be assigned

This endpoint accepts two formats for the request payload:

**Single Parameter Format** Specify a single parameter to assign to the specified profile

**Parameter Array Format** Specify multiple parameters to assign to the specified profile

**Warning:** Most API endpoints dealing with parameters treat `secure` as a boolean value, whereas this endpoint takes the legacy approach of treating it as an integer. Be careful when passing data back and forth, as boolean values will **not** be accepted by this endpoint!



## Single Parameter Format

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 923: Request Example - Single Parameter Format

```
POST /api/4.0/profiles/name/test/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 99
Content-Type: application/json

{
  "name": "test",
  "configFile": "quest",
  "value": "A test parameter for API examples",
  "secure": 0
}
```

## Parameter Array Format

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

### 924: Request Example - Parameter Array Format

```
POST /api/4.0/profiles/name/test/parameters HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 212
Content-Type: application/json

[ {
  "name": "test",
  "configFile": "quest",
```

(continues on next page)

(continued from previous page)

```

    "value": "A test parameter for API examples",
    "secure": 0
  },
  {
    "name": "foo",
    "configFile": "bar",
    "value": "Another test parameter for API examples",
    "secure": 0
  }
]
```

## Response Structure

**parameters** An array of objects representing the *Parameters* which have been assigned

**configFile** The *Parameter's Config File*

**name** *Name* of the *Parameter*

**secure** A boolean value that describes whether or not the *Parameter* is *Secure*

**value** The *Parameter's Value*

**profileId** The *ID* of the *Profile* to which the *Parameter(s)* have been assigned

**profileName** *Name* of the *Profile* to which the *Parameter(s)* have been assigned

### 925: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: HQWq0koYHjvcxheWPrH0b0oZnUC+qLG1L040jtsLLnZYVUIu/
→qgJrzvziPnKq3FEHUWaZrnDCZM/iZD8AXOKBw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 14:20:22 GMT
Content-Length: 357

{ "alerts": [
  {
    "text": "Assign parameters successfully to profile test",
```

(continues on next page)

(continued from previous page)

```

        "level": "success"
    }
],
"response": {
    "parameters": [
        {
            "configFile": "quest",
            "name": "test",
            "secure": 0,
            "value": "A test parameter for API examples",
            "id": 126
        },
        {
            "configFile": "bar",
            "name": "foo",
            "secure": 0,
            "value": "Another test parameter for API examples",
            "id": 129
        }
    ],
    "profileId": 18,
    "profileName": "test"
}}

```

---

**Note:** The format of the request does not affect the format of the response. `parameters` will be an array either way.

---

## regions

### GET

Retrieves information about *Regions*

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** REGION:READ

**Response Type** Object

## Request Structure

Table 384: Request Query Parameters

Name	Required	Description
division	no	Filter <i>Regions</i> by the integral, unique identifier of the <i>Division</i> which contains them
id	no	Filter <i>Regions</i> by integral, unique identifier
name	no	Filter <i>Regions</i> by name
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n</i> <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 926: Request Example

```
GET /api/4.0/regions?division=1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**divisionName** The name of the division which contains this region

**divisionId** The integral, unique identifier of the division which contains this region

**id** An integral, unique identifier for this region

**lastUpdated** The date and time at which this region was last updated, in *Traffic Ops's Custom Date/Time Format*

**name** The region name

### 927: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie (continues on next page)
```

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→nSYbR+fRXaxhYl7dWgf0Udo2AsiXEnwvED1CPbk7ZNWK03I3TOhtmCQx9ABnJJ6xKYnl6EKMeopVTK0n
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 01:58:38 GMT
Content-Length: 117

{ "response": [
  {
    "divisionName": "Quebec",
    "division": 1,
    "id": 2,
    "lastUpdated": "2018-12-05 17:50:58+00",
    "name": "Montreal"
  }
]}
```

## POST

Creates a new region

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** REGION:CREATE, REGION:READ

**Response Type** Object

## Request Structure

**division** The integral, unique identifier of the division which shall contain the new region<sup>1</sup>

**divisionName** The name of the division which shall contain the new region<sup>1</sup>

**name** The name of the region

<sup>1</sup> The only “division” key that actually matters in the request body is `division`; `divisionName` is not validated and has no effect - particularly not the effect of re-naming the division - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the division. Note that if `divisionName` is not present in the request body it will be `null` in the response, but again further requests will show the true division name (provided it has been assigned to a division).

#### 928: Request Example

```
POST /api/4.0/regions HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 65
Content-Type: application/json

{
  "name": "Manchester",
  "division": "4",
  "divisionName": "England"
}
```

### Response Structure

- divisionName** The name of the division which contains this region
- divisionId** The integral, unique identifier of the division which contains this region
- id** An integral, unique identifier for this region
- lastUpdated** The date and time at which this region was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The region name

#### 929: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳ezxk+iP7o7KE7zpWmGc0j8nz5k+1wAzY0HiNiA2xswTQrt+N+6CgQqUV2r9G1HAsPNr0HF2PhYs/
  ↳Xr7DrY0w0A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 02:14:45 GMT
Content-Length: 178

{ "alerts": [
```

(continues on next page)

(continued from previous page)

```
{
  "text": "region was created.",
  "level": "success"
},
"response": {
  "divisionName": "England",
  "division": 3,
  "id": 5,
  "lastUpdated": "2018-12-06 02:14:45+00",
  "name": "Manchester"
}}
```

## DELETE

Deletes a region. If no query parameter is specified, a 400 Bad Request response is returned.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** REGION:DELETE

**Response Type** Object

## Request Structure

Table 385: Request Query Parameters

Name	Required	Description
id	no	Delete <i>Region</i> by integral, unique identifier
name	no	Delete <i>Region</i> by name

930: Request Example

```
DELETE /api/4.0/regions?name=Manchester HTTP/1.1
User-Agent: curl/7.29.0
Host: trafficops.infra.ciab.test
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

931: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 07 Feb 2020 13:56:24
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳yNqXKcoiohEbJrEkH8LD1zifh87dIusuvUqgQnYueyKqCXkfd5bQvQ00hQ2AAdSZa/
    ↳oe2SAqMjojGsUlxHtIQw==
X-Server-Name: traffic_ops_golang/
Date: Fri, 07 Feb 2020 12:56:24 GMT
Content-Length: 62

{
  "alerts": [
    {
      "text": "region was deleted.",
      "level": "success"
    }
  ]
}
```



regions/{{ID}}

PUT

Updates a *Region*.

**Auth. Required** Yes

**Role(s) Required** “admin” or “operator”

**Permissions Required** REGION:UPDATE, REGION:READ

**Response Type** Object

## Request Structure

Table 386: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the region to update

**division** The new integral, unique identifier of the division which shall contain the region<sup>1</sup>

**divisionName** The new name of the division which shall contain the region<sup>1</sup>

**name** The new name of the region

### 932: Request Example

```
PUT /api/4.0/regions/5 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 60
Content-Type: application/json

{
  "name": "Leeds",
  "division": 3,
  "divisionName": "England"
}
```

<sup>1</sup> The only “division” key that actually matters in the request body is `division`; `divisionName` is not validated and has no effect - particularly not the effect of re-naming the division - beyond changing the name in the API response to this request. Subsequent requests will reveal the true name of the division. Note that if `divisionName` is not present in the request body it will be `null` in the response, but again further requests will show the true division name (provided it has been assigned to a division).

## Response Structure

- divisionName** The name of the division which contains this region
- divisionId** The integral, unique identifier of the division which contains this region
- id** An integral, unique identifier for this region
- lastUpdated** The date and time at which this region was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The region name

### 933: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳7SVj4q7dtSTNQEJlBApEwlad28WBVFnpdHaatoIpNfeLltfcpcdVTcOKB4JXQv7rlSD2p/
    ↳TxBQC6EXpxwYTnKQ==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 02:23:40 GMT
Content-Length: 173

{ "alerts": [
  {
    "text": "region was updated.",
    "level": "success"
  }
],
"response": {
  "divisionName": "England",
  "division": 3,
  "id": 5,
  "lastUpdated": "2018-12-06 02:23:40+00",
  "name": "Leeds"
}}
```

## roles

### GET

Retrieves all user *Roles*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** ROLE:READ

**Response Type** Array

### Request Structure

Table 387: Request Query Parameters

Name	Re- quired	Description
id	no	Return only the <i>Role</i> identified by this integral, unique identifier
name	no	Return only the <i>Role</i> with this name
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 934: Request Example

```
GET /api/4.0/roles?name=read-only HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**permissions** An array of the names of the Permissions given to this *Role*

**description** A description of the *Role*

**id** The integral, unique identifier for this *Role*

**name** The name of the *Role*

### 935: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳TEDXlQqWMSnJbL10JtFdbw0nqciNpjc4bd6m7iAB8aymakWeF+ghs1k5LayjdzHcjeDE8UNF/
    ↳HXSxOFvoLFEuA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 25 Aug 2021 20:10:34 GMT
Content-Length: 888

{ "response": [
  {
    "name": "read-only",
    "description": "Has access to all read capabilities",
    "permissions": [
      "auth",
      "api-endpoints-read",
      "asns-read",
      "cache-config-files-read",
      "cache-groups-read",
      "capabilities-read",
```

(continues on next page)

(continued from previous page)

```

        "cdns-read",
        "cdn-security-keys-read",
        "change-logs-read",
        "consistenthash-read",
        "coordinates-read",
        "delivery-services-read",
        "delivery-service-security-keys-read",
        "delivery-service-requests-read",
        "delivery-service-servers-read",
        "divisions-read",
        "to-extensions-read",
        "federations-read",
        "hwinfo-read",
        "jobs-read",
        "origins-read",
        "parameters-read",
        "phys-locations-read",
        "profiles-read",
        "regions-read",
        "roles-read",
        "server-capabilities-read",
        "servers-read",
        "service-categories-read",
        "stats-read",
        "statuses-read",
        "static-dns-entries-read",
        "steering-read",
        "steering-targets-read",
        "system-info-read",
        "tenants-read",
        "types-read",
        "users-read"
    ],
    "lastUpdated": "2021-05-03T14:50:18.93513-06:00",
}
]]}

```

### POST

Creates a new *Role*.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ROLE:CREATE, ROLE:READ

**Response Type** Object

### Request Structure

**permissions** An optional array of permission names that will be granted to the new *Role*<sup>1</sup>

**description** A helpful description of the *Role*’s purpose.

**name** The name of the new *Role*

#### 936: Request Example

```
POST /api/4.0/roles HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "name": "test",
  "description": "quest"
}
```

### Response Structure

**permissions** An array of the names of the Permissions given to this *Role*

---

**Tip:** This can be null *or* empty, depending on whether it was present in the request body, or merely empty. Obviously, it can also be a populated array.

---

**description** A description of the *Role*

**id** The integral, unique identifier for this *Role*

---

<sup>1</sup> permissions cannot include permissions that are not included in the permissions of the requesting user. In POST requests, if permissions is omitted or explicitly null, it is treated as an empty set/array.

**name** The name of the *Role*

#### 937: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: gzfc7m/
    ↳in5vVsVP+Y9h6JJfDhgpXKn9VAzoiPENhKbQfP8Q6jug08Rt2AK/
    ↳3Nz1cx5zZ8P9IjVxDdIg7m1C8bw==
X-Server-Name: traffic_ops_golang/
Date: Wed, 04 Sep 2019 17:44:42 GMT
Content-Length: 128

{ "alerts": [{
    "text": "role was created.",
    "level": "success"
}],
  "response": {
    "name": "test",
    "description": "quest",
    "permissions": null
  }
}
```

## PUT

Replaces an existing *Role* with one provided by the request<sup>2</sup>.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ROLE:UPDATE, ROLE:READ

**Response Type**

<sup>2</sup> The special *Role* with the name “admin” cannot be modified or deleted - regardless of user Permissions.

## Request Structure

Table 388: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Role</i> to be updated

**permissions** An optional array of permission names that will be granted to the new *Role*

**Warning:** When not present, the affected *Role*'s Permissions will be unchanged - *not* removed, unlike when the array is empty.

**description** A helpful description of the *Role*'s purpose.

**name** The new name of the *Role*

### 938: Request Example

```
PUT /api/4.0/roles?name=test HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "name": "test",
  "description": "quest_updated"
}
```

## Response Structure

**permissions** An array of the names of the Permissions given to this *Role*

---

**Tip:** This can be null *or* empty, depending on whether it was present in the request body, or merely empty. Obviously, it can also be a populated array.

---

**Warning:** If no permissions array was given in the request, this will *always* be null, even if the *Role* has Permissions that would have gone unchanged.



**description** A description of the *Role*

**name** The name of the *Role*

#### 939: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳mlHQenE1Q3gjrIK2lC2hfueQ0aTCpdYefboN0A9vYPUIwTiaF5ZaAMPQBdfGyiAhgHRxowITs3bR7s1L+
X-Server-Name: traffic_ops_golang/
Date: Thu, 05 Sep 2019 12:56:46 GMT
Content-Length: 136

{
  "alerts": [
    {
      "text": "role was updated.",
      "level": "success"
    }
  ],
  "response": {
    "name": "test",
    "description": "quest_updated",
    "permissions": null
  }
}
```

## DELETE

Deletes a *Role*?

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ROLE:DELETE, ROLE:READ

**Response Type** undefined

## Request Structure

Table 389: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Role</i> to be deleted

### 940: Request Example

```
DELETE /api/4.0/roles?name=test HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 941: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 10jeFZihtbvAus/XyHAW8rhgS9JBD+X/
    ↳ezCp1iExYkEcHxN4gjr1L6x8zDFXORueBSlFldgtbWKT7QsmwCHUWA==
X-Server-Name: traffic_ops_golang/
Date: Thu, 05 Sep 2019 13:02:06 GMT
Content-Length: 60

{ "alerts": [{
    "text": "role was deleted.",
    "level": "success"
}]}
```

## server\_capabilities

### GET

Retrieves *Server Capabilities*.

**Auth. Required** Yes

**Roles Required** “read-only”

**Permissions Required** SERVER-CAPABILITY:READ

**Response Type** Array

## Request Structure

Table 390: Request Query Parameters

Name	Required	Description
name	no	Return the <i>Server Capability</i> with this name

942: Request Structure

```
GET /api/4.0/server_capabilities?name=RAM HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**name** The name of this *Server Capability*

**lastUpdated** The date and time at which this *Server Capability* was last updated, in ISO-like format

943: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpgT3YRyKJ/
→p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 21:36:13 GMT
Content-Length: 68

{
  "response": [
    {
      "name": "RAM",
      "lastUpdated": "2019-10-07 20:38:24+00"
    }
  ]
}
```

## POST

Create a new *Server Capability*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER-CAPABILITY:CREATE, SERVER-CAPABILITY:READ

**Response Type** Object

## Request Structure

**name** The name of the *Server Capability*

944: Request Example

```
POST /api/4.0/server_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 15
Content-Type: application/json

{
  "name": "RAM"
}
```

## Response Structure

**name** The name of this *Server Capability*

**lastUpdated** The date and time at which this *Server Capability* was last updated, in ISO-like format

### 945: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ysdopC//
    ↳JQI79BRUa61s6M2HzHxYHpo5Rdcuau0oqCYxiV0oUhNZfOVydVkv8zDN2qA374XKnym4kWj3VzQIXg==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:10:00 GMT
Content-Length: 137

{
  "alerts": [
    {
      "text": "server capability was created.",
      "level": "success"
    }
  ],
  "response": {
    "name": "RAM",
    "lastUpdated": "2019-10-07 22:10:00+00"
  }
}
```

## PUT

Update an existing *Server Capability*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER-CAPABILITY:UPDATE, SERVER-CAPABILITY:READ

**Response Type** Object

## Request Structure

**name** The name of the *Server Capability*

946: Request Example

```
PUT /api/4.0/server_capabilities?name=RAM HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 15
Content-Type: application/json

{
  "name": "HDD"
}
```

## Response Structure

**name** The name of this *Server Capability*

**lastUpdated** The date and time at which this *Server Capability* was last updated, in ISO-like format

947: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ysdopC//
  ↳JQI79BRUa61s6M2HzHxYHpo5RdcuauOoqCYxiV0oUhNZfOVydVkv8zDN2qA374XKnym4kWj3VzQIXg==
X-Server-Name: traffic_ops_golang/
Date: Wed, 03 March 2021 21:22:08 GMT
Content-Length: 137

{
  "alerts": [
    {
      "text": "server capability was updated.",
      "level": "success"
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```

    }
  ],
  "response": {
    "name": "HDD",
    "lastUpdated": "2021-03-03 21:22:08+00"
  }
}

```

## DELETE

Deletes a specific *Server Capability*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER-CAPABILITY:DELETE, SERVER-CAPABILITY:READ

**Response Type** undefined

## Request Structure

Table 391: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Server Capability</i> to be deleted

### 948: Request Example

```

DELETE /api/4.0/server_capabilities?name=RAM HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...

```

## Response Structure

### 949: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie

```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 8zCAATbCzcqiqigGVBy7WF1duDuXu1Wg2DBe9yfqTw/
→c+yhE2eUk73hFTA/Oqt0kocaN7+1GkbFdPkQPvbnRaA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 20:44:40 GMT
Content-Length: 72

{
  "alerts": [
    {
      "text": "server capability was deleted.",
      "level": "success"
    }
  ]
}
```

## server\_server\_capabilities

### GET

Gets all associations of *Server Capabilities* to *cache servers*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** SERVER:READ, SERVER-CAPABILITY:READ

**Response Type** Array



## Request Structure

Table 392: Request Query Parameters

Name	Re- quired	Description
serverId	no	Filter <i>Server Capability</i> associations by the integral, unique identifier of the server to which they are assigned
server- Host- Name	no	Filter <i>Server Capability</i> associations by the host name of the server to which they are assigned
server- Ca- pabil- ity	no	Filter <i>Server Capability</i> associations by <i>Server Capability</i> name
or- derby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <b>response</b> array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <b>limit</b> long and the first page is 1. If <b>offset</b> was defined, this query parameter has no effect. <b>limit</b> must be defined to make use of <b>page</b> .

### 950: Request Example

```
GET /api/4.0/server_server_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**serverHostName** The server’s host name

**serverId** The server’s integral, unique identifier

**lastUpdated** The date and time at which this association between the server and the *Server Capability* was last updated, in *Traffic Ops’s Custom Date/Time Format*

**serverCapability** The *Server Capability*’s name

## 951: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
    ↳jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
    ↳DbBwhMAOJoeqDIImoDlrLnrVjQG04AooA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 150

{
  "response": [
    {
      "lastUpdated": "2019-10-07 22:05:31+00",
      "serverHostName": "atlanta-org-1",
      "serverId": 260,
      "serverCapability": "ram"
    },
    {
      "lastUpdated": "2019-10-07 22:05:31+00",
      "serverHostName": "atlanta-org-2",
      "serverId": 261,
      "serverCapability": "disk"
    }
  ]
}

```

## POST

Associates a *Server Capability* to a server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:UPDATE,      SERVER:READ,      SERVER-  
CAPABILITY:READ

**Response Type** Object

## Request Structure

**serverId** The integral, unique identifier of a server to be associated with a *Server Capability*

**serverCapability** The *Server Capability*'s name to associate

---

**Note:** The server referenced must be either an *Edge-tier* or *Mid-tier cache server*.

---

### 952: Request Example

```
POST /api/4.0/server_server_capabilities HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 84
Content-Type: application/json

{
  "serverId": 1,
  "serverCapability": "disk"
}
```

## Response Structure

**serverId** The integral, unique identifier of the newly associated server

**lastUpdated** The date and time at which this association between the server and the *Server Capability* was last updated, in *Traffic Ops's Custom Date/Time Format*

**serverCapability** The *Server Capability*'s name

### 953: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
  ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
  ↳eQrl48zWids0kDpfCYmmtYMpegjnFxfOVvlBYxxLSfp7P7p6oWX4uiC+/
  ↳Cfh2X9i3G+MQ36eH95gukJqOBOGbQ==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 157

{
  "alerts": [
    {
      "text": "server server_capability was created.",
      "level": "success"
    }
  ],
  "response": {
    "lastUpdated": "2019-10-07 22:15:11+00",
    "serverId": 1,
    "serverCapability": "disk"
  }
}
```

## DELETE

Disassociate a server from a *Server Capability*.

---

**Note:** If the serverCapability is a *Server Capability* required by a *Delivery Service* that to which the server is assigned the DELETE will be blocked until either the server is unassigned from the *Delivery Service* or the *Server Capability* is no longer required by the *Delivery Service*.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:UPDATE, SERVER:READ, SERVER-CAPABILITY:READ

**Response Type** undefined

## Request Structure

Table 393: Request Query Parameters

Name	Re-quired	Description
serverId	yes	The integral, unique identifier of the server to disassociate
serverCapabil-ity	yes	term: <i>Server Capability</i> name to disassociate from given server

#### 954: Request Example

```
DELETE /api/4.0/server_server_capabilities?serverId=1&
→serverCapability=disk HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

#### 955: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: UF03/
→jcBFmFZM7CsrsIwTfPc5v8gUiXqJm6BNp1boPb4EQBnWNXZh/
→DbBwhMAOJoeqDImoDlrLnrVjQG04AooA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Oct 2019 22:15:11 GMT
Content-Length: 96

{
  "alerts": [
    {
      "text": "server server_capability was deleted.",
      "level": "success"
    }
  ]
}
```

## servercheck

### See also:

*Check Extensions*

### GET

Fetches identifying and meta information as well as “check” values regarding all servers that have a *Type* with a name beginning with “EDGE” or “MID” (ostensibly this is equivalent to all *cache servers*).

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** SERVER-CHECK:READ, SERVER:READ

**Response Type** Array

### Request Structure

Table 394: Request Query Parameters

Name	Required	Description
id	no	Return only <i>cache servers</i> with this integral, unique identifier (id)
hostName	no	Return only <i>cache servers</i> with this host_name

956: Request Example with hostName query param

```
GET /api/4.0/servercheck?hostName=edge HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

957: Request Example with id query param

```
GET /api/4.0/servercheck?id=12 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**adminState** The name of the server’s *Status* - called “adminState” for legacy reasons

**cacheGroup** The name of the *Cache Group* to which the server belongs

**checks** An optionally present map of the names of “checks” to their values. Only numeric and boolean checks are represented, and boolean checks are represented as integers with 0 meaning “false” and 1 meaning “true”. Will not appear if the server in question has no valued “checks”.

**hostName** The (short) hostname of the server

**id** The server’s integral, unique identifier

**profile** The name of the *Profile* used by the server

**revalPending** A boolean that indicates whether or not the server has pending revalidations

**type** The name of the server’s *Type*

**updPending** A boolean that indicates whether or not the server has pending updates

958: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Thu, 18 Feb 2021 20:00:19
    ↳GMT; Max-Age=3600; HttpOnly
X-Server-Name: traffic_ops_golang/
Date: Thu, 18 Feb 2021 19:00:19 GMT
Content-Length: 352
```

(continues on next page)

(continued from previous page)

```
{ "response": [
  {
    "adminState": "REPORTED",
    "cacheGroup": "CDN_in_a_Box_Edge",
    "id": 12,
    "hostName": "edge",
    "revalPending": false,
    "profile": "ATS_EDGE_TIER_CACHE",
    "type": "EDGE",
    "updPending": false
  }
]}
```

## POST

Post a server check result to the “serverchecks” table. Updates the resulting value from running a given check extension on a server.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** SERVER-CHECK:CREATE, SERVER-CHECK:READ,  
SERVER:READ

**Response Type** Object

## Request Structure

The request only requires to have either `host_name` or `id` defined.

**host\_name** The hostname of the server to which this “servercheck” refers.

**id** The id of the server to which this “servercheck” refers.

**servercheck\_short\_name** The short name of the “servercheck”.

**value** The value of the “servercheck”

### 959: Request Example

```
POST /api/4.0/servercheck HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

<sup>1</sup> No roles are required to use this endpoint, however access is controlled by username. Only the reserved user extension is permitted the use of this endpoint.



(continued from previous page)

```
Content-Length: 113
Content-Type: application/json

{
  "id": 1,
  "host_name": "edge",
  "servercheck_short_name": "test",
  "value": 1
}
```

## Response Structure

960: Response Example

```
{ "alerts": [
  {
    "level": "success",
    "text": "Server Check was successfully updated."
  }
]}
```

### servercheck/extensions

#### See also:

*Managing Traffic Ops Extensions*

#### GET

Retrieves the list of Traffic Ops extensions.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** SERVER-CHECK:READ, SERVER:READ

**Response Type** Array

## Request Structure

Table 395: Request Query Parameters

Name	Re-quired	Description
id	no	Filter TO Extensions by the integral, unique identifier of an Extension
name	no	Filter TO Extensions by the name of an Extension
script_file	no	Filter TO Extensions by the base filename of the script that runs for the Extension
is-active	no	Boolean used to return either only active (1) or inactive(0) TO Extensions
type	no	Filter TO Extensions by the type of Extension
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit.
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 961: Request Example

```
GET /api/4.0/servercheck/extensions HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**additional\_config\_json** A string containing a JSON-encoded object with extra configuration options... inside a JSON object...

**description** A short description of the extension

---

**Note:** This is, unfortunately, null for all default extensions

---

**id** An integral, unique identifier for this extension definition

**info\_url** A URL where info about this extension may be found

**isactive** An integer describing the boolean notion of whether or not the extension is active; one of:

**0** disabled

1 enabled

**name** The name of the extension

**script\_file** The base filename of the script that runs for the extension

**servercheck\_shortname** The name of the column in the table at ‘Monitor’ -> ‘Cache Checks’ in Traffic Portal, where “Check Extension” output is displayed

**type** The Check *Type* of the extension. This will always be a CHECK\_EXTENSION type with the naming convention of CHECK\_EXTENSION\_\*.

**version** A (hopefully) semantic version number describing the version of the plugin

## 962: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Tue, 11 Dec 2018 20:51:48 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: n73jg9XR4V5Cwqq56Rf3wuIi99k3mM5u2NAjcZ/
→gQB8jvAFymDlnZqKeJ+wTll1vjIsHpXCOVXV7+5UGakLgA==
Transfer-Encoding: chunked

{ "response": [
  {
    "script_file": "ToPingCheck.pl",
    "version": "1.0.0",
    "name": "ILO_PING",
    "description": null,
    "info_url": "-",
    "additional_config_json": "{ check_name: \"ILO\", \"base_url\
→\": \"https://localhost\", \"select\": \"ilo_ip_address\", \"cron\": \
→\"9 * * * *\" }",
    "isactive": 1,
    "type": "CHECK_EXTENSION_BOOL",
    "id": 1,
    "servercheck_short_name": "ILO"
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "script_file": "ToPingCheck.pl",
      "version": "1.0.0",
      "name": "10G_PING",
      "description": null,
      "info_url": "-",
      "additional_config_json": "{ check_name: \"10G\", \"base_url\
→\": \"https://localhost\", \"select\": \"ip_address\", \"cron\": \"18_
→* * * *\" }",
      "isactive": 1,
      "type": "CHECK_EXTENSION_BOOL",
      "id": 2,
      "servercheck_short_name": "10G"
    }
  ]}

```

## POST

Creates a new Traffic Ops check extension.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** SERVER-CHECK:CREATE, SERVER-CHECK:READ,  
SERVER:READ

**Response Type** undefined

## Request Structure

**additional\_config\_json** An optional string containing a JSON-encoded object with extra configuration options... inside a JSON object...

**description** A short description of the extension

**info\_url** A URL where info about this extension may be found

**isactive** An integer describing the boolean notion of whether or not the extension is active; one of:

**0** disabled

**1** enabled

**name** The name of the extension

<sup>1</sup> No roles are required to use this endpoint, however access is controlled by username. Only the reserved user extension is permitted the use of this endpoint.

**script\_file** The base filename of the script that runs for the extension

**See also:**

*Managing Traffic Ops Extensions* for details on where the script should be located on the Traffic Ops server

**servercheck\_shortname** The name of the column in the table at ‘Monitor’ -> ‘Cache Checks’ in Traffic Portal, where “Check Extension” output is displayed

**type** The *Type* of extension. Must be CHECK\_EXTENSION type with the naming convention of CHECK\_EXTENSION\_.\*.

**version** A (hopefully) semantic version number describing the version of the plugin

#### 963: Request Example

```
POST /api/4.0/servercheck/extensions HTTP/1.1
Host: cache1.example.com:6443
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 208
Content-Type: application/json

{
  "name": "test",
  "version": "0.0.1-1",
  "info_url": "",
  "script_file": "",
  "isactive": 0,
  "description": "A test extension for API examples",
  "servercheck_short_name": "test",
  "type": "CHECK_EXTENSION_NUM"
}
```

## Response Structure

#### 964: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Date: Wed, 12 Dec 2018 16:37:44 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 7M67PYnli6WzGQFS3g8Gh1SOyq6VENZMqm/
→kUffOTLLFfuWSEuSLA65R5R+VyJiNjdqOG5Bp78mk+JYcqhtVGw==
Content-Length: 89

{ "supplemental":
  {
    "id": 5
  },
  "alerts": [{
    "level": "success",
    "text": "Check Extension Loaded."
  }]
}
```

## servercheck/extensions/{ID}

### DELETE

Deletes a Traffic Ops server check extension definition. This does **not** delete the actual extension file.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** SERVER-CHECK:DELETE, SERVER-CHECK:READ,  
SERVER:READ

**Response Type** undefined

### Request Structure

Table 396: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the extension definition to be deleted

<sup>1</sup> No roles are required to use this endpoint, however access is controlled by username. Only the reserved user extension is permitted the use of this endpoint.

#### 965: Request Example

```
DELETE /api/4.0/servercheck/extensions/16 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

#### 966: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Wed, 12 Dec 2018 16:33:52 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: EB0Nu85azbGzaehDTAODP3NPqWbByIza1XQhgwtsW2WTXyK/
    ↳dxQtncp0YiJXy00tH9H+n+6BBfojB0b5h0dFPA==
Content-Length: 60

{ "alerts": [
  {
    "level": "success",
    "text": "Extension deleted."
  }
]}
```

### servers

#### GET

Retrieves properties of all servers across all CDNs.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** SERVER:READ, DELIVERY-SERVICE:READ, CDN:READ, PHYSICAL-LOCATION:READ, CACHE-GROUP:READ, TYPE:READ, PROFILE:READ

**Response Type** Array

## Request Structure

Table 397: Request Query Parameters

Name	Required	Description
cacheGroup	no	Return only those servers within the <i>Cache Group</i> that has this <i>ID</i>
cacheGroupName	no	Return only those servers within the <i>Cache Group</i> that has this <i>Name</i>
dsId	no	Return only those servers assigned to the <i>Delivery Service</i> identified by this integral, unique identifier. If the Delivery Service has a <i>Topology</i> assigned to it, the <i>servers</i> endpoint will return each server whose <i>Cache Group</i> is associated with a <i>Topology Node</i> of that Topology and has the <i>Server Capabilities</i> that are <i>required by the Delivery Service</i> but excluding <i>Origin Servers</i> that are not assigned to the Delivery Service. For more information, see <i>Configure Multi-Site Origin</i> .
hostName	no	Return only those servers that have this (short) hostname
id	no	Return only the server with this integral, unique identifier
profileName	no	Return only those servers that are using the <i>Profile</i> that has this <i>Name</i>
status	no	Return only those servers with this status - see <i>Health Protocol</i>
type	no	Return only servers of this <i>Type</i>
topology	no	Return only servers who belong to cachegroups assigned to the <i>Topology</i> identified by this name
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n<sup>th</sup></i> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 967: Request Example

```
GET /api/4.0/servers?hostName=mid HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)



(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** The integral, unique identifier of the CDN to which the server belongs

**cdnName** Name of the CDN to which the server belongs

**configUpdateTime** The last time an update was requested for this server. This field defaults to standard epoch

**configApplyTime** The last time an update was applied for this server. This field defaults to standard epoch

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be null

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service<sup>1</sup>

**iloIpGateway** The IPv4 gateway address of the server's ILO service<sup>Page 1422, 1</sup>

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service<sup>Page 1422, 1</sup>

**iloPassword** The password of the of the server's ILO service user<sup>Page 1422, 1</sup> - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' *Role(s)*

**iloUsername** The user name for the server's ILO service<sup>Page 1422, 1</sup>

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

---

<sup>1</sup> For more information see the [Wikipedia page on Lights-Out management](#).

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**routerPortName** The human-readable name of the router responsible for reaching this server's interface.

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server's interface.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server is in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the physical location where the server resides

**physLocationId** An integral, unique identifier for the physical location where the server resides

**profileNames** List of *Name* of the *Profiles* used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**revalUpdateTime** The last time a content invalidation/revalidation request was submitted for this server. This field defaults to standard epoch

**revalApplyTime** The last time a content invalidation/revalidation request was applied by this server. This field defaults to standard epoch

**rack** A string indicating "server rack" location

**status** The *Status* of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

See also:

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the *Type* of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (*t3c*, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 968: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:06:25.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:06:25 GMT
Content-Length: 538

{ "response": [{
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "configUpdateTime": "1969-12-31T17:00:00-07:00",
  "configApplyTime": "1969-12-31T17:00:00-07:00",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "mid",
  "httpsPort": 443,
  "id": 12,
  "iloIpAddress": "",
  "iloIpGateway": "",
```

(continues on next page)

(continued from previous page)

```

"iloIpNetmask": "",
"iloPassword": "",
"iloUsername": "",
"lastUpdated": "2020-05-19 14:49:39+00",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocation": "Apachecon North America 2018",
"physLocationId": 1,
"profileNames": ["ATS_MID_TIER_CACHE"],
"rack": "",
"revalPending": false,
"revalUpdateTime": "1969-12-31T17:00:00-07:00",
"revalApplyTime": "1969-12-31T17:00:00-07:00",
"status": "REPORTED",
"statusId": 3,
"tcpPort": 80,
"type": "MID",
"typeId": 12,
"updPending": false,
"xmppId": "",
"xmppPasswd": "",
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "172.26.0.4/16",
        "gateway": "172.26.0.1",
        "serviceAddress": true
      }
    ],
    "maxBandwidth": null,
    "monitor": false,
    "mtu": 1500,
    "name": "eth0",
    "routerHostName": "",
    "routerPortName": ""
  }
],
}],
"summary": {
  "count": 13
}}

```

## Summary Fields

The summary object returned by this method of this endpoint uses only the count *standard property*.

### POST

Allows a user to create a new server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:CREATE, SERVER:READ, DELIVERY-SERVICE:READ, CDN:READ, PHYSICAL-LOCATION:READ, CACHE-GROUP:READ, TYPE:READ, PROFILE:READ

**Response Type** Object

### Request Structure

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server shall belong

**cdnId** The integral, unique identifier of the CDN to which the server shall belong

**domainName** The domain part of the server’s FQDN

**hostName** The (short) hostname of the server

**httpsPort** An optional port number on which the server listens for incoming HTTPS connections/requests

**iloIpAddress** An optional IPv4 address of the server’s ILO service Page 1422, 1

**iloIpGateway** An optional IPv4 gateway address of the server’s ILO service Page 1422, 1

**iloIpNetmask** An optional IPv4 subnet mask of the server’s ILO service Page 1422, 1

**iloPassword** An optional string containing the password of the of the server’s ILO service user Page 1422, 1 - displays as simply \*\*\*\*\* if the currently logged-in user does not have the ‘admin’ or ‘operations’ *Role(s)*

**iloUsername** An optional string containing the user name for the server’s ILO service Page 1422, 1

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be necessary, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only

one or two (usually one IPv4 address and one IPv6 address) will be necessary, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**routerPortName** The human-readable name of the router responsible for reaching this server's interface.

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server's interface.

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**physLocationId** An integral, unique identifier for the physical location where the server resides

**profileNames** List of *Name* of the *Profiles* that shall be used by this server

**rack** An optional string indicating "server rack" location

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** An optional port number on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with "HTTP port", as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**typeId** The integral, unique identifier of the 'type' of this server

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router's consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** An optional password used in XMPP communications with the server



969: Request Example

```
POST /api/4.0/servers HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 599
Content-Type: application/json

{
  "cachegroupId": 6,
  "cdnId": 2,
  "domainName": "infra.ciab.test",
  "hostName": "test",
  "httpsPort": 443,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "interfaces": [
    {
      "ipAddresses": [
        {
          "address": "::1",
          "gateway": "::2",
          "serviceAddress": true
        },
        {
          "address": "0.0.0.1/24",
          "gateway": "0.0.0.2",
          "serviceAddress": true
        }
      ],
      "maxBandwidth": null,
      "monitor": true,
      "mtu": 1500,
      "name": "eth0",
      "routerHostName": "",
      "routerPortName": ""
    }
  ],
  "interfaceMtu": 1500,
  "interfaceName": "eth0",
  "ip6Address": "::1",
```

(continues on next page)

(continued from previous page)

```

"ip6Gateway": "::2",
"ipAddress": "0.0.0.1",
"ipGateway": "0.0.0.2",
"ipNetmask": "255.255.255.0",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocationId": 1,
"profileNames": ["ATS_MID_TIER_CACHE"],
"statusId": 3,
"tcpPort": 80,
"typeId": 12
}

```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** The integral, unique identifier of the CDN to which the server belongs

**cdnName** Name of the CDN to which the server belongs

**configUpdateTime** The last time an update was requested for this server. This field defaults to standard epoch

**configApplyTime** The last time an update was applied for this server. This field defaults to standard epoch

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be `null`

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service Page 1422, 1

**iloIpGateway** The IPv4 gateway address of the server's ILO service Page 1422, 1

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service [Page 1422, 1](#)

**iloPassword** The password of the of the server's ILO service user [Page 1422, 1](#) - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' `ROLE(s) <ROLE>`

**iloUsername** The user name for the server's ILO service [Page 1422, 1](#)

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*[Health Protocol](#)*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics

and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**routerPortName** The human-readable name of the router responsible for reaching this server's interface.

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server's interface.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server is in `ADMIN_DOWN` or `OFFLINE` status

**physLocation** The name of the *Physical Location* where the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* where the server resides

**profileNames** List of *Name* of the *Profiles* used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**revalUpdateTime** The last time a content invalidation/revalidation request was submitted for this server. This field defaults to standard epoch

**revalApplyTime** The last time a content invalidation/revalidation request was applied by this server. This field defaults to standard epoch

**rack** A string indicating “server rack” location

**status** The status of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the ‘type’ of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (T3C, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 970: Response Example

```
HTTP/1.1 201 Created
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:34:40 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:34:40 GMT
Content-Length: 562

{ "alerts": [
  {
    "text": "Server created",
    "level": "success"
  }
],
```

(continues on next page)

(continued from previous page)

```
"response": {
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "configUpdateTime": "1969-12-31T17:00:00-07:00",
  "configApplyTime": "1969-12-31T17:00:00-07:00",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "test",
  "httpsPort": 443,
  "id": 14,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "lastUpdated": "2020-05-19 16:34:40+00",
  "mgmtIpAddress": "",
  "mgmtIpGateway": "",
  "mgmtIpNetmask": "",
  "offlineReason": "",
  "physLocation": "Apachecon North America 2018",
  "physLocationId": 1,
  "profileNames": ["ATS_MID_TIER_CACHE"],
  "rack": null,
  "revalPending": false,
  "revalUpdateTime": "1969-12-31T17:00:00-07:00",
  "revalApplyTime": "1969-12-31T17:00:00-07:00",
  "status": "REPORTED",
  "statusId": 3,
  "tcpPort": 80,
  "type": "MID",
  "typeId": 12,
  "updPending": false,
  "xmppId": null,
  "xmppPasswd": null,
  "interfaces": [
    {
      "ipAddresses": [
        {
          "address": "::1",
          "gateway": "::2",
          "serviceAddress": true
        }
      ]
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```

        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": true
    }
],
"maxBandwidth": null,
"monitor": true,
"mtu": 1500,
"name": "eth0",
"routerHostName": "",
"routerPortName": ""
}
]
}}
```

`servers/{HostName-Or-ID}/update`

**POST**

*Queue* or dequeue updates and revalidation updates for a specific server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:UPDATE, SERVER:READ

**Response Type** undefined

## Request Structure

Table 398: Request Path Parameters

Name	Description
HostName-OR-ID	The hostName or integral, unique identifier of the server on which updates are being queued or dequeued

Table 399: Request Query Parameters

Name	Re-quired	Description
updated	no	The value to set for the queue update flag on this server. May be ‘true’ or ‘false’.
reval_updated	no	The value to set for the reval update flag on this server. May be ‘true’ or ‘false’.
con-fig_apply_time	no	The value to set for when a queue update is applied for this server. Must be a valid RFC333Nano timestamp.
revali-date_apply_time	no	The value to set for when a reval update is applied for this server. Must be a valid RFC333Nano timestamp.

**Note:** While none of the timestamps is required individually, at least one must be sent to the API.

#### 971: Request Example

```
POST /api/4.0/servers/my-edge/update?config_apply_time=2022-01-31T12
→%3A00%3A00.123456-07%3A00&revalidate_apply_time=2022-01-31T12%3A00
→%3A00.123456-07%3A00 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 972: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Mon, 10 Dec 2018 18:20:04 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 9Mmo9hIFZyF5gAvfdJD//
→VH9eNgiHVLinXt88H0GLJSHhwND8gMxaFyC+f9XZfiNAoGd1MKi1934ZJGmaIR6qQ==
(continues on next page)
```



(continued from previous page)

Content-Length: 49

```
{
  "alerts" :
  [
    {
      "text" : "successfully set server 'my-edge' config_
→apply_time=2022-01-31T12:00:00.123456-07:00 revalidate_apply_
→time=2022-01-31T12:00:00.123456-07:00",
      "level" : "success"
    }
  ]
}
```

## servers/{hostname}/update\_status

**Note:** This endpoint only truly has meaning for *cache servers*, though it will return a valid response for any server configured in Traffic Ops.

### GET

Retrieves information regarding pending updates and *Content Invalidation Jobs* for a given server

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** SERVER:READ

**Response Type** Array

Changed in version 4.0: Prior to API version 4.0, the response was a top-level array rather than the normal response object.

### Request Structure

Table 400: Request Path Parameters

Name	Description
hostname	The (short) hostname of the server being inspected

### 973: Request Example

```
GET /api/4.0/servers/edge/update_status HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

Each object in the returned array<sup>1</sup> will contain the following fields:

- configUpdateTime** The last time an update was requested for this server. This field defaults to standard epoch
- configApplyTime** The last time an update was applied for this server. This field defaults to standard epoch
- host\_id** The integral, unique identifier for the server for which the other fields in this object represent the pending updates and revalidation status
- host\_name** The (short) hostname of the server for which the other fields in this object represent the pending updates and revalidation status
- parent\_pending** A boolean telling whether or not any *Topology* ancestor or *parent* of this server has pending updates
- parent\_reval\_pending** A boolean telling whether or not any *Topology* ancestor or *parent* of this server has pending *Content Invalidation Jobs*
- reval\_pending** true if the server has pending *Content Invalidation Jobs*, false otherwise
- revalUpdateTime** The last time a content invalidation/revalidation request was submitted for this server. This field defaults to standard epoch
- revalApplyTime** The last time a content invalidation/revalidation request was applied by this server. This field defaults to standard epoch
- status** The name of the status of this server

#### See also:

*Health Protocol* gives more information on how these statuses are used, and the GET method of the *statuses* endpoint can be used to retrieve information about all server statuses configured in Traffic Ops.

- upd\_pending** true if the server has pending updates, false otherwise

---

<sup>1</sup> The returned object is an array, and there is no guarantee that one server exists for a given hostname. However, for each server in the array, that server's update status will be accurate for the server with that particular server ID.

**use\_reval\_pending** A boolean which tells *ORT* whether or not this version of Traffic Ops should use pending *Content Invalidation Jobs*

**Note:** This field was introduced to give *ORT* the ability to work with Traffic Control versions 1.x and 2.x seamlessly - as of Traffic Control v3.0 there is no reason for this field to ever be false.

#### 974: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳R6BjNVrcechGn3eGDqQ1yDiBnEDGQe7QtOMIsRwlpck9SZR8chRQznrkTF3YdROAZ1l8BxR3fXTIvKHIz
    ↳dA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 04 Feb 2019 16:24:01 GMT
Content-Length: 174

{ "response": [{
    "host_name": "edge",
    "upd_pending": false,
    "reval_pending": false,
    "use_reval_pending": true,
    "host_id": 10,
    "status": "REPORTED",
    "parent_pending": false,
    "parent_reval_pending": false,
    "config_update_time": "2022-02-18T13:52:47.129174-07:00",
    "config_apply_time": "2022-02-18T13:52:47.129174-07:00",
    "revalidate_update_time": "2022-02-28T15:44:15.895145-07:00",
    "revalidate_apply_time": "2022-02-18T13:52:47.129174-07:00"
}]}
```

`servers/{ID}`

**PUT**

Allow user to edit a server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:UPDATE, SERVER:READ, DELIVERY-SERVICE:READ, CDN:READ, PHYSICAL-LOCATION:READ, CACHE-GROUP:READ, TYPE:READ, PROFILE:READ

**Response Type** Object

## Request Structure

Table 401: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a server

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server shall belong

**cdnId** The integral, unique identifier of the CDN to which the server shall belong

**domainName** The domain part of the server’s FQDN

**hostName** The (short) hostname of the server

**httpsPort** An optional port number on which the server listens for incoming HTTPS connections/requests

**iloIpAddress** An optional IPv4 address of the server’s ILO service<sup>1</sup>

**iloIpGateway** An optional IPv4 gateway address of the server’s ILO service<sup>Page 1422, 1</sup>

**iloIpNetmask** An optional IPv4 subnet mask of the server’s ILO service<sup>Page 1422, 1</sup>

**iloPassword** An optional string containing the password of the of the server’s ILO service user<sup>Page 1422, 1</sup> - displays as simply \*\*\*\*\* if the currently logged-in user does not have the ‘admin’ or ‘operations’ ROLE(s) <ROLE>

**iloUsername** An optional string containing the user name for the server’s ILO service<sup>Page 1422, 1</sup>

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be necessary, but it is illegal for this set to be an empty collection.

---

<sup>1</sup> For more information see the [Wikipedia page on Lights-Out management](#).

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be necessary, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface. This unsigned integer must not be less than 1280.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**routerPortName** The human-readable name of the router responsible for reaching this server's interface.

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server's interface.

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**physLocationId** An integral, unique identifier for the physical location where the server resides

**profileNames** List of *Name* of the *Profiles* that shall be used by this server

**rack** An optional string indicating "server rack" location

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** An optional port number on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with "HTTP port", as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**typeId** The integral, unique identifier of the 'type' of this server

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router's consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** An optional password used in XMPP communications with the server

975: Request Example

```
PUT /api/4.0/servers/14 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 599
Content-Type: application/json

{
  "cachegroupId": 6,
  "cdnId": 2,
  "domainName": "infra.ciab.test",
  "hostName": "quest",
  "httpsPort": 443,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "interfaces": [
    {
      "ipAddresses": [
        {
          "address": "::1",
          "gateway": "::2",
          "serviceAddress": true
        },
        {
          "address": "0.0.0.1/24",
          "gateway": "0.0.0.2",
          "serviceAddress": false
        }
      ],
      "maxBandwidth": null,
      "monitor": true,
      "mtu": 1500,
      "name": "bond0",
      "routerHostName": "",
      "routerPortName": ""
    }
  ],
  "interfaceMtu": 1500,
  "interfaceName": "eth0",
  "ip6Address": "::1",
```

(continues on next page)

(continued from previous page)

```

"ip6Gateway": "::2",
"ipAddress": "0.0.0.1",
"ipGateway": "0.0.0.2",
"ipNetmask": "255.255.255.0",
"mgmtIpAddress": "",
"mgmtIpGateway": "",
"mgmtIpNetmask": "",
"offlineReason": "",
"physLocationId": 1,
"profileNames": ["ATS_MID_TIER_CACHE"],
"statusId": 3,
"tcpPort": 80,
"typeId": 12
}

```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belongs

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belongs

**cdnId** The integral, unique identifier of the CDN to which the server belongs

**configUpdateTime** The last time an update was requested for this server. This field defaults to standard epoch

**configApplyTime** The last time an update was applied for this server. This field defaults to standard epoch

**cdnName** Name of the CDN to which the server belongs

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

---

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be null

---

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listens for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service Page 1422, 1

**iloIpGateway** The IPv4 gateway address of the server's ILO service Page 1422, 1



**iloIpNetmask** The IPv4 subnet mask of the server's ILO service [Page 1422, 1](#)

**iloPassword** The password of the of the server's ILO service user [Page 1422, 1](#) - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' `ROLE(s) <ROLE>`

**iloUsername** The user name for the server's ILO service [Page 1422, 1](#)

**interfaces** A set of the network interfaces in use by the server. In most scenarios, only one will be present, but it is illegal for this set to be an empty collection.

**ipAddresses** A set of objects representing IP Addresses assigned to this network interface. In most scenarios, only one or two (usually one IPv4 address and one IPv6 address) will be present, but it is illegal for this set to be an empty collection.

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address". It is illegal for a server to not have at least one service address. It is also illegal for a server to have more than one service address of the same address family (i.e. more than one IPv4 service address and/or more than one IPv6 address). Finally, all service addresses for a server must be contained within one interface - which is therefore sometimes referred to as the "service interface" for the server.

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors will consider the entire server unhealthy - which includes *all* configured network interfaces. If this is `null`, it has the meaning "no limit". It has no effect if `monitor` is not true for this interface.

**See also:**

*[Health Protocol](#)*

**monitor** A boolean which describes whether or not this interface should be monitored by Traffic Monitor for statistics

and health consideration.

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information is either not available or not applicable for this interface.

**name** The name of the interface. No two interfaces of the same server may share a name. It is the same as the network interface's device name on the server, e.g. `eth0`.

**routerPortName** The human-readable name of the router responsible for reaching this server's interface.

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server's interface.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server is in `ADMIN_DOWN` or `OFFLINE` status

**physLocation** The name of the *Physical Location* where the server resides

**physLocationId** An integral, unique identifier for the *Physical Location* where the server resides

**profileNames** List of *Name* of the *Profiles* used by this server

**revalPending** A boolean value which, if `true` indicates that this server has pending content invalidation/revalidation

**revalUpdateTime** The last time a content invalidation/revalidation request was submitted for this server. This field defaults to standard epoch

**revalApplyTime** The last time a content invalidation/revalidation request was applied by this server. This field defaults to standard epoch

**rack** A string indicating “server rack” location

**status** The status of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listens for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the ‘type’ of this server

**typeId** The integral, unique identifier of the ‘type’ of this server

**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (*t3c*, formerly ORT)

**xmppId** A system-generated UUID used to generate a server `hashId` for use in Traffic Router’s consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 976: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:46:33.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:46:33 GMT
Content-Length: 566

{ "alerts": [
```

(continues on next page)

(continued from previous page)

```
{
  "text": "Server updated",
  "level": "success"
},
],
"response": {
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "configUpdateTime": "2022-02-28T15:44:15.895145-07:00",
  "configApplyTime": "2022-02-18T13:52:47.129174-07:00",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "quest",
  "httpsPort": 443,
  "id": 14,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "lastUpdated": "2020-05-19 16:46:33+00",
  "mgmtIpAddress": "",
  "mgmtIpGateway": "",
  "mgmtIpNetmask": "",
  "offlineReason": "",
  "physLocation": "Apachecon North America 2018",
  "physLocationId": 1,
  "profileNames": ["ATS_MID_TIER_CACHE"],
  "rack": null,
  "revalPending": false,
  "revalUpdateTime": "1969-12-31T17:00:00-07:00",
  "revalApplyTime": "1969-12-31T17:00:00-07:00",
  "status": "REPORTED",
  "statusId": 3,
  "tcpPort": 80,
  "type": "MID",
  "typeId": 12,
  "updPending": true,
  "xmppId": null,
  "xmppPasswd": null,
  "interfaces": [
    {
      "ipAddresses": [
```

(continues on next page)

(continued from previous page)

```

        "address": "::1",
        "gateway": "::2",
        "serviceAddress": true
    },
    {
        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": false
    }
],
"maxBandwidth": null,
"monitor": true,
"mtu": 1500,
"name": "bond0",
"routerHostName": "",
"routerPortName": ""
}
]
}}
```

## DELETE

Allow user to delete server through api.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:DELETE, SERVER:READ, DELIVERY-SERVICE:READ, CDN:READ, PHYSICAL-LOCATION:READ, CACHE-GROUP:READ, TYPE:READ, PROFILE:READ

**Response Type** Object

Changed in version 3.0: In older versions of the API, this endpoint did not return a response object. It now returns a representation of the deleted server.

## Request Structure

Table 402: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a server

### 977: Request Example

```
DELETE /api/4.0/servers/14 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**cachegroup** A string that is the *name of the Cache Group* to which the server belonged

**cachegroupId** An integer that is the *ID of the Cache Group* to which the server belonged

**cdnId** The integral, unique identifier of the CDN to which the server belonged

**cdnName** Name of the CDN to which the server belonged

**configUpdateTime** The last time an update was requested for this server. This field defaults to standard epoch

**configApplyTime** The last time an update was applied for this server. This field defaults to standard epoch

**domainName** The domain part of the server's FQDN

**guid** An identifier used to uniquely identify the server

**Note:** This is a legacy key which only still exists for compatibility reasons - it should always be null

**hostName** The (short) hostname of the server

**httpsPort** The port on which the server listened for incoming HTTPS connections/requests

**id** An integral, unique identifier for this server

**iloIpAddress** The IPv4 address of the server's ILO service Page 1422, 1

**iloIpGateway** The IPv4 gateway address of the server's ILO service Page 1422, 1

**iloIpNetmask** The IPv4 subnet mask of the server's ILO service Page 1422, 1

**iloPassword** The password of the of the server's ILO service user Page 1422, 1 - displays as simply `*****` if the currently logged-in user does not have the 'admin' or 'operations' *Role(s)*

**iloUsername** The user name for the server's ILO service Page 1422, 1

**interfaces** A set of the network interfaces that were in use by the server

**ipAddresses** A set of objects representing IP Addresses that were assigned to this network interface

**address** The actual IP address, including any mask as a CIDR-notation suffix

**gateway** Either the IP address of the network gateway for this address, or `null` to signify that no such gateway exists

**serviceAddress** A boolean that describes whether or not the server's main service is available at this IP address. When this property is `true`, the IP address is referred to as a "service address".

**maxBandwidth** The maximum healthy bandwidth allowed for this interface. If bandwidth exceeds this limit, Traffic Monitors would have considered the entire server unhealthy - which includes *all* configured network interfaces. If this was `null`, it has the meaning "no limit". It had no effect if `monitor` was not true for this interface.

**See also:**

*Health Protocol*

**monitor** A boolean which describes whether or not this interface should have been monitored by Traffic Monitor for statistics and health consideration

**mtu** The MTU of this interface. If it is `null`, it may be assumed that the information was either not available or not applicable for this interface.

**name** The name of the interface. It is the same as the network interface's device name on the server, e.g. `eth0`.

**routerPortName** The human-readable name of the router responsible for reaching this server's interface.

**routerPortName** The human-readable name of the port used by the router responsible for reaching this server's interface.

**lastUpdated** The date and time at which this server description was last modified

**mgmtIpAddress** The IPv4 address of some network interface on the server that was used for 'management'

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpGateway** The IPv4 address of a gateway used by some network interface on the server that was used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**mgmtIpNetmask** The IPv4 subnet mask used by some network interface on the server that was used for ‘management’

Deprecated since version 3.0: This field is deprecated and will be removed in a future API version. Operators should migrate this data into the `interfaces` property of the server.

**offlineReason** A user-entered reason why the server was in ADMIN\_DOWN or OFFLINE status

**physLocation** The name of the physical location where the server resided

**physLocationId** An integral, unique identifier for the physical location where the server resided

**profileNames** List of *Name* of the *Profiles* which was used by this server

**revalPending** A boolean value which, if true indicates that this server has pending content invalidation/revalidation

**revalUpdateTime** The last time a content invalidation/revalidation request was submitted for this server. This field defaults to standard epoch

**revalApplyTime** The last time a content invalidation/revalidation request was applied by this server. This field defaults to standard epoch

**rack** A string indicating “server rack” location

**status** The *Status* of the server

**See also:**

*Health Protocol*

**statusId** The integral, unique identifier of the status of this server

**See also:**

*Health Protocol*

**tcpPort** The port on which this server listened for incoming TCP connections

---

**Note:** This is typically thought of as synonymous with “HTTP port”, as the port specified by `httpsPort` may also be used for incoming TCP connections.

---

**type** The name of the *Type* of this server

**typeId** The integral, unique identifier of the ‘type’ of this server



**updPending** A boolean value which, if `true`, indicates that the server has updates of some kind pending, typically to be acted upon by Traffic Control Cache Config (T3C, formerly ORT)

**xmppId** A system-generated UUID used to generate a server hashId for use in Traffic Router's consistent hashing algorithm. This value is set when a server is created and cannot be changed afterwards.

**xmppPasswd** The password used in XMPP communications with the server

#### 978: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 19 May 2020 17:50:13 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Tue, 19 May 2020 16:50:13 GMT
Content-Length: 568

{ "alerts": [
  {
    "text": "Server deleted",
    "level": "success"
  }
],
"response": {
  "cachegroup": "CDN_in_a_Box_Mid",
  "cachegroupId": 6,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "configUpdateTime": "1969-12-31T17:00:00-07:00",
  "configApplyTime": "1969-12-31T17:00:00-07:00",
  "domainName": "infra.ciab.test",
  "guid": null,
  "hostName": "quest",
  "httpsPort": 443,
  "id": 14,
  "iloIpAddress": "",
  "iloIpGateway": "",
  "iloIpNetmask": "",
  "iloPassword": "",
  "iloUsername": "",
  "lastUpdated": "2020-05-19 16:46:33+00",
  "mgmtIpAddress": "",
  "mgmtIpGateway": "",
  "mgmtIpNetmask": "",
```

(continues on next page)

(continued from previous page)

```

"offlineReason": "",
"physLocation": "Apachecon North America 2018",
"physLocationId": 1,
"profileNames": ["ATS_MID_TIER_CACHE"],
"rack": null,
"revalPending": false,
"revalUpdateTime": "1969-12-31T17:00:00-07:00",
"revalApplyTime": "1969-12-31T17:00:00-07:00",
"status": "REPORTED",
"statusId": 3,
"tcpPort": 80,
"type": "MID",
"typeId": 12,
"updPending": false,
"xmppId": null,
"xmppPasswd": null,
"interfaces": [
  {
    "ipAddresses": [
      {
        "address": "0.0.0.1/24",
        "gateway": "0.0.0.2",
        "serviceAddress": false
      },
      {
        "address": "::1",
        "gateway": "::2",
        "serviceAddress": true
      }
    ],
    "maxBandwidth": null,
    "monitor": true,
    "mtu": 1500,
    "name": "bond0",
    "routerHostName": "",
    "routerPortName": ""
  }
]
}
}

```

`servers/{{ID}}/deliveryservices`

GET

Retrieves all *Delivery Services* assigned to a specific server.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** DELIVERY-SERVICE:READ, SERVER:READ

**Response Type** Array

## Request Structure

Table 403: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the server for which assigned <i>Delivery Services</i> shall be listed

Table 404: Request Query Parameters

Name	Re- quired	Description
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

979: Request Example

```
GET /api/4.0/servers/10/deliveryservices HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

<sup>1</sup> Only the *Delivery Services* visible to the requesting user’s *Tenant* will appear, regardless of their *Role* or the *Delivery Services*’ actual ‘server assignment’ status.

## Response Structure

**active** A boolean that defines *Active*.

**anonymousBlockingEnabled** A boolean that defines *Anonymous Blocking*

**ccrDnsTtl** The *DNS TTL* - named “ccrDnsTtl” for legacy reasons

**cdnId** The integral, unique identifier of the *CDN* to which the *Delivery Service* belongs

**cdnName** Name of the *CDN* to which the *Delivery Service* belongs

**checkPath** A *Check Path*

**consistentHashRegex** A *Consistent Hashing Regular Expression*

**consistentHashQueryParams** An array of *Consistent Hashing Query Parameters*

**deepCachingType** The *Deep Caching* setting for this *Delivery Service*

**displayName** The *Display Name*

**dnsBypassCname** A *DNS Bypass CNAME*

**dnsBypassIp** A *DNS Bypass IP*

**dnsBypassIp6** A *DNS Bypass IPv6*

**dnsBypassTtl** The *DNS Bypass TTL*

**dscp** A *DSCP* to be used within the *Delivery Service*

**ecsEnabled** A boolean that defines the *EDNS0 Client Subnet Enabled* setting on this *Delivery Service*

**edgeHeaderRewrite** A set of *Edge Header Rewrite Rules*

**exampleURLs** An array of *Example URLs*

**firstHeaderRewrite** A set of *First Header Rewrite Rules*

**fqPacingRate** The *Fair-Queuing Pacing Rate Bps*

**geoLimit** An integer that defines the *Geo Limit*

**geoLimitCountries** An array of strings defining the *Geo Limit Countries*

**geoLimitRedirectUrl** A *Geo Limit Redirect URL*

**geoProvider** The *Geolocation Provider*

**globalMaxMbps** The *Global Max Mbps*

**globalMaxTps** The *Global Max TPS*

**httpBypassFqdn** A *HTTP Bypass FQDN*

**id** An integral, unique identifier for this *Delivery Service*

**infoUrl** An *Info URL*

- initialDispersion** The *Initial Dispersion*
- innerHeaderRewrite** A set of *Inner Header Rewrite Rules*
- ipv6RoutingEnabled** A boolean that defines the *IPv6 Routing Enabled* setting on this *Delivery Service*
- lastHeaderRewrite** A set of *Last Header Rewrite Rules*
- lastUpdated** The date and time at which this *Delivery Service* was last updated, in :rfc:3339 format
- Changed in version 4.0: Prior to API version 4.0, this property used *Traffic Ops's Custom Date/Time Format*.
- logsEnabled** A boolean that defines the *Logs Enabled* setting on this *Delivery Service*
- longDesc** The *Long Description* of this *Delivery Service*
- matchList** The *Delivery Service's Match List*
- pattern** A regular expression - the use of this pattern is dependent on the **type** field (backslashes are escaped)
  - setNumber** An integer that provides explicit ordering of *Match List* items - this is used as a priority ranking by Traffic Router, and is not guaranteed to correspond to the ordering of items in the array.
  - type** The type of match performed using pattern.
- maxDnsAnswers** The *Max DNS Answers* allowed for this *Delivery Service*
- maxOriginConnections** The *Max Origin Connections*
- midHeaderRewrite** A set of *Mid Header Rewrite Rules*
- missLat** The *Geo Miss Default Latitude* used by this *Delivery Service*
- missLong** The *Geo Miss Default Longitude* used by this *Delivery Service*
- multiSiteOrigin** A boolean that defines the use of *Use Multi-Site Origin Feature* by this *Delivery Service*
- orgServerFqdn** The *Origin Server Base URL*
- originShield** A *Origin Shield* string
- profileDescription** The *Description* of the *Profile* with which this *Delivery Service* is associated
- profileId** The *ID* of the *Profile* with which this *Delivery Service* is associated
- profileName** The *Name* of the *Profile* with which this *Delivery Service* is associated
- protocol** An integral, unique identifier that corresponds to the *Protocol* used by this *Delivery Service*

- qstringIgnore** An integral, unique identifier that corresponds to the *Query String Handling* setting on this *Delivery Service*
- rangeRequestHandling** An integral, unique identifier that corresponds to the *Range Request Handling* setting on this *Delivery Service*
- regexRemap** A *Regex Remap Expression*
- regionalGeoBlocking** A boolean defining the *Regional Geoblocking* setting on this *Delivery Service*
- remapText** *Raw Remap Text*
- signed** true if and only if *signingAlgorithm* is not null, false otherwise
- signingAlgorithm** Either a *Signing Algorithm* or null to indicate URL/URI signing is not implemented on this *Delivery Service*
- rangeSliceBlockSize** An integer that defines the byte block size for the ATS Slice Plugin. It can only and must be set if *rangeRequestHandling* is set to 3.
- sslKeyVersion** This integer indicates the *SSL Key Version*
- tenantId** The integral, unique identifier of the *Tenant* who owns this *Delivery Service*
- tlsVersions** A list of explicitly supported *TLS Versions*
- New in version 4.0.
- topology** The unique name of the *Topology* that this *Delivery Service* is assigned to
- trRequestHeaders** If defined, this defines the *Traffic Router Log Request Headers* used by Traffic Router for this *Delivery Service*
- trResponseHeaders** If defined, this defines the *Traffic Router Additional Response Headers* used by Traffic Router for this *Delivery Service*
- type** The *Type* of this *Delivery Service*
- typeId** The integral, unique identifier of the *Type* of this *Delivery Service*
- xmlId** This *Delivery Service*'s *xml\_id*

#### 980: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
```

(continues on next page)

(continued from previous page)

```
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 08 Jun 2021 01:15:07 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: R04tVfDdqx0rEU9BqlRmvsYXmVgVNkivqr6LhJlMulFR+1bLGivP8z93jy3N9bejcMdQw11RwJojM3Mbw
X-Server-Name: traffic_ops_golang/
Date: Tue, 08 Jun 2021 00:15:07 GMT
Content-Length: 806
```

```
{ "response": [{
  "active": false,
  "anonymousBlockingEnabled": false,
  "ccrDnsTtl": 3600,
  "cdnId": 2,
  "cdnName": "CDN-in-a-Box",
  "checkPath": null,
  "consistentHashQueryParams": [],
  "consistentHashRegex": null,
  "deepCachingType": "NEVER",
  "displayName": "test",
  "dnsBypassCname": null,
  "dnsBypassIp": null,
  "dnsBypassIp6": null,
  "dnsBypassTtl": null,
  "dscp": 0,
  "ecsEnabled": false,
  "edgeHeaderRewrite": null,
  "exampleURLs": [
    "http://cdn.test.mycdn.ciab.test"
  ],
  "firstHeaderRewrite": null,
  "fqPacingRate": null,
  "geoLimit": 0,
  "geoLimitCountries": null,
  "geoLimitRedirectURL": null,
  "geoProvider": 0,
  "globalMaxMbps": null,
  "globalMaxTps": null,
  "httpBypassFqdn": null,
  "id": 7,
  "infoUrl": null,
  "initialDispersion": 1,
  "innerHeaderRewrite": null,
  "ipv6RoutingEnabled": true,
  "lastHeaderRewrite": null,
  "lastUpdated": "2021-06-08T00:14:04.959292Z",
```

(continues on next page)

(continued from previous page)

```
"logsEnabled": false,
"longDesc": "Apachecon North America 2018",
"matchList": [
  {
    "type": "HOST_REGEX",
    "setNumber": 0,
    "pattern": ".*\\.test\\.\\.\\.*"
  }
],
"maxDnsAnswers": null,
"maxOriginConnections": 0,
"maxRequestHeaderBytes": 0,
"midHeaderRewrite": null,
"missLat": 41.881944,
"missLong": -87.627778,
"multiSiteOrigin": false,
"originShield": null,
"orgServerFqdn": "http://origin.infra.ciab.test",
"profileDescription": null,
"profileId": null,
"profileName": null,
"protocol": 0,
"qstringIgnore": 0,
"rangeRequestHandling": 0,
"rangeSliceBlockSize": null,
"regexRemap": null,
"regionalGeoBlocking": false,
"remapText": null,
"routingName": "cdn",
"serviceCategory": null,
"signed": false,
"signingAlgorithm": null,
"sslKeyVersion": null,
"tenant": "root",
"tenantId": 1,
"tlsVersions": null,
"topology": null,
"trResponseHeaders": null,
"trRequestHeaders": null,
"type": "HTTP",
"typeId": 1,
"xmlId": "test"
}]}
```



## POST

Assign an arbitrary number of *Delivery Services* to a single server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** DELIVERY-SERVICE:READ, SERVER:READ,  
DELIVERY-SERVICE:UPDATE, SERVER:UPDATE

**Response Type** Object

## Request Structure

Table 405: Request Path Parameters

Name	Re- quired	Description
ID	Yes	The integral, unique identifier of the server that you want to assign delivery services to.

Table 406: Request Query Parameters

Name	Re- quired	Description
re- place	Yes	Whether the list of <i>Delivery Services</i> you provide should replace the existing list or be merged with the existing list. Must be a 1, or true, or 0, or false.

The request body is an array of IDs of *Delivery Services* that you want to assign to the server. The array can be empty, but it must be provided.

### 981: Request Example

```
POST /api/4.0/servers/6/deliveryservices?replace=1 HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 3

[
  1
]
```

## Response Structure

**dsIds** An array of integral, unique identifiers for *Delivery Services* which the request added to server. If `:replace:` is false, *Delivery Services* that are already assigned will remain, though they are not listed by `:dsIds:`.

**replace** The `:replace:` value you provided in the body of the request, or null if none was provided.

**serverId** The server's integral, unique identifier

### 982: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 09:08:32
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳iV+JzAZSsmlxRZsNtIRg3oA9470hAwRMpq5xhcYVi0Y831Trx2YRlsyhYp0PqHg5+QPoXHGf0nx8uso0f
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 08:08:32 GMT
Content-Length: 129

{
  "alerts": [
    {
      "text": "successfully assigned dses to server",
      "level": "success"
    }
  ],
  "response": {
    "serverId": 6,
    "dsIds": [
      1
    ],
    "replace": true
  }
}
```

## servers/{{ID}}/queue\_update

**Caution:** In the vast majority of cases, it is advisable that the PUT method of the `servers/{{ID}}` endpoint be used instead.

### POST

*Queue* or dequeue updates for a specific server.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:QUEUE, SERVER:READ

**Response Type** Object

### Request Structure

Table 407: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the server on which updates are being queued or dequeued

**action** A string describing what action to take regarding server updates; one of:

**queue** *Queue Updates* for the server, propagating configuration changes to the actual server

**dequeue** Cancels any pending updates on the server

#### 983: Request Example

```
POST /api/4.0/servers/13/queue_update HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 22
Content-Type: application/json

{
  "action": "dequeue"
}
```

## Response Structure

**action** The action processed, one of:

**queue** *Queue Updates* was performed on the server, propagating configuration changes to the actual server

**dequeue** Canceled any pending updates on the server

**serverId** The integral, unique identifier of the server on which action was taken

### 984: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Mon, 10 Dec 2018 18:20:04 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 9Mmo9hIFZyF5gAvfdJD//
→VH9eNgiHVLinXt88H0GlJSHhwND8gMxaFyC+f9XZfiNAoGd1MKi1934ZJGmaIR6qQ==
Content-Length: 49

{
  "response": {
    "serverId": "13",
    "action": "dequeue"
  }
}
```

## servers/{ID}/status

### PUT

Updates server status and queues updates on all descendant *Topology* nodes or child caches if server type is EDGE or MID. Also, captures offline reason if status is set to ADMIN\_DOWN or OFFLINE and prepends offline reason with the user that initiated the status change.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:UPDATE, SERVER:READ, STATUS:READ

**Response Type** undefined

## Request Structure

Table 408: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the server whose status is being changed

**offlineReason** A string containing the reason for the status change

**status** The name or integral, unique identifier of the server's new status

985: Request Example

```
PUT /api/4.0/servers/13/status HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 56
Content-Type: application/json

{
  "status": "ADMIN_DOWN",
  "offlineReason": "Bad drives"
}
```

## Response Structure

986: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Mon, 10 Dec 2018 18:08:44 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
  ↳LS1jCo5eMVKxmeYDol0I2LgLYazocSggR5hynNoLcPmMov9u2s3ulksPdQtG1N3aS+VM9tdMsCrahFPral
```

(continues on next page)

(continued from previous page)

```
Content-Length: 158

{ "alerts": [
  {
    "level": "success",
    "text": "Updated status [ ADMIN_DOWN ] for quest.infra.ciab.
→test [ admin: Bad drives ] and queued updates on all child caches"
  }
]}
```

## service\_categories

### GET

Get all requested *Service Categories*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** SERVICE-CATEGORY:READ

**Response Type** Array

## Request Structure

Table 409: Request Query Parameters

Name	Description
name	Filter for <i>Service Categories</i> with this name
order-by	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the n <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 987: Request Example

```
GET /api/4.0/service_categories?name=SERVICE_CATEGORY_NAME HTTP/1.1
Host: trafficops.infra.ciab.test
```

(continues on next page)

(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**name** This *Service Category*'s name

**lastUpdated** The date and time at which this *Service Category* was last modified,  
in *Traffic Ops's Custom Date/Time Format*

### 988: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Yzr6TfhxgpZ3pbrrr4TRG4wC3PlnHDDzgs2igtz/
    ↳1ppLSy2MzugqaGW4y5yzwzl5T3+7q6HWej7GQZt1XIVeZQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 11 Mar 2020 20:02:47 GMT
Content-Length: 102

{
  "response": [
    {
      "lastUpdated": "2020-03-04 15:46:20-07",
      "name": "SERVICE_CATEGORY_NAME"
    }
  ]
}
```

## POST

Create a new service category.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVICE-CATEGORY:CREATE, SERVICE-CATEGORY:READ

**Response Type** Object

## Request Structure

**name** This *Service Category*’s name

989: Request Example

```
POST /api/4.0/service_categories HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 48
Content-Type: application/json

{
  "name": "SERVICE_CATEGORY_NAME",
}
```

## Response Structure

**name** This *Service Category*’s name

**lastUpdated** The date and time at which this *Service Category* was last modified, in *Traffic Ops*’s *Custom Date/Time Format*

990: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)



(continued from previous page)

```
Whole-Content-Sha512: +pJm4c30+JTaSXNt+LP+u240Ba/
→SsvSSDOQ4rDc6hcyZ0FIL+iY/WWrMHhpLulRGKGY88bM4YPCMaxGn3FZ9yQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 11 Mar 2020 20:12:20 GMT
Content-Length: 154

{
  "alerts": [
    {
      "text": "serviceCategory was created.",
      "level": "success"
    }
  ],
  "response": {
    "lastUpdated": "2020-03-11 14:12:20-06",
    "name": "SERVICE_CATEGORY_NAME"
  }
}
```

`service_categories/{name}`

**PUT**

Update a service category.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVICE-CATEGORY:UPDATE, SERVICE-CATEGORY:READ

**Response Type** Object

## Request Structure

**name** The *Service Category*’s new name

Table 410: Request Path Parameters

Name	Description
name	The current name of the <i>Service Category</i>

#### 991: Request Example

```
PUT /api/4.0/service_categories/sc-name HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 48
Content-Type: application/json

{
  "name": "New Name",
}
```

### Response Structure

**name** This *Service Category*'s name

**lastUpdated** The date and time at which this *Service Category* was last modified, in *Traffic Ops's Custom Date/Time Format*

#### 992: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: +pJm4c30+JTaSXNt+LP+u240Ba/
→SsvSSDOQ4rDc6hcyZ0FIL+iY/WWrMHhpLulRGKGY88bM4YPCMaxGn3FZ9yQ==
X-Server-Name: traffic_ops_golang/
Date: Wed, 11 Mar 2020 20:12:20 GMT
Content-Length: 189

{
  "alerts": [
    {
      "text": "Service Category was updated.",
      "level": "success"
    }
  ],
  "response": {
```

(continues on next page)

(continued from previous page)

```

    "lastUpdated": "2020-03-11 14:12:20-06",
    "name": "New Name"
  }
}

```

## DELETE

Deletes a specific *Service Category*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVICE-CATEGORY:DELETE, SERVICE-CATEGORY:READ

**Response Type** undefined

## Request Structure

Table 411: Request Path Parameters

Name	Description
name	The current name of the <i>Service Category</i> to be deleted

### 993: Request Example

```

DELETE /api/4.0/service_categories/my-service-category HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0

```

## Response Structure

### 994: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE

```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 17 Aug 2020 16:13:31.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: yErJobzG9IA0khvqZQK+Yi7X4pFVvOqxn6PjrdzN5DnKVM/
→K8Kka3REul1XmKJnMXVRY8RayoEVDm16mBFfe4Q==
X-Server-Name: traffic_ops_golang/
Date: Mon, 17 Aug 2020 15:13:31 GMT
Content-Length: 103

{
  "alerts": [
    {
      "text": "my-service-category was deleted.",
      "level": "success"
    }
  ]
}
```

## snapshot

### PUT

Performs a CDN *Snapshot*. Effectively, this propagates the new *configuration* of the CDN to its *operating state*, which replaces the output of the `cdns/{{name}}/snapshot` endpoint with the output of the `cdns/{{name}}/snapshot/new` endpoint. This also changes the output of the `cdns/{{name}}/configs/monitoring` endpoint since that endpoint returns the latest monitoring information from the *operating state*.

---

**Note:** By default, snapshotting the CDN also deletes all HTTPS certificates for every *Delivery Service* which has been deleted since the last *Snapshot*. In order to disable this behavior, set `disable_auto_cert_deletion` in `cdn.conf` to true.

---

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** CDN-SNAPSHOT:CREATE, CDN-SNAPSHOT:READ

**Response Type** undefined

## Request Structure

Table 412: Request Query Parameters

Name	Description
cdn	The name of the CDN for which a <i>Snapshot</i> shall be taken
cdnID	The id of the CDN for which a <i>Snapshot</i> shall be taken

**Note:** At least one query parameter must be given.

### 995: Request Example

```
PUT /api/4.0/snapshot?cdn=CDN-in-a-Box HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 996: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳gmaWI0tKgNFPY00zMrLCGDosBJkPbeIvW4BH9tEh96VjBqyWqzjgPySoMa3ViM1BQXA6VAUOGmc76VyhB
X-Server-Name: traffic_ops_golang/
Date: Wed, 18 Mar 2020 15:51:48 GMT
Content-Length: 47

{
  "response": "SUCCESS"
}
```

## sslkey\_expirations

### GET

Retrieves SSL certificate expiration information.

**Auth. Required** Yes

**Roles Required** “admin”

**Permissions Required** ACME:READ

**Response Type** Array

### Request Structure

Table 413: Request Query Parameters

Name	Re-quired	Description
days	no	Return only the expiration information for SSL certificates expiring in the next given number of days.

#### 997: Request Example

```
GET /api/4.0/sslkey_expirations?days=30 HTTP/1.1
Host: trafficops.infra.ciab.test
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

**deliveryservice** The *xml\_id* for the *Delivery Service* corresponding to this SSL certificate.

**cdn** The ID for the CDN corresponding to this SSL certificate.

**provider** The provider of this SSL certificate, generally the name of the Certificate Authority or the ACME account.

**expiration** The expiration date of this SSL certificate.

**federated** A boolean indicating if this SSL certificate is use in a federated *Delivery Service*.

998: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 07 Jun 2021 22:52:20
    ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Mon, 07 Jun 2021 21:52:20 GMT
Content-Length: 384

{ "response": [
  {
    "deliveryservice": "foo1",
    "cdn": "cdn1",
    "provider": "Self Signed",
    "expiration": "2022-08-02T15:38:06-06:00",
    "federated": false
  },
  {
    "deliveryservice": "foo2",
    "cdn": "cdn2",
    "provider": "Lets Encrypt",
    "expiration": "2022-07-12T12:14:00-06:00",
    "federated": true
  }
]}
```

## staticdnsentries

### GET

Retrieve all static DNS entries configured within Traffic Control

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STATIC-DN:READ, CACHE-GROUP:READ,  
DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

Table 414: Request Query Parameters

Name	Re-quired	Description
ad- dress	no	Return only static DNS entries that operate on this address/CNAME
cachegroup	no	Return only static DNS entries assigned to the <i>Cache Group</i> that has this <i>Name</i>
cachegroupId	no	Return only static DNS entries assigned to the <i>Cache Group</i> that has this <i>ID</i>
deliv- ery- ser- vice	no	Return only static DNS entries that apply within the domain of the <i>Delivery Service</i> with this <i>xml_id</i>
deliv- ery- servi- ceId	no	Return only static DNS entries that apply within the domain of the <i>Delivery Service</i> identified by this integral, unique identifier
host	no	Return only static DNS entries that resolve this FQDN
id	no	Return only the static DNS entry with this integral, unique identifier
ttl	no	Return only static DNS entries with this TTL
type	no	Return only static DNS entries of this type
typeId	no	Return only static DNS entries of the type identified by this integral, unique identifier
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n<sup>th</sup></i> page of results, where “n” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .



#### 999: Request Example

```
GET /api/4.0/staticdnsentries?address=foo.bar HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**address** If `typeId` identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which `host` shall be resolved

**cachegroup** An optional string containing the *Name of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryservice** The name of a *Delivery Service* under the domain of which this static DNS entry shall be active

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If `typeId` identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the Fully Qualified Domain Name (FQDN) which shall resolve to `address`

**id** An integral, unique identifier for this static DNS entry

**ttl** The TTL of this static DNS entry in seconds

**type** The name of the type of this static DNS entry

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

## 1000: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳Px1zTH3ihg+hfmADGcap0Juuud39fGsx5Y3CzqaFNmRwFu1ZLMz0sy0EN2pb7vp0tpI6/
    ↳zeIUYAC3dbSbw0mA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:04:33 GMT
Content-Length: 226

{ "response": [
  {
    "address": "foo.bar.",
    "cachegroup": null,
    "cachegroupId": null,
    "deliveryservice": "demo1",
    "deliveryserviceId": 1,
    "host": "test",
    "id": 2,
    "lastUpdated": "2018-12-10 19:59:56+00",
    "ttl": 300,
    "type": "CNAME_RECORD",
    "typeId": 41
  }
]}

```

## POST

Creates a new, static DNS entry.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** STATIC-DN:CREATE, STATIC-DN:READ, CACHE-GROUP:READ, DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

**address** If `typeId` identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which `host` shall be resolved

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If `typeId` identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the FQDN which shall resolve to `address`

**ttl** The TTL of this static DNS entry in seconds

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

### 1001: Request Example

```
POST /api/4.0/staticdnsentries HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 92
Content-Type: application/json

{
  "address": "test.quest.",
  "deliveryserviceId": 1,
  "host": "test",
  "ttl": 300,
  "typeId": 41
}
```

## Response Structure

**address** If `typeId` identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which `host` shall be resolved

**cachegroup** An optional string containing the *Name of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryservice** The name of a *Delivery Service* under the domain of which this static DNS entry shall be active

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If `typeId` identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the Fully Qualified Domain Name (FQDN) which shall resolve to `address`

**id** An integral, unique identifier for this static DNS entry

**ttl** The TTL of this static DNS entry in seconds

**type** The name of the *Type* of this static DNS entry

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

### 1002: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 8dcJyjw2NJZx0L90z16P7g/
→7j5A1jlpYiY6Y+rRVQ2wGcwYI3yiGPrz6ur0qKzgqEBBsh8aPF44WTHAR9jUJdg==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 19:54:19 GMT
Content-Length: 282

{ "alerts": [
  {
    "text": "staticDNSEntry was created.",
    "level": "success"
  }
],
"response": {
  "address": "test.quest.",
  "cachegroup": null,
  "cachegroupId": null,
  "deliveryservice": null,
  "deliveryserviceId": 1,
  "host": "test",
  "id": 2,
  "lastUpdated": "2018-12-10 19:54:19+00",
  "ttl": 300,
  "type": "CNAME_RECORD",
  "typeId": 41
}}
```

## PUT

Updates a static DNS entry.

**Auth. Required** Yes

**Role(s) Required** “admin” or “operator”

**Permissions Required** STATIC-DN:UPDATE, STATIC-DN:READ, CACHE-GROUP:READ, DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

Table 415: Request Query Parameters

Name	Description
id	The integral, unique identifier of the static DNS entry to modify

**address** If typeId identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which host shall be resolved

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If typeId identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the Fully Qualified Domain Name (FQDN) which shall resolve to address

**ttd** The TTL of this static DNS entry in seconds

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

### 1003: Request Example

```
PUT /api/4.0/staticdnsentries?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 89
Content-Type: application/json

{
  "address": "foo.bar.",
  "deliveryserviceId": 1,
  "host": "test",
  "ttl": 300,
  "typeId": 41
}
```

## Response Structure

**address** If typeId identifies a CNAME type record, this is the Canonical Name (CNAME) of the server with a trailing period, otherwise it is the IP address to which host shall be resolved

**cachegroup** An optional string containing the *Name of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**cachegroupId** An optional, integer that is the *ID of a Cache Group* which will service this static DNS entry

---

**Note:** This field has no effect, and is not used by any part of Traffic Control. It exists for legacy compatibility reasons.

---

**deliveryservice** The name of a *Delivery Service* under the domain of which this static DNS entry shall be active

**deliveryserviceId** The integral, unique identifier of a *Delivery Service* under the domain of which this static DNS entry shall be active

**host** If typeId identifies a CNAME type record, this is an alias for the CNAME of the server, otherwise it is the FQDN which shall resolve to address

**id** An integral, unique identifier for this static DNS entry

**ttl** The TTL of this static DNS entry in seconds

**type** The name of the *Type* of this static DNS entry

**typeId** The integral, unique identifier of the *Type* of this static DNS entry

#### 1004: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳+FaYmpnlIIzVSBq0nosw29NZcV9xFhlVgWuUqXUyiDihVUSzX4jrdAlORDgzDvKsYQB8LSkPdGHwt1zjg
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 19:59:56 GMT
Content-Length: 279

{ "alerts": [
  {
    "text": "staticDNSEntry was updated.",
    "level": "success"
  }
],
"response": {
  "address": "foo.bar.",
  "cachegroup": null,
  "cachegroupId": null,
```

(continues on next page)

(continued from previous page)

```
"deliveryservice": null,
"deliveryserviceId": 1,
"host": "test",
"id": 2,
"lastUpdated": "2018-12-10 19:59:56+00",
"ttl": 300,
"type": "CNAME_RECORD",
"typeId": 41
}}
```

## DELETE

Delete staticdnsentries.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** STATIC-DN:DELETE, STATIC-DN:READ,  
DELIVERY-SERVICE:READ, CACHE-GROUP:READ

**Response Type** undefined

## Request Structure

Table 416: Request Query Parameters

Name	Description
id	The integral, unique identifier of the static DNS entry to delete



#### 1005: Request Example

```
DELETE /api/4.0/staticdnsentries?id=2 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

#### 1006: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳g6uqHPU44LuTtqU2ahtazrVCpcpNWVc9kWJQ0YRuiVLDnsm39KOB/xt3XM6j0/
    ↳X3WYiIawnNspkxRC85LJHwFA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:05:52 GMT
Content-Length: 69

{ "alerts": [
  {
    "text": "staticDNSEntry was deleted.",
    "level": "success"
  }
]}
```

### stats\_summary

#### GET

Either retrieve a list of summary stats or the timestamp of the latest recorded stats summary.

What is returned is driven by the query parameter `lastSummaryDate`.

If the parameter is set it will return an object with the latest timestamp, else an array of summary stats will be returned.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STAT:READ,                      CDN:READ,                      DELIVERY-SERVICE:READ

**Response Type** Array or Object

## Request Structure

### Summary Stats

Table 417: Request Query Parameters

Name	Re- quired	Description
deliveryServiceName	no	Return only summary stats that were reported for <i>Delivery Service</i> with the given display name
cdnName	no	Return only summary stats that were reported for CDN with the given name
statName	no	Return only summary stats that were reported for given stat name
orderBy	no	Choose the ordering of the results - can only be one of deliveryServiceName, statName or cdnName
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

#### 1007: Request Example

```
GET /api/4.0/stats_summary HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Last Updated Summary Stat

Table 418: Request Query Parameters

Name	Required	Description
statName	no	Get latest updated date for the given stat
lastSummaryDate	yes	Tells route to get only latest updated timestamp

#### 1008: Request Example

```
GET /api/4.0/stats_summary?lastSummaryDate=true HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### Summary Stats

**cdnName** The CDN name for which the summary stat was taken for

---

**Note:** If the `cdn` is equal to `all` it represents `summary_stats` across all delivery services across all CDNs

---

**deliveryServiceName** The *Delivery Service* display name for which the summary stat was taken for

---

**Note:** If the `deliveryServiceName` is equal to `all` it represents `summary_stats` across all delivery services within the given CDN

---

**statName** Stat name summary stat represents

**statValue** Summary stat value

**summaryTime** Timestamp of summary, in *Traffic Ops's Custom Date/Time Format*

**statDate** Date stat was taken, in YYYY-MM-DD format

#### 1009: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGs1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
    ↳3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": [
  {
    "cdnName": "CDN-in-a-Box",
    "deliveryServiceName": "all",
    "statName": "daily_maxgbps",
    "statValue": 5,
    "summaryTime": "2019-11-19 00:04:06+00",
    "statDate": "2019-11-19"
  },
  {
    "cdnName": "CDN-in-a-Box",
    "deliveryServiceName": "all",
    "statName": "daily_maxgbps",
    "statValue": 3,
    "summaryTime": "2019-11-18 00:04:06+00",
    "statDate": "2019-11-18"
  },
  {
    "cdnName": "CDN-in-a-Box",
    "deliveryServiceName": "all",
    "statName": "daily_bytesserved",
    "statValue": 1000,
    "summaryTime": "2019-11-19 00:04:06+00",
    "statDate": "2019-11-19"
  }
]}
```

## Last Updated Summary Stat

**summaryTime** Timestamp of the last updated summary, in *Traffic Ops's Custom Date/Time Format*

1010: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGs1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
→3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": {
  "summaryTime": "2019-11-19 00:04:06+00"
}}
```

## POST

Post a stats summary for a given stat.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STAT:CREATE, STAT:READ, CDN:READ,  
DELIVERY-SERVICE:READ

**Response Type** Object

## Request Structure

**cdnName** The CDN name for which the summary stat was taken for

---

**Note:** If the `cdn` is equal to `all` it represents `summary_stats` across all delivery services across all CDNs

---

**deliveryServiceName** The *Delivery Service* display name for which the summary stat was taken for

---

**Note:** If the `deliveryServiceName` is equal to `all` it represents summary\_stats across all delivery services within the given CDN

---

**statName** Stat name summary stat represents

**statValue** Summary stat value

**summaryTime** Timestamp of summary, in *Traffic Ops's Custom Date/Time Format*

**statDate** Date stat was taken, in YYYY-MM-DD format

---

**Note:** `statName`, `statValue` and `summaryTime` are required. If `cdnName` and `deliveryServiceName` are not given they will default to `all`.

---

### 1011: Request Example

```
POST /api/4.0/stats_summary HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 113
Content-Type: application/json

{
  "cdnName": "CDN-in-a-Box",
  "deliveryServiceName": "all",
  "statName": "daily_maxgbps",
  "statValue": 10,
  "summaryTime": "2019-12-05 00:03:57+00",
  "statDate": "2019-12-05"
}
```

## Response Structure

### 1012: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)

(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→ezxk+iP7o7KE7zpWmGc0j8nz5k+1wAzY0HiNiA2xswTQrt+N+6CgQqUV2r9G1HAsPNr0HF2PhYs/
→Xr7DrY0w0A==
X-Server-Name: traffic_ops_golang/
Date: Thu, 06 Dec 2018 02:14:45 GMT
Content-Length: 97

{ "alerts": [
  {
    "text": "Stats Summary was successfully created",
    "level": "success"
  }
]}
```

## statuses

### GET

Retrieves a list of all server *Statuses*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STATUS:READ

**Response Type** Array

## Request Structure

Table 419: Request Query Parameters

Name	Required	Description
description	no	Return only <i>Statuses</i> with this <i>exact</i> description
id	no	Return only the <i>Status</i> with this integral, unique identifier
name	no	Return only <i>Statuses</i> with this name
orderBy	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n</i> <sup>th</sup> page of results, where “n” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

1013: Request Example

```
GET /api/4.0/statuses?name=REPORTED HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**description** A short description of the status

**id** The integral, unique identifier of this status

**lastUpdated** The date and time at which this status was last modified, in *Traffic Ops’s Custom Date/Time Format*

**name** The name of the status

1014: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
```

(continues on next page)



(continued from previous page)

```
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGs1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
→3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": [
  {
    "description": "Server is online and reported in the health.
→protocol.",
    "id": 3,
    "lastUpdated": "2018-12-10 19:11:17+00",
    "name": "REPORTED"
  }
]}
```

## POST

Creates a Server *Status*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STATUS:CREATE, STATUS:READ

**Response Type** Array

## Request Structure

**description** Create a *Status* with this description

**name** Create a *Status* with this name

1015: Request Example

```
POST /api/4.0/statuses HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

(continued from previous page)

```
{ "description": "test", "name": "example" }
```

## Response Structure

**description** A short description of the status

**id** The integral, unique identifier of this status

**lastUpdated** The date and time at which this status was last modified, in *Traffic Ops's Custom Date/Time Format*

**name** The name of the status

### 1016: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGS1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
    ↳3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 167

{ "alerts": [
  {
    "text": "status was created.",
    "level": "success"
  }
], "response": [
  {
    "description": "test",
    "name": "example"
    "id": 3,
    "lastUpdated": "2018-12-10 19:11:17+00",
  }
]}
```

`statuses/{{ID}}`

**GET**

Retrieves a list of all server *Statuses*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STATUS:READ

**Response Type** Array

## Request Structure

Table 420: Request Query Parameters

Name	Required	Description
<code>description</code>	no	Return only <i>Statuses</i> with this <i>exact</i> description
<code>id</code>	no	Return only the <i>Status</i> with this integral, unique identifier
<code>name</code>	no	Return only <i>Statuses</i> with this name
<code>orderby</code>	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the <code>response</code> array
<code>sortOrder</code>	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
<code>limit</code>	no	Choose the maximum number of results to return
<code>offset</code>	no	The number of results to skip before beginning to return results. Must use in conjunction with <code>limit</code>
<code>page</code>	no	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are <code>limit</code> long and the first page is 1. If <code>offset</code> was defined, this query parameter has no effect. <code>limit</code> must be defined to make use of <code>page</code> .

#### 1017: Request Example

```
GET /api/4.0/statuses/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

- description** A short description of the status
- id** The integral, unique identifier of this status
- lastUpdated** The date and time at which this status was last modified, in *Traffic Ops's Custom Date/Time Format*
- name** The name of the status

#### 1018: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: dHNip9kpTGGS1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
→3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 150

{ "response": [
  {
    "description": "Server is online and reported in the health
→protocol.",
    "id": 3,
    "lastUpdated": "2018-12-10 19:11:17+00",
    "name": "REPORTED"
  }
]}
```

## PUT

Updates a *Status*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STATUS:UPDATE, STATUS:READ

**Response Type** Array

## Request Structure

**description** The description of the updated *Status*

**name** The name of the updated *Status*

1019: Request Example

```
POST /api/4.0/statuses/3 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...

{ "description": "test", "name": "example" }
```

## Response Structure

**description** A short description of the status

**id** The integral, unique identifier of this status

**lastUpdated** The date and time at which this status was last modified, in *Traffic Ops's Custom Date/Time Format*

**name** The name of the status

1020: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
➔Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
➔GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Whole-Content-Sha512: dHNip9kpTGGs1w39/fWcFehNktgmXZus8XaufnmDpv0PyG/
↪3fK/KfoC03Z0j9V74/CCffps7doEygWeL/xRtKA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 20:56:59 GMT
Content-Length: 167

{ "alerts": [
  {
    "text": "status was created.",
    "level": "success"
  }
], "response": [
  {
    "description": "test",
    "name": "example"
    "id": 3,
    "lastUpdated": "2018-12-10 19:11:17+00",
  }
]}
```

## steering

### GET

Gets a list of all *Steering Targets* in the Traffic Ops database.

**Auth. Required** Yes

**Roles Required** “Portal”, “Steering”, “Federation”, “operations” or “admin”

**Permissions Required** STEERING:READ, DELIVERY-SERVICE:READ

**Response Type** Array

## Request Structure

No parameters available.

### 1021: Request Example

```
GET /api/4.0/steering HTTP/1.1
User-Agent: python-requests/2.22.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**clientSteering** Whether this is a *client steering* Delivery Service.

**targets** The delivery services that the *Steering Delivery Service* targets.

**order** If this is a *STEERING\_ORDER* target, this is the value of the order. Otherwise, 0.

**weight** If this is a *STEERING\_WEIGHT* target, this is the value of the weight. Otherwise, 0.

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**filters** Filters of type *STEERING\_REGEX* that exist on either of the targets.

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**pattern** A regular expression - the use of this pattern is dependent on the *type* field (backslashes are escaped)

### 1022: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 24 Feb 2020 18:56:57
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳hcJa4xVLDx7bxBmoSjYo5YUwdSBWQr9GlgqRYrc6ZU7LeenjiV3go22YlIHt/
    ↳GtjLcHQjJ5DulKRhdsvFMq7Fng==
X-Server-Name: traffic_ops_golang/
Date: Mon, 24 Feb 2020 17:56:57 GMT
Content-Length: 167

{
  "response": [
    {
      "deliveryService": "steering1",
      "clientSteering": true,
      "targets": [
        {
```

(continues on next page)

(continued from previous page)

```

        "order": 0,
        "weight": 1,
        "deliveryService": "demo1"
      },
      {
        "order": 0,
        "weight": 2,
        "deliveryService": "demo2"
      }
    ],
    "filters": [
      {
        "deliveryService": "demo1",
        "pattern": ".*\\.demo1\\.\\.*)"
      },
      {
        "deliveryService": "demo2",
        "pattern": ".*\\.demo2*\\.\\.*)"
      }
    ]
  }
]
}

```

## steering/{{ID}}/targets

### GET

Get all targets for a steering *Delivery Service*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** STEERING:READ, DELIVERY-SERVICE:READ,  
TYPE:READ

**Response Type** Array



## Request Structure

Table 421: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i> for which targets shall be listed

Table 422: Request Query Parameters

Name	Description
target	Return only the target mappings that target the <i>Delivery Service</i> identified by this integral, unique identifier
orderby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the <i>n</i> <sup>th</sup> page of results, where “ <i>n</i> ” is the value of this parameter, pages are <i>limit</i> long and the first page is 1. If <i>offset</i> was defined, this query parameter has no effect. <i>limit</i> must be defined to make use of <i>page</i> .

### 1023: Request Structure

```
GET /api/4.0/steering/2/targets?target=1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

- deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*
- deliveryServiceId** An integral, unique identifier for the steering *Delivery Service*
- target** A string that is the *xml\_id* of this target *Delivery Service*
- targetId** An integral, unique identifier for this target *Delivery Service*
- type** The steering type of this target *Delivery Service*. This should be one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT
- typeId** An integral, unique identifier for the *steering type* of this target *Delivery Service*

**value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ attributed to this steering target as an integer

#### 1024: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→utlJK4oYS2l6Ff7NzAqRuQeMEtazYn3rM3Nlux2XgTLxvSyslHy0mJrwDExSU05gVMdrgYCLZrZEvPH1E
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 14:09:23 GMT
Content-Length: 130

{ "response": [
  {
    "deliveryService": "test",
    "deliveryServiceId": 2,
    "target": "demo1",
    "targetId": 1,
    "type": "STEERING_ORDER",
    "typeId": 44,
    "value": 100
  }
]}
```

## POST

Create a steering target.

**Auth. Required** Yes

**Roles Required** Portal, Steering, Federation, “operations” or “admin”

**Permissions Required** STEERING:CREATE, STEERING:READ, DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** Object

## Request Structure

Table 423: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i> to which a target shall be added

**targetId** The integral, unique identifier of a *Delivery Service* which shall be a new steering target for the *Delivery Service* identified by the ID path parameter

**typeId** The integral, unique identifier of the steering type of the new target *Delivery Service*. This should be corresponding to one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT

**value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ which shall be attributed to the new target *Delivery Service*

### 1025: Request Example

```
POST /api/4.0/steering/2/targets HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 43
Content-Type: application/json

{
  "targetId": 1,
  "value": 100,
  "typeId": 43
}
```

## Response Structure

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**deliveryServiceId** An integral, unique identifier for the steering *Delivery Service*

**target** A string that is the *xml\_id* of this target *Delivery Service*

**targetId** An integral, unique identifier for this target *Delivery Service*

**type** The steering type of this target *Delivery Service*. This should be one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT

**typeId** An integral, unique identifier for the *steering type* of this target *Delivery Service*

**value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ attributed to this steering target as an integer

#### 1026: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
→+dTvfzrnOhdwAOMmY28r0+gFV5z+3aABI2FfAMziTYcU+pZrDanrJzMXpKWIL5Q/
→oCUBZpJDRt9hRCFkT4oGYw==
X-Server-Name: traffic_ops_golang/
Date: Mon, 10 Dec 2018 21:22:17 GMT
Content-Length: 196

{ "alerts": [
  {
    "text": "steeringtarget was created.",
    "level": "success"
  }
],
"response": {
  "deliveryService": "test",
  "deliveryServiceId": 2,
  "target": "demo1",
  "targetId": 1,
  "type": "HTTP",
  "typeId": 1,
  "value": 100
}}
```

`steering/{ID}/targets/{targetID}`

**PUT**

Updates a steering target.

**Auth. Required** Yes

**Roles Required** Portal, Steering, Federation, “operations” or “admin”

**Permissions Required** STEERING:UPDATE, STEERING:READ, DELIVERY-SERVICE:READ, TYPE:READ

Response Type Object

## Request Structure

Table 424: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i>
targetID	The integral, unique identifier of a <i>Delivery Service</i> which is a target of the <i>Delivery Service</i> identified by ID

**typeId** The integral, unique identifier of the *steering type* of the target *Delivery Service*. This should be corresponding to one of STEERING\_WEIGHT, STEERING\_ORDER, STEERING\_GEO\_ORDER or STEERING\_GEO\_WEIGHT

**value** The ‘weight’, ‘order’, ‘geo\_order’ or ‘geo\_weight’ which shall be attributed to the target *Delivery Service*

1027: Request Example

```
PUT /api/4.0/steering/2/targets/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 26
Content-Type: application/json

{
  "value": 1,
  "typeId": 43
}
```

## Response Structure

**deliveryService** A string that is the *xml\_id* of the steering *Delivery Service*

**deliveryServiceId** An integral, unique identifier for the steering *Delivery Service*

**target** A string that is the *xml\_id* of this target *Delivery Service*

**targetId** An integral, unique identifier for this target *Delivery Service*

**type** The steering type of this target *Delivery Service*

**typeId** An integral, unique identifier for the *steering type* of this target *Delivery Service*

**value** The ‘weight’ attributed to this steering target as an integer

## 1028: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳AfXsIRzdtU3HZYkr93qBMVTZRJ5oTF2u5sKYnd+DSqxZ+RQxY6vXtCupnnXCf9dxMt5QXRW1EFOW/
    ↳FBG6lFrTg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 14:34:22 GMT
Content-Length: 194

{ "alerts": [
  {
    "text": "steeringtarget was updated.",
    "level": "success"
  }
],
"response": {
  "deliveryService": "test",
  "deliveryServiceId": 2,
  "target": "demo1",
  "targetId": 1,
  "type": "HTTP",
  "typeId": 1,
  "value": 1
}}
```

## DELETE

Removes a specific target mapping from a specific *Delivery Service*

**Auth. Required** Yes

**Roles Required** Portal, Steering, Federation, “operations” or “admin”

**Permissions Required** STEERING:DELETE, STEERING:READ, DELIVERY-SERVICE:READ, TYPE:READ

**Response Type** undefined

## Request Structure

Table 425: Request Path Parameters

Name	Description
ID	The integral, unique identifier of a steering <i>Delivery Service</i> - a target of which shall be deleted
targetID	The integral, unique identifier of a <i>Delivery Service</i> which is a target to be removed of the <i>Delivery Service</i> identified by ID

1029: Request Example

```
DELETE /api/4.0/steering/2/targets/1 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

1030: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳N6h8Kl7uQveqpTc3fmKXFDY2yYe5smApNcaTow4ab0DHGFdJfqQh89I4nvvaXvmVNhxVAqX3UE/
    ↳6blb08/9Xqg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 14:42:54 GMT
Content-Length: 69

{ "alerts": [
  {
    "text": "steeringtarget was deleted.",
    "level": "success"
  }
]}
```

### system/info

#### GET

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** None

**Response Type** Object

### Request Structure

No parameters available.

### Response Structure

**parameters** An object containing information about the Traffic Ops server

---

**Note:** These are all the *Parameters* in *The GLOBAL Profile*, so the keys below are merely those present by default required for Traffic Control to operate

---

**default\_geo\_miss\_latitude** The default latitude used when geographic lookup of an IP address fails

**default\_geo\_miss\_longitude** The default longitude used when geographic lookup of an IP address fails

**tm.instance\_name** The name of the Traffic Ops instance; typically used when multiple instances are active

**tm.toolname** The name of the Traffic Ops tool (usually “Traffic Ops”) - used in several API endpoints and written in comment headers on most Apache Traffic Server (ATS) configuration files generated by Traffic Ops

**tm.url** The URL for this Traffic Ops instance ) - used in several API endpoints and written in comment headers on most Apache Traffic Server (ATS) configuration files generated by Traffic Ops

**use\_reval\_pending** A string containing an integer which represents a boolean value (hold your applause); one of:

“0” Do not use pending revalidations - this effectively prohibits the use of “Content Invalidation Jobs”

“1” Use pending revalidations - this effectively enables the use of “Content Invalidation Jobs”



### 1031: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ObxOXk1jrC1/
    ↳JtrqElUICceyx9iJKJxZydEIHvAU7khTTQwt0QGvS04ELdkdrbu3ctFo3pf3NMaMM9tAkNokg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 19:06:01 GMT
Content-Length: 285

{ "response": {
  "parameters": {
    "default_geo_miss_latitude": "0",
    "default_geo_miss_longitude": "-1",
    "tm.instance_name": "CDN-In-A-Box",
    "tm.toolname": "Traffic Ops",
    "tm.url": "https://trafficops.infra.ciab.test:443/",
    "use_reval_pending": "0"
  }
}}
```

## tenants

### GET

Get all requested *Tenants*.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** TENANT:READ

**Response Type** Array

## Request Structure

Table 426: Request Query Parameters

Name	Description
active	If true, return only active <i>Tenants</i> ; if false return only inactive <i>Tenants</i>
id	Return only <i>Tenants</i> with this integral, unique identifier
name	Return only <i>Tenants</i> with this name
orderby	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	Choose the maximum number of results to return
offset	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	Return the $n^{\text{th}}$ page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

1032: Request Example

```
GET /api/4.0/tenants?name=root HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

**active** A boolean which indicates whether or not the *Tenant* is active

**id** The integral, unique identifier of this *Tenant*

**name** This *Tenant*’s name

**parentId** The integral, unique identifier of this *Tenant*’s parent

**parentName** The name of the parent of this *Tenant*

1033: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: Yzr6TfhxgpZ3pbrrr4TRG4wC3PlnHDDzgs2igtz/
→1ppLSy2MzugqaGW4y5yzwzl5T3+7q6HWej7GQZt1XIVeZQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 19:57:58 GMT
Content-Length: 106

{ "response": [
  {
    "id": 1,
    "name": "root",
    "active": true,
    "lastUpdated": "2018-12-10 19:11:17+00",
    "parentId": null
  }
]}
```

## POST

Create a new tenant.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TENANT:CREATE, TENANT:READ

**Response Type** Object

## Request Structure

**active** An optional boolean - default: false - which indicates whether or not the tenant shall be immediately active

**name** The name of the tenant

**parentId** The integral, unique identifier of the parent of this tenant

1034: Request Example

```
POST /api/4.0/tenants HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

(continues on next page)

(continued from previous page)

```
Content-Length: 48
Content-Type: application/json

{
  "active": true,
  "name": "test",
  "parentId": 1
}
```

## Response Structure

**active** A boolean which indicates whether or not the tenant is active

**id** The integral, unique identifier of this tenant

**name** This tenant's name

**parentId** The integral, unique identifier of this tenant's parent

### 1035: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ysdopC//
→JQI79BRUa61s6M2HzHxYHpo5Rdcuau0oqCYxiV0oUhNZfOVydVkv8zDN2qA374XKnym4kWj3VzQIXg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 19:37:16 GMT
Content-Length: 162

{ "alerts": [
  {
    "text": "tenant was created.",
    "level": "success"
  }
],
"response": {
  "id": 9,
  "name": "test",
  "active": true,
  "lastUpdated": "2018-12-11 19:37:16+00",
```

(continues on next page)

(continued from previous page)

```
"parentId": 1
}}
```

tenants/{ID}

**PUT**

Updates a specific tenant.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TENANT:UPDATE, TENANT:READ

**Response Type** Object

## Request Structure

Table 427: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the tenant being modified

**active** An optional boolean - default: false - which indicates whether or not the tenant shall be immediately active

**name** The name of the tenant

**parentId** The integral, unique identifier of the parent of this tenant

### 1036: Request Example

```
PUT /api/4.0/tenants/9 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 59
Content-Type: application/json

{
  "active": true,
  "name": "quest",
  "parentId": 3
}
```

## Response Structure

**active** A boolean which indicates whether or not the tenant is active

**id** The integral, unique identifier of this tenant

**name** This tenant's name

**parentId** The integral, unique identifier of this tenant's parent

### 1037: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: 5soYQFrG2x5ZJ1e5UZIOLUv/
→928qyd2Bfgw21Wv85rqjLpyeT3djkrVD1/xpKConulNrZs2czJKrrwZA7X61w==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 20:30:54 GMT
Content-Length: 163

{ "alerts": [
  {
    "text": "tenant was updated.",
    "level": "success"
  }
],
"response": {
  "id": 9,
  "name": "quest",
  "active": true,
  "lastUpdated": "2018-12-11 20:30:54+00",
  "parentId": 3
}}
```

## DELETE

Deletes a specific tenant.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TENANT:DELETE, TENANT:READ

**Response Type** undefined

## Request Structure

Table 428: Request Path Parameters

Name	Description
ID	The integral, unique identifier for the tenant being deleted

### 1038: Request Example

```
DELETE /api/4.0/tenants/9 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

## Response Structure

### 1039: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: KU0XIbFoD0Cy06kzH2G159pBqie/
    ↳TEFJgh33mssGNwXJZ1RkTLaSTHT8Df4X+pOs7UauZH10akGvaA0UTiN/vg==
X-Server-Name: traffic_ops_golang/
Date: Tue, 11 Dec 2018 20:40:31 GMT
Content-Length: 61
```

(continues on next page)

(continued from previous page)

```
{ "alerts": [
  {
    "text": "tenant was deleted.",
    "level": "success"
  }
]}
```

## topologies

### GET

Retrieves *Topologies*.

**Auth. Required** Yes

**Roles Required** “read-only”

**Permissions Required** TOPOLOGY:READ, CACHE-GROUP:READ

**Response Type** Array

## Request Structure

Table 429: Request Query Parameters

Name	Required	Description
name	no	Return the <i>Topology</i> with this name



#### 1040: Request Example

```
GET /api/4.0/topologies HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

### Response Structure

- description** A short sentence that describes the *Topology*.
- lastUpdated** The date and time at which this *Topology* was last updated, in ISO-like format
- name** The name of the *Topology*. This can only be letters, numbers, and dashes.
- nodes** An array of nodes in the *Topology*
  - cachegroup** The name of a *Cache Group*
  - parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

#### 1041: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:22:32
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳1F4MCJCinuQWz0f1LAAZBrzbuPVsHrNn2BtTozRZojEjGpm76IsXBQK5Q0wSwBoHac+D0C1Z3p7M8kdjc
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:22:32 GMT
Content-Length: 205

{
  "response": [
    {
      "description": "This is my topology",
      "name": "my-topology",
```

(continues on next page)

(continued from previous page)

```
"nodes": [
  {
    "cachegroup": "edge1",
    "parents": [
      7
    ]
  },
  {
    "cachegroup": "edge2",
    "parents": [
      7,
      8
    ]
  },
  {
    "cachegroup": "edge3",
    "parents": [
      8,
      9
    ]
  },
  {
    "cachegroup": "edge4",
    "parents": [
      9
    ]
  },
  {
    "cachegroup": "mid1",
    "parents": []
  },
  {
    "cachegroup": "mid2",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid3",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid4",
    "parents": [
```

(continues on next page)

(continued from previous page)

```

        5
      ]
    },
    {
      "cachegroup": "mid5",
      "parents": [
        5,
        6
      ]
    },
    {
      "cachegroup": "mid6",
      "parents": [
        6
      ]
    }
  ],
  "lastUpdated": "2020-04-13 17:12:34+00"
}
]
}

```

## POST

Create a new *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TOPOLOGY:CREATE, TOPOLOGY:READ, CACHE-GROUP:READ

**Response Type** Object

## Request Structure

**description** A short sentence that describes the topology.

**name** The name of the topology. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group* with at least 1 server in it

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

1042: Request Example

```
POST /api/4.0/topologies HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 924
Content-Type: application/json

{
  "name": "my-topology",
  "description": "This is my topology",
  "nodes": [
    {
      "cachegroup": "edge1",
      "parents": [
        7
      ]
    },
    {
      "cachegroup": "edge2",
      "parents": [
        7,
        8
      ]
    },
    {
      "cachegroup": "edge3",
      "parents": [
        8,
        9
      ]
    },
    {
      "cachegroup": "edge4",
      "parents": [
        9
      ]
    },
    {
      "cachegroup": "mid1",
      "parents": []
    },
    {
      "cachegroup": "mid2",
```

(continues on next page)

(continued from previous page)

```

        "parents": [
            4
        ]
    },
    {
        "cachegroup": "mid3",
        "parents": [
            4
        ]
    },
    {
        "cachegroup": "mid4",
        "parents": [
            5
        ]
    },
    {
        "cachegroup": "mid5",
        "parents": [
            5,
            6
        ]
    },
    {
        "cachegroup": "mid6",
        "parents": [
            6
        ]
    }
]
}

```

## Response Structure

**description** A short sentence that describes the topology.

**lastUpdated** The date and time at which this *Topology* was last updated, in ISO-like format

**name** The name of the topology. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group*

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

### 1043: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:12:34
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: ftNcDRjYCDMkQM+o/
    ↳szayKZriQZHGpcT0vNY0HpKgy88i0pXeEEeLGbUPh6LXtK7TvL76EgGECTzvCkcm+2LVA==
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:12:34 GMT
Content-Length: 239

{
  "alerts": [
    {
      "text": "topology was created.",
      "level": "success"
    }
  ],
  "response": {
    "description": "This is my topology",
    "name": "my-topology",
    "nodes": [
      {
        "cachegroup": "edge1",
        "parents": [
          7
        ]
      },
      {
        "cachegroup": "edge2",
        "parents": [
          7,
          8
        ]
      },
      {
        "cachegroup": "edge3",
        "parents": [
          8,
          9
        ]
      }
    ]
  }
}

```

(continues on next page)

(continued from previous page)

```

    ]
  },
  {
    "cachegroup": "edge4",
    "parents": [
      9
    ]
  },
  {
    "cachegroup": "mid1",
    "parents": []
  },
  {
    "cachegroup": "mid2",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid3",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid4",
    "parents": [
      5
    ]
  },
  {
    "cachegroup": "mid5",
    "parents": [
      5,
      6
    ]
  },
  {
    "cachegroup": "mid6",
    "parents": [
      6
    ]
  }
],
"lastUpdated": "2020-04-13 17:12:34+00"
}

```

(continues on next page)

(continued from previous page)

```
}

```

## PUT

Updates a specific *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TOPOLOGY:UPDATE, TOPOLOGY:READ, CACHE-GROUP:READ

**Response Type** Object

## Request Structure

Table 430: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Topology</i> to be updated

**description** A short sentence that describes the *Topology*.

**name** The name of the *Topology*. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group* with at least 1 server in it

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

### 1044: Request Example

```
PUT /api/4.0/topologies?name=my-topology HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 853
Content-Type: application/json

{
  "name": "my-topology",
  "description": "The description is updated, too",

```

(continues on next page)



(continued from previous page)

```
"nodes": [
  {
    "cachegroup": "edge1",
    "parents": [
      6
    ]
  },
  {
    "cachegroup": "edge2",
    "parents": [
      6,
      7
    ]
  },
  {
    "cachegroup": "edge3",
    "parents": [
      7,
      8
    ]
  },
  {
    "cachegroup": "edge4",
    "parents": [
      8
    ]
  },
  {
    "cachegroup": "mid2",
    "parents": []
  },
  {
    "cachegroup": "mid3",
    "parents": []
  },
  {
    "cachegroup": "mid4",
    "parents": [
      4
    ]
  },
  {
    "cachegroup": "mid5",
    "parents": [
      4,
      5
    ]
  }
]
```

(continues on next page)

(continued from previous page)

```

    ]
  },
  {
    "cachegroup": "mid6",
    "parents": [
      5
    ]
  }
]
}

```

## Response Structure

**description** A short sentence that describes the *Topology*.

**lastUpdated** The date and time at which this *Topology* was last updated, in ISO-like format

**name** The name of the *Topology*. This can only be letters, numbers, and dashes.

**nodes** An array of nodes in the *Topology*

**cachegroup** The name of a *Cache Group*

**parents** The indices of the parents of this node in the nodes array, 0-indexed. 2 parents max

### 1045: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:33:13
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: WV0tsoOVrEWcVjWM2TmT5DXy/
→a5Q0ygTZEQRhbkHHUmz9dgVLK1F5Joc9jtKA8gZu8/eM5+Tqqguh3mzrhAy/Q==
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:33:13 GMT
Content-Length: 237

{
  "alerts": [
    {

```

(continues on next page)

(continued from previous page)

```

        "text": "topology was updated.",
        "level": "success"
    }
],
"response": {
    "description": "The description is updated, too",
    "name": "my-topology",
    "nodes": [
        {
            "cachegroup": "edge1",
            "parents": [
                6
            ]
        },
        {
            "cachegroup": "edge2",
            "parents": [
                6,
                7
            ]
        },
        {
            "cachegroup": "edge3",
            "parents": [
                7,
                8
            ]
        },
        {
            "cachegroup": "edge4",
            "parents": [
                8
            ]
        },
        {
            "cachegroup": "mid2",
            "parents": []
        },
        {
            "cachegroup": "mid3",
            "parents": []
        },
        {
            "cachegroup": "mid4",
            "parents": [
                4
            ]
        }
    ]
}

```

(continues on next page)

(continued from previous page)

```

    ]
  },
  {
    "cachegroup": "mid5",
    "parents": [
      4,
      5
    ]
  },
  {
    "cachegroup": "mid6",
    "parents": [
      5
    ]
  }
],
"lastUpdated": "2020-04-13 17:33:13+00"
}

```

## DELETE

Deletes a specific *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TOPOLOGY:DELETE, TOPOLOGY:READ, CACHE-GROUP:READ

**Response Type** undefined

## Request Structure

Table 431: Request Query Parameters

Name	Required	Description
name	yes	The name of the <i>Topology</i> to be deleted

1046: Request Example

```

DELETE /api/4.0/topologies?name=my-topology HTTP/1.1
User-Agent: python-requests/2.23.0
Accept-Encoding: gzip, deflate

```

(continues on next page)

(continued from previous page)

```
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

1047: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 13 Apr 2020 18:35:32
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: yErJobzG9IA0khvqZQK+Yi7X4pFVvOqxn6PjrdzN5DnKVm/
    ↳K8Kka3REul1XmKJnMXVRY8RayoEVDm16mBFfe4Q==
X-Server-Name: traffic_ops_golang/
Date: Mon, 13 Apr 2020 17:35:32 GMT
Content-Length: 87

{
  "alerts": [
    {
      "text": "topology was deleted.",
      "level": "success"
    }
  ]
}
```

**topologies/{name}/queue\_update**

**POST**

*Queue* or “dequeue” updates for all servers assigned to the *Cache Groups* in a specific *Topology*.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** SERVER:QUEUE,  
SERVER:READ, CACHE-GROUP:READ

TOPOLOGY:READ,

**Response Type** Object

## Request Structure

Table 432: Request Path Parameters

Name	Description
name	The name of the <i>Topology</i> on which to queue or dequeue updates.

**action** One of “queue” or “dequeue” as appropriate

**cdnId** The integral, unique identifier for the CDN on which to (de)queue updates

1048: Request Example

```
POST /api/4.0/topologies/demo1-top/queue_update HTTP/1.1
User-Agent: python-requests/2.24.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 28

{
  "action": "queue",
  "cdnId": 1
}
```

## Response Structure

**action** The action processed, either "queue" or "dequeue"

**cdnId** The CDN ID on which *Queue Updates* was performed or cleared

**topology** The name of the *Topology* on which *Queue Updates* was performed or cleared

1049: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
```

(continues on next page)

(continued from previous page)

```
Content-Encoding: gzip
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 08 Sep 2020 17:35:42.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→nmu3TMVmlcHeEstLuiqPsEypNV2jcs5PyrqsqJKkexxxb8N7qk84AWzTJWUpsfdVWrj/
→JzRiCPGJS4hw0phQ==
X-Server-Name: traffic_ops_golang/
Date: Tue, 08 Sep 2020 16:35:42 GMT
Content-Length: 79

{
  "response": {
    "action": "queue",
    "cdnId": 1,
    "topology": "demo1-top"
  }
}
```

## types

### GET

Retrieves all of the *Types* of things configured in Traffic Ops. Yes, that is as specific as a description of a ‘type’ can be.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** TYPE:READ

**Response Type** Array

## Request Structure

Table 433: Request Query Parameters

Name	Re- quired	Description
id	no	Return only the type that is identified by this integral, unique identifier
name	no	Return only types with this name
useInTable	no	Return only types that are used to identify the type of the object stored in the Traffic Ops database table that has this name

#### 1050: Request Structure

```
GET /api/4.0/types?name=TC_LOC HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
```

### Response Structure

- description** A short description of this type
- id** An integral, unique identifier for this type
- lastUpdated** The date and time at which this type was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of this type
- useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type

#### 1051: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpgT3YRyKJ/
→p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 12 Dec 2018 22:59:22 GMT
Content-Length: 168

{ "response": [
  {
    "id": 48,
    "lastUpdated": "2018-12-12 16:26:41+00",
    "name": "TC_LOC",
    "description": "Location for Traffic Control Component Servers
→",
    "useInTable": "cachegroup"
```

(continues on next page)



(continued from previous page)

```
}
[]}
```

## POST

Creates a type

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TYPE:CREATE, TYPE:READ

**Response Type** Object

## Request Structure

**description** A short description of this type

**name** The name of this type

**useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type.

---

**Note:** The only useInTable value that is allowed to be created dynamically is ‘server’

---

### 1052: Request Structure

```
POST /api/4.0/type HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 67
Content-Type: application/json

{
  "name": "Example01",
  "description": "Example",
  "useInTable": "server"
}
```

## Response Structure

- description** A short description of this type
- id** An integral, unique identifier for this type
- lastUpdated** The date and time at which this type was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of this type
- useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type

### 1053: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo80rCu79Tz9xpgT3YRyKJ/
    ↳p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 Feb 2020 18:58:41 GMT
Content-Length: 171

{
  "alerts": [
    {
      "text": "type was created.",
      "level": "success"
    }
  ],
  "response": [
    {
      "id": 3004,
      "lastUpdated": "2020-02-26 18:58:41+00",
      "name": "Example01",
      "description": "Example"
      "useInTable": "server"
    }
  ]
}
```

types/{ID}

## PUT

Updates a type

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TYPE:UPDATE, TYPE:READ

**Response Type** Object

## Request Structure

Table 434: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the type being updated

**description** A short description of this type

**name** The name of this type

**useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type.

---

**Note:** Only types with useInTable set to ‘server’ are allowed to be updated.

---

### 1054: Request Structure

```
PUT /api/4.0/type/3004 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 68
Content-Type: application/json

{
  "name": "Example01",
  "description": "Example2",
  "useInTable": "server"
}
```

## Response Structure

- description** A short description of this type
- id** An integral, unique identifier for this type
- lastUpdated** The date and time at which this type was last updated, in *Traffic Ops's Custom Date/Time Format*
- name** The name of this type
- useInTable** The name of the Traffic Ops database table that contains objects which are grouped, identified, or described by this type

### 1055: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo80rCu79Tz9xpgT3YRyKJ/
    ↳p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 Feb 2020 18:58:41 GMT
Content-Length: 172

{
  "alerts": [
    {
      "text": "type was updated.",
      "level": "success"
    }
  ],
  "response": [
    {
      "id": 3004,
      "lastUpdated": "2020-02-26 18:58:41+00",
      "name": "Example02",
      "description": "Example"
      "useInTable": "server"
    }
  ]
}
```

## DELETE

Deletes a type

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** TYPE:DELETE, TYPE:READ

**Response Type** Object

## Request Structure

Table 435: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the type being deleted

**Note:** Only types with useInTable set to “server” are allowed to be deleted.

### 1056: Request Structure

```
DELETE /api/4.0/type/3004 HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 0
```

## Response Structure

### 1057: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
→GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: EH8jo8OrCu79Tz9xpgT3YRyKJ/
→p2NcTmbS3huwtqRByHz9H6qZLQjA59RIPaVSq3ZxsU6QhTaox5nBkQ9LPSAA==
```

(continues on next page)

(continued from previous page)

```
X-Server-Name: traffic_ops_golang/
Date: Wed, 26 Feb 2020 18:58:41 GMT
Content-Length: 84

{
  "alerts": [
    {
      "text": "type was deleted.",
      "level": "success"
    }
  ],
}
```

## user/current

### GET

**Caution:** As a username is needed to log in, any administrator or application must necessarily know the current username at any given time. Thus it's generally better to use the username query parameter of a GET request to *users* instead.

Retrieves the details of the authenticated user.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** None

**Response Type** Object

## Request Structure

No parameters available.

### 1058: Request Example

```
GET /api/4.0/user/current HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

- addressLine1** The user's address - including street name and number
- addressLine2** An additional address field for e.g. apartment number
- changeLogCount** The number of change log entries created by the user
- city** The name of the city wherein the user resides
- company** The name of the company for which the user works
- country** The name of the country wherein the user resides
- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user
  - Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- id** An integral, unique identifier for this user
- lastAuthenticated** The date and time at which the user was last authenticated, in **RFC 3339** format
- lastUpdated** The date and time at which the user was last modified, in **RFC 3339** format
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the [users/register](#) endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The name of the [Role](#) assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the [Tenant](#) to which this user belongs
- tenantId** The integral, unique identifier of the [Tenant](#) to which this user belongs
- ucdn** The name of the uCDN to which the user belongs
  - New in version 4.0.
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user

Deprecated since version 4.0: This field serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**username** The user's username

### 1059: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 13 May 2022 23:42:05 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 May 2022 22:42:05 GMT
Content-Length: 311

{ "response": {
  "addressLine1": null,
  "addressLine2": null,
  "changeLogCount": 1,
  "city": null,
  "company": null,
  "country": null,
  "email": "admin@no-reply.atc.test",
  "fullName": "Development Admin User",
  "gid": null,
  "id": 2,
  "lastAuthenticated": "2022-05-13T22:42:05.495439Z",
  "lastUpdated": "2022-05-13T22:42:05.495439Z",
  "newUser": false,
  "phoneNumber": null,
  "postalCode": null,
  "publicSshKey": null,
  "registrationSent": null,
  "role": "admin",
  "stateOrProvince": null,
  "tenant": "root",
  "tenantId": 1,
  "ucdn": "",
  "uid": null,
  "username": "admin"
}}
```



## PUT

**Warning:** Assuming the current user’s integral, unique identifier is known, it’s generally better to use the PUT method of the [users](#) instead.

**Warning:** Users that login via LDAP pass-back cannot be modified

Updates the date for the authenticated user.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** None

**Response Type** Object

## Request Structure

**addressLine1** The user’s address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides

**email** The user’s email address - cannot be an empty string<sup>1</sup>. The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user’s full name, e.g. “John Quincy Adams”

**gid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - please don’t use this

Deprecated since version 4.0: This field is serves no known purpose, and shouldn’t be used for anything so it can be removed in the future.

**id** The user’s integral, unique, identifier - this cannot be changed<sup>Page 1422, 1</sup>

**localPasswd** Optionally, the user’s password. This should never be given if it will not be changed. An empty string or null can be used to explicitly specify no change.

**phoneNumber** The user’s phone number

<sup>1</sup> This field cannot be null.

**postalCode** The user's postal code

**publicSshKey** The user's public encryption key used for the SSH protocol

**role** The integral, unique identifier of the highest permission *Role* which will be permitted to the user - this cannot be altered from the user's current *Role*<sup>Page 1422, 1</sup>

**stateOrProvince** The state or province in which the user resides

**tenantId** The integral, unique identifier of the *Tenant* to which the new user shall belong<sup>2Page 1422, 1</sup>

**ucdn** The name of the uCDN to which the user belongs

New in version 4.0.

**uid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - please don't use this

Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**username** The user's new username<sup>Page 1422, 1</sup>

Changed in version 4.0: In all earlier versions of the API, all of these objects needed to be properties of the "user" property of the JSON object request body.

### 1060: Request Example

```
PUT /api/4.0/user/current HTTP/1.1
```

```
User-Agent: python-requests/2.25.1
```

```
Accept-Encoding: gzip, deflate
```

```
Accept: */*
```

```
Connection: keep-alive
```

```
Cookie: mojolicious=...
```

```
Content-Length: 562
```

```
{
  "addressLine1": null,
  "addressLine2": null,
  "changeLogCount": 1,
  "city": null,
  "company": null,
  "country": null,
  "email": "admin@no-reply.atc.test",
  "fullName": "Development Admin User",
  "gid": null,
  "id": 2,
  "lastAuthenticated": "2022-05-13T22:42:05.495439Z",
```

(continues on next page)

---

<sup>2</sup> This endpoint respects tenancy; a user cannot assign itself to a *Tenant* that is not the same *Tenant* to which it was previously assigned or a descendant thereof.

(continued from previous page)

```

    "lastUpdated": "2022-05-13T22:42:05.495439Z",
    "newUser": false,
    "phoneNumber": null,
    "postalCode": null,
    "publicSshKey": null,
    "registrationSent": null,
    "role": "admin",
    "stateOrProvince": null,
    "tenant": "root",
    "tenantId": 1,
    "ucdn": "",
    "uid": null,
    "username": "admin"
  }

```

## Response Structure

**addressLine1** The user's address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**changeLogCount** The number of change log entries created by the user

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides

**email** The user's email address validated (circuitously) by [GitHub user asaskevich's regular expression](#) . Note that it can't actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user's full name, e.g. "John Quincy Adams"

**gid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user

Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**id** An integral, unique identifier for this user

**lastAuthenticated** The date and time at which the user was last authenticated, in [RFC 3339](#)

**lastUpdated** The date and time at which the user was last modified, in [Traffic Ops's Custom Date/Time Format](#)

**newUser** A meta field with no apparent purpose

- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the [users/register](#) endpoint, this will be the date and time at which the registration email was sent - otherwise it will be null
- role** The name of the [Role](#) assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the [Tenant](#) to which this user belongs
- tenantId** The integral, unique identifier of the [Tenant](#) to which this user belongs
- ucdn** The name of the uCDN to which the user belongs
- New in version 4.0.
- uid** A legacy field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- username** The user's username

#### 1061: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 13 May 2022 23:45:22 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 May 2022 22:45:22 GMT
Content-Length: 370

{ "alerts": [
  {
    "text": "User profile was successfully updated",
    "level": "success"
  }
],
"response": {
  "addressLine1": null,
  "addressLine2": null,
  "changeLogCount": 1,
  "city": null,
```

(continues on next page)

(continued from previous page)

```

"company": null,
"country": null,
"email": "admin@no-reply.atc.test",
"fullName": "Development Admin User",
"gid": null,
"id": 2,
"lastAuthenticated": "2022-05-13T22:44:55.973452Z",
"lastUpdated": "2022-05-13T22:45:22.505401Z",
"newUser": false,
"phoneNumber": null,
"postalCode": null,
"publicSshKey": null,
"registrationSent": null,
"role": "admin",
"stateOrProvince": null,
"tenant": "root",
"tenantId": 1,
"ucdn": "",
"uid": null,
"username": "admin"
}

```

## user/login

### POST

Authentication of a user using username and password. Traffic Ops will send back a session cookie.

**Auth. Required** No

**Roles Required** None

**Permissions Required** None

**Response Type** undefined

## Request Structure

**p** Password

**u** Username

### 1062: Request Example

```

POST /api/4.0/user/login HTTP/1.1
Host: trafficops.infra.ciab.test

```

(continues on next page)

(continued from previous page)

```
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 26
Content-Type: application/json

{
  "u": "admin",
  "p": "twelve"
}
```

## Response Structure

### 1063: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
↳Ud06T3tMNctnVusDXzRjVwwYOnD7jmnBzPEB9PvOt2bHajTv3SKTPiIZjDzvhU6EX4p+JoG4fA5wlhgxp
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 15:21:33 GMT
Content-Length: 65

{ "alerts": [
  {
    "text": "Successfully logged in.",
    "level": "success"
  }
]}
```

## user/login/oauth

### POST

Authentication of a user by exchanging a code for an encrypted JSON Web Token from an OAuth service. Traffic Ops will POST to the `authCodeTokenUrl` to exchange the code for an encrypted JSON Web Token. It will then decode and validate the token, validate the key set domain, and send back a session cookie.

**Auth. Required** No

**Roles Required** None

**Permissions Required** None

**Response Type** undefined

### Request Structure

**authCodeTokenUrl** URL for code-to-token conversion

**code** Code

**clientId** Client Id

**redirectUri** Redirect URI

#### 1064: Request Example

```
POST /api/4.0/user/login/oauth HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 26
Content-Type: application/json

{
  "authCodeTokenUrl": "https://url-to-convert-code-to-token.example.
  ↪com",
  "code": "AbCd123",
  "clientId": "oauthClientId",
  "redirectUri": "https://traffic-portal.example.com/sso"
}
```

## Response Structure

### 1065: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳UdO6T3tMNctnVusDXzRjVwwYOnD7jmnBzPEB9PvOt2bHajTv3SKTPiIZjDzvhU6EX4p+JoG4fA5wlhgxp
X-Server-Name: traffic_ops_golang/
Date: Thu, 13 Dec 2018 15:21:33 GMT
Content-Length: 65

{ "alerts": [
  {
    "text": "Successfully logged in.",
    "level": "success"
  }
]}
```

### user/login/token

#### POST

Authentication of a user using a token. Normally, the token is obtained via a call to either *user/reset\_password* or *users/register*.

**Auth. Required** No

**Roles Required** None

**Permissions Required** None

**Response Type** undefined



## Request Structure

**t** A UUID generated for the user.

---

### Implementation Detail

Though not strictly necessary for authentication provided direct database access, the tokens generated for use with this endpoint are compliant with [RFC 4122](#).

---

#### 1066: Request Example

```
POST /api/4.0/user/login/token HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 44
Content-Type: application/json

{
  "t": "18EE200C-FF24-11E8-BF01-870C776752A3"
}
```

## Response Structure

#### 1067: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512: FuS3TkVosxHtpxRGMJ2on+WnFdYTNSPjxz/Gh1iT4UCJ2/
↳P0twUbAGQ3tTx9EfGiAzg9CNQiVUFGnYjJZ6NCpg==
X-Server-Name: traffic_ops_golang/
Date: Fri, 20 Sep 2019 15:02:43 GMT
Content-Length: 66

{ "alerts": [
  {
```

(continues on next page)

(continued from previous page)

```

    "text": "Successfully logged in.",
    "level": "success"
  }
}]

```

**user/logout**

**POST**

User logout. Invalidates the session cookie of the currently logged-in user.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** None

**Response Type** undefined

## Request Structure

No parameters available

## Response Structure

1068: Response Example

```

HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 21:25:36 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512: 6KEdr1ZC512zk0l03KwvQE0L7qrJ/
→+ek6ztymkYy9p8gdPUyYyzGEAJ/Ldb8GY0UBFYmgqeZ3yWHvTcEsOQMiw==
Content-Length: 61

{ "alerts": [

```

(continues on next page)

(continued from previous page)

```
{
  "level": "success",
  "text": "You are logged out."
}
]}
```

## **user/reset\_password**

### **POST**

Sends an email to reset a user's password.

**Auth. Required** No

**Roles Required** None

**Permissions Required** None

**Response Type** undefined

### **Request Structure**

**email** The email address of the user to initiate password reset

#### 1069: Request Example

```
POST /api/4.0/user/reset_password HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 35
Content-Type: application/json

{
  "email": "test@example.com"
}
```

## Response Structure

#### 1070: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
  ↳Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 22:11:53 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54
  ↳GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
  ↳lKWwVYbgKklk7bljnQJZwWV5bljIk+GkooD6SAC3CSEXVKvfbL9dgL5iBc/
  ↳BNNRk2pIU5I/1GglcdLrXsF1ZA==
Content-Length: 109

{ "alerts": [
  {
    "level": "success",
    "text": "Successfully sent password reset to email
    ↳'test@example.com'"
  }
]}
```

## users

### GET

Retrieves all requested users.

**Auth. Required** Yes

**Roles Required** None<sup>1</sup>

**Permissions Required** USER:READ

**Response Type** Array

### Request Structure

Table 436: Request Query Parameters

Name	Re-quired	Description
id	no	Return only the user identified by this integral, unique identifier
tenant	no	Return only users belonging to the <i>Tenant</i> identified by tenant name
role	no	Return only users belonging to the <i>Role</i> identified by role name
username	no	Return only the user with this username
orderby	no	Choose the ordering of the results - must be the name of one of the fields of the objects in the response array
sortOrder	no	Changes the order of sorting. Either ascending (default or “asc”) or descending (“desc”)
limit	no	Choose the maximum number of results to return
offset	no	The number of results to skip before beginning to return results. Must use in conjunction with limit
page	no	Return the <i>n</i> <sup>th</sup> page of results, where “n” is the value of this parameter, pages are limit long and the first page is 1. If offset was defined, this query parameter has no effect. limit must be defined to make use of page.

### 1071: Request Example

```
GET /api/4.0/users?username=mike HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

<sup>1</sup> While no roles are required, this endpoint does respect tenancy. A user will only be able to see, create, delete or modify other users belonging to the same tenant, or its descendants.

## Response Structure

- addressLine1** The user's address - including street name and number
- addressLine2** An additional address field for e.g. apartment number
- changeLogCount** The number of change log entries created by the user
- city** The name of the city wherein the user resides
- company** The name of the company for which the user works
- country** The name of the country wherein the user resides
- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- id** An integral, unique identifier for this user
- lastAuthenticated** The date and time at which the user was last authenticated, in **RFC 3339**
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The name of the role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- ucdn** The name of the uCDN to which the user belongs
- New in version 4.0.
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`

Deprecated since version 4.0: This field serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**username** The user's username

#### 1072: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 13 May 2022 23:16:14 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 May 2022 22:16:14 GMT
Content-Length: 350

{ "response": [
  {
    "addressLine1": "22 Mike Wazowski You've Got Your Life Back Lane",
    "addressLine2": null,
    "changeLogCount": 0,
    "city": "Monstropolis",
    "company": null,
    "country": null,
    "email": "mwazowski@minc.biz",
    "fullName": "Mike Wazowski",
    "gid": null,
    "id": 3,
    "lastAuthenticated": null,
    "lastUpdated": "2022-05-13T22:13:54.605052Z",
    "newUser": true,
    "phoneNumber": null,
    "postalCode": null,
    "publicSshKey": null,
    "registrationSent": null,
    "role": "admin",
    "stateOrProvince": null,
    "tenant": "root",
    "tenantId": 1,
    "ucdn": "",
    "uid": null,
    "username": "mike"
  }
]}
```

### POST

Creates a new user.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”<sup>Page 1422, 1</sup>

**Permissions Required** USER:CREATE, USER:READ

**Response Type** Object

### Request Structure

**addressLine1** An optional field which should contain the user’s address - including street name and number

**addressLine2** An optional field which should contain an additional address field for e.g. apartment number

**city** An optional field which should contain the name of the city wherein the user resides

**company** An optional field which should contain the name of the company for which the user works

**country** An optional field which should contain the name of the country wherein the user resides

**email** The user’s email address The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user’s full name, e.g. “John Quincy Adams”

**gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user

Deprecated since version 4.0: This field is serves no known purpose, and shouldn’t be used for anything so it can be removed in the future.

**localPasswd** The user’s password

**newUser** An optional meta field with no apparent purpose - don’t use this

**phoneNumber** An optional field which should contain the user’s phone number

**postalCode** An optional field which should contain the user’s postal code

**publicSshKey** An optional field which should contain the user’s public encryption key used for the SSH protocol

**role** The name that corresponds to the highest permission role which will be permitted to the user



**stateOrProvince** An optional field which should contain the name of the state or province in which the user resides

**tenantId** The integral, unique identifier of the tenant to which the new user shall belong

**ucdn** The name of the uCDN to which the user belongs

New in version 4.0.

**uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user

Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**username** The new user's username

#### 1073: Request Example

```
POST /api/4.0/users HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 304
Content-Type: application/json

{
  "username": "mike",
  "addressLine1": "22 Mike Wazowski You've Got Your Life Back Lane",
  "city": "Monstropolis",
  "compary": "Monsters Inc.",
  "email": "mwazowski@minc.biz",
  "fullName": "Mike Wazowski",
  "localPasswd": "BFFsully",
  "confirmLocalPasswd": "BFFsully",
  "newUser": true,
  "role": "admin",
  "tenantId": 1
}
```

## Response Structure

- addressLine1** The user's address - including street name and number
- addressLine2** An additional address field for e.g. apartment number
- changeLogCount** The number of change log entries created by the user
- city** The name of the city wherein the user resides
- company** The name of the company for which the user works
- country** The name of the country wherein the user resides
- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- id** An integral, unique identifier for this user
- lastAuthenticated** The date and time at which the user was last authenticated, in **RFC 3339**
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The name of the role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- ucdn** The name of the uCDN to which the user belongs
- New in version 4.0.
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`

Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**username** The user's username

#### 1074: Response Example

```
HTTP/1.1 201 Created
Content-Encoding: gzip
Content-Type: application/json
Location: /api/4.0/users?id=3
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 13 May 2022 23:13:54 GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 May 2022 22:13:54 GMT
Content-Length: 382

{ "alerts": [
  {
    "text": "user was created.",
    "level": "success"
  }
],
"response": {
  "addressLine1": "22 Mike Wazowski You've Got Your Life Back Lane",
  "addressLine2": null,
  "changeLogCount": null,
  "city": "Monstropolis",
  "company": null,
  "country": null,
  "email": "mwazowski@minc.biz",
  "fullName": "Mike Wazowski",
  "gid": null,
  "id": 3,
  "lastAuthenticated": null,
  "lastUpdated": "2022-05-13T22:13:54.605052Z",
  "newUser": true,
  "phoneNumber": null,
  "postalCode": null,
  "publicSshKey": null,
  "registrationSent": null,
  "role": "admin",
  "stateOrProvince": null,
  "tenant": "root",
  "tenantId": 1,
  "ucdn": ""
}
```

(continues on next page)

(continued from previous page)

```
"uid": null,
"username": "mike"
}}
```

users/{{ID}}

GET

Retrieves a specific user.

**Auth. Required** Yes

**Roles Required** None

**Permissions Required** USER:READ

**Response Type** Array

## Request Structure

Table 437: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the user to be retrieved

1075: Request Example

```
GET /api/4.0/users/3 HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
```

## Response Structure

**addressLine1** The user's address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**changeLogCount** The number of change log entries created by the user

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides

- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- id** An integral, unique identifier for this user
- lastAuthenticated** The date and time at which the user was last authenticated, in **RFC 3339**
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The name of the role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- ucdn** The name of the uCDN to which the user belongs
- New in version 4.0.
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- username** The user's username

#### 1076: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 13 May 2022 23:48:14.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 May 2022 22:48:14 GMT
Content-Length: 350

{ "response": [
  {
    "addressLine1": "22 Mike Wazowski You've Got Your Life Back_
↳Lane",
    "addressLine2": null,
    "changeLogCount": 0,
    "city": "Monstropolis",
    "company": null,
    "country": null,
    "email": "mwazowski@minc.biz",
    "fullName": "Mike Wazowski",
    "gid": null,
    "id": 3,
    "lastAuthenticated": null,
    "lastUpdated": "2022-05-13T22:13:54.605052Z",
    "newUser": true,
    "phoneNumber": null,
    "postalCode": null,
    "publicSshKey": null,
    "registrationSent": null,
    "role": "admin",
    "stateOrProvince": null,
    "tenant": "root",
    "tenantId": 1,
    "ucdn": "",
    "uid": null,
    "username": "mike"
  }
]}
```

## PUT

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** USER:UPDATE, USER:READ

**Response Type** Object

## Request Structure

Table 438: Request Path Parameters

Name	Description
ID	The integral, unique identifier of the user to be modified

**addressLine1** An optional field which should contain the user’s address - including street name and number

**addressLine2** An optional field which should contain an additional address field for e.g. apartment number

**city** An optional field which should contain the name of the city wherein the user resides

**company** An optional field which should contain the name of the company for which the user works

**country** An optional field which should contain the name of the country wherein the user resides

**email** The user’s email address The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**fullName** The user’s full name, e.g. “John Quincy Adams”

**gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always null

Deprecated since version 4.0: This field is serves no known purpose, and shouldn’t be used for anything so it can be removed in the future.

**id** This field *may* optionally be given, but **must** match the user’s existing ID as IDs are immutable

**localPasswd** The user’s password

**newUser** An optional meta field with no apparent purpose - don’t use this

**phoneNumber** An optional field which should contain the user’s phone number

**postalCode** An optional field which should contain the user’s postal code

**publicSshKey** An optional field which should contain the user’s public encryption key used for the SSH protocol

**role** The name of the Role which will be granted to the user

**stateOrProvince** An optional field which should contain the name of the state or province in which the user resides

**tenantId** The integral, unique identifier of the tenant to which the new user shall belong

**ucdn** The name of the uCDN to which the user belongs

New in version 4.0.

**uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`

Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.

**username** The user's username

### 1077: Request Structure

```
PUT /api/4.0/users/3 HTTP/1.1
User-Agent: python-requests/2.25.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Cookie: mojolicious=...
Content-Length: 476

{
  "addressLine1": "not a real address",
  "addressLine2": "not a real address either",
  "city": "not a real city",
  "company": "not a real company",
  "country": "not a real country",
  "email": "mwazowski@minc.biz",
  "fullName": "Mike Wazowski",
  "phoneNumber": "not a real phone number",
  "postalCode": "not a real postal code",
  "publicSshKey": "not a real ssh key",
  "stateOrProvince": "not a real state or province",
  "tenantId": 1,
  "role": "admin",
  "username": "mike"
}
```

### Response Structure

**addressLine1** The user's address - including street name and number

**addressLine2** An additional address field for e.g. apartment number

**changeLogCount** The number of change log entries created by the user

**city** The name of the city wherein the user resides

**company** The name of the company for which the user works

**country** The name of the country wherein the user resides



- email** The user's email address
- fullName** The user's full name, e.g. "John Quincy Adams"
- gid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX group ID of the user - now it is always `null`
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- id** An integral, unique identifier for this user
- lastAuthenticated** The date and time at which the user was last authenticated, in **RFC 3339**
- lastUpdated** The date and time at which the user was last modified, in *Traffic Ops's Custom Date/Time Format*
- newUser** A meta field with no apparent purpose that is usually `null` unless explicitly set during creation or modification of a user via some API endpoint
- phoneNumber** The user's phone number
- postalCode** The postal code of the area in which the user resides
- publicSshKey** The user's public key used for the SSH protocol
- registrationSent** If the user was created using the *users/register* endpoint, this will be the date and time at which the registration email was sent - otherwise it will be `null`
- role** The name of the role assigned to this user
- stateOrProvince** The name of the state or province where this user resides
- tenant** The name of the tenant to which this user belongs
- tenantId** The integral, unique identifier of the tenant to which this user belongs
- ucdn** The name of the uCDN to which the user belongs
- New in version 4.0.
- uid** A deprecated field only kept for legacy compatibility reasons that used to contain the UNIX user ID of the user - now it is always `null`
- Deprecated since version 4.0: This field is serves no known purpose, and shouldn't be used for anything so it can be removed in the future.
- username** The user's username

#### 1078: Response Example

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Content-Type: application/json
Permissions-Policy: interest-cohort=()
Set-Cookie: mojolicious=...; Path=/; Expires=Fri, 13 May 2022 23:50:25.
→GMT; Max-Age=3600; HttpOnly
```

(continues on next page)

(continued from previous page)

```
Vary: Accept-Encoding
X-Server-Name: traffic_ops_golang/
Date: Fri, 13 May 2022 22:50:25 GMT
Content-Length: 399

{ "alerts": [
  {
    "text": "user was updated.",
    "level": "success"
  }
],
"response": {
  "addressLine1": "not a real address",
  "addressLine2": "not a real address either",
  "changeLogCount": 0,
  "city": "not a real city",
  "company": "not a real company",
  "country": "not a real country",
  "email": "mwazowski@minc.biz",
  "fullName": "Mike Wazowski",
  "gid": null,
  "id": 3,
  "lastAuthenticated": null,
  "lastUpdated": "2022-05-13T22:50:25.965004Z",
  "newUser": false,
  "phoneNumber": "not a real phone number",
  "postalCode": "not a real postal code",
  "publicSshKey": "not a real ssh key",
  "registrationSent": null,
  "role": "admin",
  "stateOrProvince": "not a real state or province",
  "tenant": "root",
  "tenantId": 1,
  "ucdn": "",
  "uid": null,
  "username": "mike"
}}
```

## users/register

### POST

Register a user and send registration email.

**Auth. Required** Yes

**Roles Required** “admin” or “operations”

**Permissions Required** USER:CREATE, USER:READ

**Response Type** undefined

### Request Structure

**email** Email address of the new user The given email is validated (circuitously) by [GitHub user asaskevich’s regular expression](#) . Note that it can’t actually distinguish a valid, deliverable, email address but merely ensure the email is in a commonly-found format.

**role** The integral, unique identifier of the highest permissions *Role* which will be afforded to the new user. It restricts the allowed values to identifiers for *Roles* with at most the same permissions level as the requesting user.

**tenantId** A field containing the integral, unique identifier of the *Tenant* to which the new user will belong. It restricts the allowed values to identifiers for *Tenants* within the requesting user’s *Tenant*’s permissions.

#### 1079: Request Example

```
POST /api/4.0/users/register HTTP/1.1
Host: trafficops.infra.ciab.test
User-Agent: curl/7.47.0
Accept: */*
Cookie: mojolicious=...
Content-Length: 59
Content-Type: application/json

{
  "email": "test@example.com",
  "role": 3,
  "tenantId": 1
}
```

### Response Structure

#### 1080: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
→Accept
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Content-Type: application/json
Date: Thu, 13 Dec 2018 22:03:22 GMT
X-Server-Name: traffic_ops_golang/
Set-Cookie: mojolicious=...; Path=/; Expires=Mon, 18 Nov 2019 17:40:54.
→GMT; Max-Age=3600; HttpOnly
Vary: Accept-Encoding
Whole-Content-Sha512:
→yvf++0qxvu3u0IAYbWLUgJKxZ4T60Mi5H9eGTxrKLxnRsHw0PdDIrThbTnWtATBkak4vU/
→dPHLLXKW85LUTEWg==
Content-Length: 160

{ "alerts": [
  {
    "level": "success",
    "text": "Sent user registration to test@example.com with the
→following permissions [ role: read-only | tenant: root ]"
  }
]}
```

## vault/ping

### GET

Pings Traffic Vault to retrieve status.

**Auth. Required** Yes

**Roles Required** “read-only”

**Permissions Required** TRAFFIC-VAULT:READ

**Response Type** Object

### Request Structure

No parameters available.

### Response Properties

**status** The status returned from the ping request to the Traffic Vault server

**server** The Traffic Vault server that was pinged

1081: Response Example

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,
    ↳Accept, Set-Cookie, Cookie
Access-Control-Allow-Methods: POST,GET,OPTIONS,PUT,DELETE
Access-Control-Allow-Origin: *
Content-Type: application/json
Set-Cookie: mojolicious=...; Path=/; Expires=Tue, 25 Feb 2020 15:37:54
    ↳GMT; Max-Age=3600; HttpOnly
Whole-Content-Sha512:
    ↳z9P1NkxGebPncUhaChDHtYKYI+XVZfhE6Y84TuwoASZFIMfISELwADLpvpPTN+wnzBfREksLYn+0313Q
X-Server-Name: traffic_ops_golang/
Date: Tue, 25 Feb 2020 14:37:55 GMT
Content-Length: 90

{ "response":
  {
    "status": "OK",
    "server": "trafficvault.infra.ciab.test:8087"
  }
}
```



A living list of tools for testing, interacting with, and developing for the Traffic Control CDN

## 6.1 Tools

This is a living list of tools used to interact with, test, and develop for the Traffic Control CDN.

### 6.1.1 `atstccfg`

**`atstccfg`** is a tool for generating configuration files server-side on ATC cache servers.

**Warning:** **`atstccfg`** does not have a stable command-line interface, it can and will change without warning. Scripts should avoid calling it for the time being, as its only intended caller is *ORT*.

The source code for **`atstccfg`** may be found in `traffic_ops/ort/atstccfg`, and the Go module reference is [github.com/apache/trafficcontrol/traffic\\_ops/ort/atstccfg](https://github.com/apache/trafficcontrol/traffic_ops/ort/atstccfg).

#### Usage

- `atstccfg -h`
- `atstccfg -v`
- `atstccfg -l`
- `atstccfg [-e ERROR_LOCATION] [-i INFO_LOCATION] [-p] [-P TO_PASSWORD] [-r N] [-s] [-t TIMEOUT] [-u TO_URL] [-U TO_USER] [-w WARNING_LOCATION] [-y] [--dir TSROOT] -n CACHE_NAME`
- `atstccfg [-e ERROR_LOCATION] [-i INFO_LOCATION] [-p] [-P TO_PASSWORD] [-r N] [-s] [-t TIMEOUT] [-u TO_URL] [-U TO_USER] [-w WARNING_LOCATION] [--dir TSROOT] -n CACHE_NAME -d DATA`

- `atstccfg [-e ERROR_LOCATION] [-i INFO_LOCATION] [-p] [-P TO_PASSWORD] [-r N] [-s] [-t TIMEOUT] [-u TO_URL] [-U TO_USER] [-w WARNING_LOCATION] [--dir TSROOT] -n CACHE_NAME -a REVAL_STATUS -q QUEUE_STATUS`

When called using the fourth form, **atstccfg** provides its normal output. This is the entirety of all configuration files necessary for the server, all provided at once. The output is in *mixed/multipart* format, defined by [RFC 1521](#). Each chunk of the message comes with the proprietary Path header, which specifies the exact location on disk of the file whose contents are contained in that chunk.

## Options

- a REVAL\_STATUS, **--set-reval-status** REVAL\_STATUS  
Sets the `reval_pending` property of the server in Traffic Ops. Must be ‘true’ or ‘false’. Requires **--set-queue-status** also be set. This disables normal output.
- e ERROR\_LOCATION, **--log-location-error** ERROR\_LOCATION  
The file location to which to log errors. Respects the special string constants of [github.com/apache/trafficcontrol/lib/go-log](#). Default: ‘stderr’
- d DATA, **--get-data** DATA  
Specifies non-configuration-file data to retrieve from Traffic Ops. This disables normal output. Valid values are:
  - chkconfig** Retrieves information about the services which should be running on the *cache server*. The output will be in JSON-encoded format as an array of objects with these fields:
    - name** The name of the service. This should correspond to an existing systemd service unit file.
    - value** A “chkconfig” line describing on which “run-levels” the services should be running. See the *chkconfig(8)* manual pages for details on what this field means.
  - packages** Retrieves information about the packages which should exist on the *cache server*. The output will be in JSON-encoded format as an array of objects with these fields:
    - name** The name of the package. This should hopefully be a meaningful package name for the *cache server*’s package management system.
    - version** The version of the package which should be installed. This might also be an empty string which means “any version will do”.
  - statuses** Retrieves all statuses from Traffic Ops. This is defined to be exactly the response object from the response to a GET request made to the *statuses* Traffic Ops API endpoint.
  - system-info** Retrieves generic information about the Traffic Control system from the *system/info* API endpoint. The output is the `parameters` object of the responses from GET requests to that endpoint (still JSON-encoded).
  - update-status** Retrieves information about the current update status using *servers/{hostname}/update\_status*. The response is in the same format as



the responses for that endpoint's GET method handler - except that that endpoint returns an array and this **atstccfg** call signature returns a single one of those elements. Which one is chosen is arbitrary (hence undefined behavior when more than one server with the same hostname exists).

**--dir** TSRROOT

Specifies a directory path in which to place Traffic Server configuration files, in the event that "location" *Parameters* are not found for them. If this is not given and location *Parameters* are not found for required files, then **atstccfg** will exit with an error.

The files that **atstccfg** considers "required" for these purposes are:

- cache.config
- hosting.config
- parent.config
- plugin.config
- records.config
- remap.config
- storage.config
- volume.config

**-h, --help**

Print usage information and exit.

**-i** INFO\_LOCATION, **--log-location-info** INFO\_LOCATION

The file location to which to log information messages. Respects the special string constants of [github.com/apache/trafficcontrol/lib/go-log](https://github.com/apache/trafficcontrol/lib/go-log). Default: 'stderr'

**-l, --list-plugins**

List the loaded plug-ins and then exit.

**-n** NAME, **--cache-host-name** NAME

Required. Specifies the (short) hostname of the *cache server* for which output will be generated. Must be the server hostname in Traffic Ops, not a URL, or FQDN. Behavior when more than one server exists with the passed hostname is undefined.

**-p, --traffic-ops-disable-proxy**

Bypass the Traffic Ops caching proxy and make requests directly to Traffic Ops. Has no effect if no such proxy exists.

**-P** TO\_PASSWORD, **--traffic-ops-password** TO\_PASSWORD

Authenticate using this password - if not given, **atstccfg** will attempt to use the value of the *TO\_PASSWORD* environment variable.

**-q** QUEUE\_STATUS, **--set-queue-status** QUEUE\_STATUS

Sets the *upd\_pending* property of the server identified by *--cache-host-name* to the specified value, which must be 'true' or 'false'. Requires *--set-reval-status* to also be set.

**-r** N, **--num-retries** N

The number of times to retry getting a file if it fails. Default: 5

**-s, --traffic-ops-insecure**

If given, SSL certificate errors will be ignored when communicating with Traffic Ops.

**Caution:** The use of this option in production environments is discouraged.

- t TIMEOUT, --traffic-ops-timeout-milliseconds TIMEOUT**  
Sets the timeout - in milliseconds - for requests made to Traffic Ops. Default: 30000
- u TO\_URL, --traffic-ops-url TO\_URL**  
Request this URL, e.g. <https://trafficops.infra.ciab.test/servers/edge/configfiles/ats>. If not given, **atstccfg** will attempt to use the value of the [TO\\_URL](#) environment variable.
- U TO\_USER, --traffic-ops-user TO\_USER**  
Authenticate as the user TO\_USER - if not given, **atstccfg** will attempt to use the value of the [TO\\_USER](#) environment variable.
- v, --version**  
Print version information and exit.
- w WARNING\_LOCATION, --log-location-warning WARNING\_LOCATION**  
The file location to which to log warnings. Respects the special string constants of [github.com/apache/trafficcontrol/lib/go-log](https://github.com/apache/trafficcontrol/lib/go-log). Default: 'stderr'
- y, --revalidate-only**  
When given, **atstccfg** will only emit files relevant for updating *Content Invalida-tion Jobs*. for Apache Traffic Server implementations, this limits the output to be only files named `regex_revalidate.config`. Has no effect if [--get-data](#) or [--set-queue-status/--set-reval-status](#) is/are used.

## Environment Variables

**atstccfg** supports authentication with a Traffic Ops instance using the environment variables [TO\\_URL](#) (if [-u/--traffic-ops-url](#) is not given), [TO\\_USER](#) (if [-U/--traffic-ops-user](#) is not given), and [TO\\_PASSWORD](#) (if [-P/--traffic-ops-password](#) is not given).

## 6.1.2 Apache-TrafficControl Package

### Contributing

When contributing Python code to Apache Traffic Control, please ensure that it passes the provided Pylint configuration file ([traffic\\_control/clients/python/pylint.rc](#)).

## Package Contents

**exception** `trafficops.InvalidJSONError(*args, resp=None)`

Bases: `ValueError`

An error that occurs when an invalid JSON payload is passed to an endpoint.

**resp = None**

Contains the response object that generated the error

**exception** `trafficops.LoginError(*args)`

Bases: `OSError`

This represents an error that occurred during server login.

**exception** `trafficops.OperationError(*args, resp=None)`

Bases: `OSError`

This class represents a generic error, indicating something went wrong with the request or on the server.

**resp = None**

Contains the response object that generated the error

**class** `trafficops.RestApiSession(host_ip, api_version=None, api_base_path='api',  
host_port=443, ssl=True, headers=None,  
verify_cert=True, create_session=False,  
max_retries=5)`

Bases: `object`

This class represents a login session with a generic REST API server. It provides base functionality inherited by [TOSession](#).

**property** `api_base_url`

Returns the base URL. (read-only)

**Returns** The base URL should match `'[w+-.]+://[w+-.]+(:d+)?'` e.g.  
`'https://to.somedomain.net/api/0.1/'`

**Return type** `str`

**property** `api_version`

Returns the api version. (read-only)

**Returns** The api version from which this instance will request endpoints.

**Return type** `str`

**close()**

Close and cleanup the requests Session object.

**Returns** `None`

**Return type** `NoneType`

**create()**

Create the requests.Session to communicate with the RESTful API.

**Returns** `None`

**Return type** `NoneType`

**delete**(*api\_path*, \*args, \*\*kwargs)

Perform HTTP DELETE requests

**Parameters**

- **api\_path** (*str*) – The path to the API end-point that you want to call which does not include the base URL e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}`
- **kwargs** (*Dict[str, Any]*) – Passed Keyword Parameters. If you need to send JSON data to the endpoint pass the keyword parameter data with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters

**Returns** Python data structure distilled from JSON from the API request.

**Return type** Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response]

**Raises** Union[LoginError, OperationError]

**get**(*api\_path*, \*args, \*\*kwargs)

Perform http get requests

**Parameters**

- **api\_path** (*str*) – The path to the API end-point that you want to call which does not include the base url e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}`
- **kwargs** (*Dict[str, Any]*) – Passed Keyword Parameters. If you need to send JSON data to the endpoint pass the keyword parameter data with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters

**Returns** Python data structure distilled from JSON from the API request.

**Return type** Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response]

**Raises** Union[LoginError, OperationError]

**head**(*api\_path*, \*args, \*\*kwargs)

Perform HTTP HEAD requests :param `api_path`: The path to the API end-point that you want to call which does not include the base URL e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}` :type `api_path`: `str` :param `kwargs`: Passed Keyword Parameters. If you need to send JSON data to the endpoint pass the keyword param-

eter data with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters :type kwargs: Dict[str, Any] :return: Python data structure distilled from JSON from the API request. :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response] :raises: Union[LoginError, OperationError]

### **property is\_open**

Is the session open to the RESTful API? (Read-only Property)

**Returns** True if yes, otherwise, False

**Return type** bool

### **options**(*api\_path*, \*args, \*\*kwargs)

Perform HTTP OPTIONS requests :param api\_path: The path to the API endpoint that you want to call which does not include the base URL e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}` :type api\_path: str :param kwargs: Passed Keyword Parameters. If you need to send JSON data to the endpoint pass the keyword parameter `data` with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters :type kwargs: Dict[str, Any] :return: Python data structure distilled from JSON from the API request. :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response] :raises: Union[LoginError, OperationError]

### **patch**(*api\_path*, \*args, \*\*kwargs)

Perform HTTP PATCH requests :param api\_path: The path to the API endpoint that you want to call which does not include the base URL e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}` :type api\_path: str :param kwargs: Passed Keyword Parameters. If you need to send JSON data to the endpoint pass the keyword parameter `data` with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters :type kwargs: Dict[str, Any] :return: Python data structure distilled from JSON from the API request. :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response] :raises: Union[LoginError, OperationError]

### **post**(*api\_path*, \*args, \*\*kwargs)

Perform http post requests

#### **Parameters**

- **api\_path** (*str*) – The path to the API end-point that you want to call which does not include the base URL e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}`
- **kwargs** (*Dict[str, Any]*) – Passed Keyword Parameters

ters. If you need to send JSON data to the endpoint pass the keyword parameter `data` with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters

**Returns** Python data structure distilled from JSON from the API request.

**Return type** Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response]

**Raises** Union[LoginError, OperationError]

**put**(*api\_path*, \*args, \*\*kwargs)

Perform http put requests

**Parameters**

- **api\_path** (*str*) – The path to the API end-point that you want to call which does not include the base URL e.g. `user/login`, `servers`, etc. This string can contain substitution parameters as denoted by a valid `field_name` replacement field specification as per `str.format()` e.g. `cachegroups/{id}` or `cachegroups/{id:d}`
- **kwargs** (*Dict[str, Any]*) – Passed Keyword Parameters. If you need to send JSON data to the endpoint pass the keyword parameter `data` with the Python data structure. This method will convert it to JSON before sending it to the API endpoint. Use `query_params` to pass a dictionary of query parameters

**Returns** Python data structure distilled from JSON from the API request.

**Return type** Tuple[Union[Dict[str, Any], List[Dict[str, Any]], munch.Munch, List[munch.Munch]], requests.Response]

**Raises** Union[LoginError, OperationError]

**property server\_url**

The URL without the api portion. (read-only)

**Returns** The URL should match `'[w+-.]+://[w+-.]+(:d+)?'` e.g. `'https://to.somedomain.net'` or `'https://to.somedomain.net:443'`

**Return type** str

**property session**

The RESTful API session (Read-only Property)

**Returns** The requests session

**Return type** requests.Session

**class** `trafficops.TOSession`(*host\_ip*, *host\_port*=443, *api\_version*='3.0', *ssl*=True, *headers*=None, *verify\_cert*=True)

Bases: `trafficops.restapi.RestApiSession`

Traffic Ops Session Class Once you login to the Traffic Ops API via `login()`, you can call one or more of the methods to retrieve, POST, PUT, DELETE, etc. data to the API. If you are not logged in, an exception will be thrown if you try to call any of the endpoint

methods. This API client is simplistic and lightly structured on purpose but adding support for new endpoints routinely takes seconds. Another nice bit of convenience that result data is, by default, wrapped in `munch.Munch` objects, which provide attribute access to the returned dictionaries/hashes - e.g. `a_dict['a_key']` with `munch` becomes `a_dict.a_key` or `a_dict['a_key']`. Also, the lack of rigid structure (loose coupling) means many changes to the Traffic Ops API, as it evolves, will probably go un-noticed (usually additions), which means fewer future problems to potentially fix in user applications.

An area of improvement for later is defining classes to represent request data instead of loading up dictionaries for request data.

Please see the [API documentation](#) for the details of the API endpoints.

Adding end-point methods

1082: Endpoint with no URL parameters and no query parameters

```
@api_request('get', 'cdns', ('3.0',))
def get_cdns(self):
    pass
```

1083: End-point with URL parameters and no query parameters

```
@api_request('get', 'cdns/{cdn_id:d}', ('3.0',))
def get_cdn_by_id(self, cdn_id=None):
    pass
```

1084: End-point with no URL parameters but with query parameters

```
@api_request('get', 'deliveryservices', ('3.0',))
def get_deliveryservices(self, query_params=None):
    pass
```

1085: End-point with URL parameters and query parameters

```
@api_request('get', 'deliveryservices/xmlId/{xml_id}/sslkeys', ('3.0',))
def get_deliveryservice_ssl_keys_by_xml_id(self, xml_id=None, query_params=None):
    pass
```

1086: End-point with request data

```
@api_request('post', 'cdns', ('3.0',))
def create_cdn(self, data=None):
    pass
```

1087: End-point with URL parameters and request data

```
@api_request('put', 'cdns', ('3.0',))
def update_cdn_by_id(self, cdn_id=None, data=None):
    pass
```

Calling end-point methods

*get\_cdns()* calls endpoint *cdns* e.g. *t.get\_cdns()*

*get\_types()* calls endpoint *types*, optionally with query parameters e.g. *get\_foo\_data(id=45, query\_params={'sort': 'asc'})* calls endpoint GET *api/2.x/foo/45?sort=asc* (presumably)

*cdns\_queue\_update()* calls endpoint *cdns/{{ID}}/queue\_update*, with an ID path parameter and a JSON payload e.g. *cdns\_queue\_update(id=1, data={'action': 'queue'})*

---

**Note:** Only a small subset of the API endpoints are implemented. More can be implemented as needed.

---

**add\_ssl\_keys\_to\_deliveryservice**(*data=None*)

Add SSL keys to a Delivery Service. *deliveryservices/sslkeys/add* :param data: The parameter data to use for adding SSL keys to a Delivery Service. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**assign\_delivery\_services\_to\_federations**(*federation\_id=None, data=None*)

Create one or more federation / delivery service assignments. *federations/{{ID}}/deliveryservices* :param federation\_id: The federation id :type federation\_id: int :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**assign\_deliveryservice\_servers\_by\_ids**(*data=None*)

Assign servers by id to a Delivery Service. (New Method) *deliveryserviceserver* :param data: The required data to create server associations to a delivery service :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**assign\_deliveryservice\_servers\_by\_names**(*xml\_id=None, data=None*)

Assign servers by name to a Delivery Service by xmlId. *deliveryservices/{{xml\_id}}/servers* :param xml\_id: The XML Id of the delivery service :type xml\_id: str :param data: The required data to assign servers to a delivery service :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]



**assign\_federation\_resolver\_to\_federations**(*federation\_id=None*,  
*data=None*)

Create one or more federation / federation resolver assignments. [federations/{{ID}}/federation\\_resolvers](#) :param federation\_id: The federation id :type federation\_id: int :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**assign\_parameter\_to\_profile\_ids**(*data=None*)

Create one or more parameter / profile assignments. [profileparameter](#) :param data: The data to assign :type data: Union[Dict[str, Any], List[Dict[str, Any]]] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**assign\_profile\_to\_parameter\_ids**(*data=None*)

Create one or more profile / parameter assignments. [profileparameter](#) :param data: The data to assign :type data: Union[Dict[str, Any], List[Dict[str, Any]]] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**associate\_paramater\_to\_profile**(*data=None*)

Associate parameter to profile. [profileparameters](#) :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**associate\_parameters\_by\_profile\_id**(*profile\_id=None*, *data=None*)

Associate Parameters to a Profile by Id. [profiles/{{ID}}/parameters](#) :param profile\_id: The profile id :type profile\_id: int :param data: The parameter data to associate :type data: Union[Dict[str, Any], List[Dict[str, Any]]] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**associate\_parameters\_by\_profile\_name**(*profile\_name=None*, *data=None*)

Associate Parameters to a Profile by Name. [profiles/name/{{name}}/parameters](#) :param profile\_name: The profile name :type profile\_name: str :param data: The parameter data to associate :type data: Union[Dict[str, Any], List[Dict[str, Any]]] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**property base\_url**

Returns the base url. (read-only)

**Returns** The base url should match '[w+-.]+://[w+-.]+(:d+)?' e.g

<https://to.somedomain.net/api/4.0/>

**Return type** str

**cachegroups\_queue\_update**(*cache\_group\_id=None*, *data=None*)

Queue Updates by Cache Group ID [cachegroups/{{ID}}/queue\\_update](#) :param cache\_group\_id: The Cache Group Id :type cache\_group\_id: int :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**cdns\_queue\_update**(*cdn\_id=None*, *data=None*)

Queue Updates by CDN Id. [cdns/{{ID}}/queue\\_update](#) :param cdn\_id: The CDN

Id :type cdn\_id: int :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**copy\_profile**(*new\_profile\_name=None, copy\_profile\_name=None, data=None*)  
Copy profile to a new profile. The new profile name must not exist *profiles/name/{{name}}/copy/{{copy}}* :param new\_profile\_name: The name of profile to copy to :type new\_profile\_name: String :param copy\_profile\_name: The name of profile copy from :type copy\_profile\_name: String :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_asn**(*data=None*)  
Create ASN *asns* :param data: The ASN data to use for ASN creation. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_cachegroups**(*data=None*)  
Create a Cache Group *cachegroups* :param data: The parameter data to use for cachegroup creation. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_cdn**(*data=None*)  
Create a new CDN. *cdns* :param data: The parameter data to use for cdn creation. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**create\_cdn\_dns\_sec\_keys**(*data=None*)  
Generates ZSK and KSK keypairs for a CDN and all associated Delivery Services *cdns/dnsseckey/generate* :param data: The parameter data to use for cachegroup creation. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**create\_coordinates**(*data=None*)  
Create coordinates *coordinates* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_deliveryservice**(*data=None*)  
Allows user to create a delivery service. *deliveryservices* :param data: The request data structure for the API request :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**create\_deliveryservice\_regexes**(*delivery\_service\_id=None, data=None*)  
Create a regex for a delivery service *deliveryservices/{{ID}}/regexes* :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :param data: The required data to create delivery service regexes :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_division**(data=None)

Create a division *divisions* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_federation**(data=None)

Allows a user to add federations for their delivery service(s). *federations* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_federation\_in\_cdn**(cdn\_name=None, data=None)

Create a federation. *cdns/{{name}}/federations* :param cdn\_name: The CDN name to find federation :type cdn\_name: String :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_federation\_resolver**(data=None)

Create a federation resolver. :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_federation\_user**(federation\_id=None, data=None)

Create one or more federation / user assignments. *federations/{{ID}}/users* :param federation\_id: Federation ID :type federation\_id: int :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_job**(data=None)

Creates a new content-invalidation job sorted by start time. *jobs* :param data: The content-invalidation job object that will be created. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_origins**(data=None)

Creates origins associated with a delivery service *origins* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_parameter**(data=None)

Create Parameter *parameters* :param data: The parameter(s) data to use for parameter creation. :type data: Union[Dict[str, Any], List[Dict[str, Any]]] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

#### **create\_profile**(data=None)

Create a profile *profiles* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises:

Union[LoginError, OperationError]

**create\_region**(*query\_params=None, data=None*)

Create a region *regions* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_role**(*data=None*)

Create a new Role. *roles* :param data: A new Role object to be created. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_server**(*data=None*)

Create a new Server. *servers* :param data: The parameter data to use for server creation :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_servercheck**(*data=None*)

Post a server check result to the serverchecks table. *servercheck* :param data: The parameter data to use for server creation :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_staticdnsentries**(*data=None*)

Create static DNS entries associated with the delivery service *staticdnsentries* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_tenant**(*data=None*)

Create a tenant *tenants* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_to\_extension**(*data=None*)

Creates a Traffic Ops extension. *servercheck/extensions* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_topology**(*data=None*)

Create a topology *topologies* :type data: Optional[Dict[str, Any]] :param data: The Topology data to use for Topology creation. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], Response] :raises: Union[LoginError, OperationError]

**create\_user**(*data=None*)

Create a user. *users* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str,

Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**create\_user\_with\_registration**(data=None)

Register a user and send registration email [users/register](#) :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_asn**(query\_params=None)

Delete ASN :to-api-asns-id: :param asn\_id: The ID of the ASN to delete :type asn\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_cache\_group\_parameters**(cache\_group\_id=None, parameter\_id=None)

Delete a cache group parameter association. This endpoint has been deprecated and will no longer be available as of TO API v4. [cachegroupparameters/{ID}/{parameterID}](#) :param cache\_group\_id: The cache group id in which the parameter will be deleted :type cache\_group\_id: int :param parameter\_id: The parameter id which will be disassociated :type parameter\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_cachegroups**(cache\_group\_id=None)

Delete a cache group [cachegroups/{ID}](#) :param cache\_group\_id: The cache group id to update :type cache\_group\_id: Integer :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_cdn\_by\_id**(cdn\_id=None)

Delete a CDN by Id. [cdns/{ID}](#) :param cdn\_id: The CDN id :type cdn\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_cdn\_dns\_sec\_keys**(cdn\_name=None)

Delete dnssec keys for a cdn and all associated delivery services [cdns/name/{name}/dnsseckey](#) :param cdn\_name: The CDN name to delete dnsseckey info for :type cdn\_name: String :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_coordinates**(query\_params=None)

Delete coordinates [coordinates](#) :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_deliveryservice\_by\_id**(delivery\_service\_id=None)

Allows user to delete a delivery service. [deliveryservices/{ID}](#) :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_deliveryservice\_regex\_by\_regex\_id**(*delivery\_service\_id=None, delivery\_service\_regex\_id=None*)

Delete a RegEx by Id for a Delivery Service by Id. [deliveryservices/{ID}/regexes/{rID}](#) :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :param delivery\_service\_regex\_id: The delivery service regex Id :type delivery\_service\_regex\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_deliveryservice\_servers\_by\_id**(*delivery\_service\_id=None, server\_id=None*)

Removes a server (cache) from a delivery service. [deliveryservice-server/{DSID}/{serverID}](#) :param delivery\_service\_id: The delivery service id :type delivery\_service\_id: int :param server\_id: The server id to remove from delivery service :type server\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_deliveryservice\_ssl\_keys\_by\_xml\_id**(*xml\_id=None, query\_params=None*)

Delete SSL keys for a Delivery Service by xmlId. [deliveryservices/xmlId/{XMLID}/sslkeys](#) :param xml\_id: The Delivery Service xmlId :type xml\_id: str :param query\_params: The url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_division**(*division\_id=None, query\_params=None*)

Delete a division by division id [divisions/{ID}](#) :param division\_id: The division id to delete :type division\_id: int :param query\_params: The required data to update delivery service regexes :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_federation\_in\_cdn**(*cdn\_name=None, federation\_id=None*)

Delete a federation. [cdns/{name}/federations/{ID}](#) :param cdn\_name: The CDN name to find federation :type cdn\_name: String :param federation\_id: The federation id :type federation\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_federation\_resolver**(*federation\_resolver\_id=None*)

Delete a federation resolver. :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_federation\_user**(*federation\_id=None, user\_id=None*)

Delete one or more federation / user assignments. [federations/{ID}/users/{userID}](#) :param federation\_id: Federation ID :type federation\_id: int :param user\_id: Federation User ID :type user\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]



**delete\_job**(*query\_params=None*)

Deletes a content-invalidation job. *jobs* :param query\_params: 'id' is a required parameter, identifying the job being deleted. :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_origins**(*query\_params=None*)

Updates origins associated with a delivery service *origins* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_parameter**(*parameter\_id=None*)

Delete Parameter *parameters/{ID}* :param parameter\_id: The parameter id to delete :type parameter\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_physical\_location**(*physical\_location\_id=None, query\_params=None*)

Delete Physical Location by id *phys\_locations/{ID}* :param physical\_location\_id: The id to delete :type physical\_location\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_profile\_by\_id**(*profile\_id=None*)

Delete Profile by Id. *profiles/{ID}* :param profile\_id: The profile Id :type profile\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_profile\_parameter\_association\_by\_id**(*profile\_id=None, parameter\_id=None*)

Delete Parameter association by Id for a Profile by Id. *profileparameters/{profileID}/{parameterID}* :param profile\_id: The profile id :type profile\_id: int :param parameter\_id: The parameter id :type parameter\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_region**(*query\_params=None*)

Delete a region by name or ID as a query parameter *regions/{ID}* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_role**(*query\_params=None*)

Delete a Role. *roles* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_server\_by\_id**(*server\_id=None*)

Delete a Server by Id. *servers/{ID}* :param server\_id: The server Id :type server\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_staticdnsentries**(*query\_params=None*)

Delete static DNS entries associated with the delivery service *staticdnsentries*  
:param query\_params: The optional url query parameters for the call :type  
query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str,  
Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**delete\_to\_extension**(*extension\_id=None*)

Deletes a Traffic Ops extension. *servercheck/extensions/{{ID}}* :param ex-  
tension\_id: The extension id to delete :type extension\_id: int :rtype:  
Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises:  
Union[LoginError, OperationError]

**delete\_topology**(*name=None*)

Delete a Topology *topologies* :type name: Optional[str] :param name: The  
name of the Topology to delete :type name: str :rtype: Tuple[Union[Dict[str, Any],  
List[Dict[str, Any]]], Response] :raises: Union[LoginError, OperationError]

**generate\_deliveryservice\_ssl\_keys**(*data=None*)

Generate an SSL certificate. (self-signed) *deliveryservices/sslkeys/generate* :param  
data: The parameter data to use for Delivery Service SSL key generation. :type  
data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises:  
Union[LoginError, OperationError]

**generate\_deliveryservice\_url\_signature\_keys**(*xml\_id=None*)

Generate URL Signature Keys for a Delivery Service by xmlId. *deliveryser-  
vices/xmlId/{{xml\_id}}/urlkeys/generate* :param xml\_id: The Delivery Service xm-  
lId :type xml\_id: str :rtype: Tuple[Dict[str, Any], requests.Response] :raises:  
Union[LoginError, OperationError]

**generate\_iso**(*data=None*)

Generate an ISO *isos* :param data: The update action. QueueUpdateRequest()  
can be used for this argument also. :type data: Dict[str, Any] :rtype:  
Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises:  
Union[LoginError, OperationError]

**get\_all\_cachegroup\_parameters**()

A collection of all cache group parameters. This endpoint has been deprecated and  
will no longer be available as of TO API v4. *cachegroupparameters* :rtype: Tu-  
ple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**get\_all\_deliveryservice\_servers**(\*args, \*\*kwargs)

Get all servers attached to all delivery services via the Traffic Ops API. :rtype:  
Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises:  
Union[LoginError, OperationError]

**get\_api\_capabilities**(*query\_params=None*)

Get all API-capability mappings *api\_capabilities* :rtype: Tuple[Union[Dict[str,  
Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, Oper-  
ationError]

**get\_asns**(*query\_params=None*)

Get ASNs. *asns* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], re-



quests.Response] :raises: Union[LoginError, OperationError]

#### **get\_authenticated\_user()**

Retrieves the profile for the authenticated user. [user/current](#) :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cache\_stats(query\_params=None)**

Retrieves statistics about the CDN. [cache\\_stats](#) :param query\_params: See API page for more information on accepted params :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cachegroup\_parameters(cache\_group\_id=None)**

Get a cache groups parameters. This endpoint has been deprecated and will no longer be available as of TO API v4. [cachegroups/{ID}/parameters](#) :param cache\_group\_id: The cache group Id :type cache\_group\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cachegroups(query\_params=None)**

Get Cache Groups. [cachegroups](#) :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_capabilities(query\_params=None)**

Retrieves capabilities [capabilities](#) :param query\_params: See API page for more information on accepted parameters :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cdn\_dns\_sec\_keys(cdn\_name=None)**

Gets a list of dnsseckey for a CDN and all associated Delivery Services [cdns/name/{name}/dnsseckey](#) :param cdn\_name: The CDN name to find dnsseckey info for :type cdn\_name: String :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cdn\_health\_by\_name(cdn\_name=None)**

Retrieves the health of all locations (cache groups) for a given CDN [cdns/{name}/health](#) :param cdn\_name: The CDN name to find health for :type cdn\_name: String :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cdn\_monitoring\_info(cdn\_name=None)**

Retrieves CDN monitoring information [cdns/{name}/configs/monitoring](#) :param cdn\_name: The CDN name to find configs for :type cdn\_name: String :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

#### **get\_cdn\_ssl\_keys(cdn\_name=None)**

Returns ssl certificates for all Delivery Services that are a part of the CDN. [cdns/name/{name}/sslkeys](#) :param cdn\_name: The CDN name to find ssl keys for :type cdn\_name: String :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_cdns(*query\_params=None*)**

Get all CDNs. *cdns* :param query\_params: See API page for more information on accepted parameters :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_cdns\_capacity()**

Retrieves the aggregate capacity percentages of all locations (cache groups) for a given CDN. *cdns/capacity* :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_cdns\_domains()**

Retrieves the different CDN domains *cdns/domains* :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_cdns\_health()**

Retrieves the health of all locations (cache groups) for all CDNs *cdns/health* :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_cdns\_routing()**

Retrieves the aggregate routing percentages of all locations (cache groups) for a given CDN. *cdns/routing* :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_change\_logs(*query\_params=None*)**

Retrieve all change logs from traffic ops *logs* :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_change\_logs\_newcount()**

Get amount of new logs from traffic ops *logs/newcount* :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_coordinates(*query\_params=None*)**

Get all coordinates associated with the cdn *coordinates* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_current\_snapshot\_crconfig(*cdn\_name=None*)**

Retrieves the CURRENT snapshot for a CDN which doesn't necessarily represent the current state of the CDN. The contents of this snapshot are currently used by Traffic Monitor and Traffic Router. *cdns/{name}/snapshot* :param cdn\_name: The CDN name :type cdn\_name: str :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_delivery\_service\_capacity(*delivery\_service\_id=None*)**

Retrieves the capacity percentages of a delivery service. Delivery service must be assigned to user if user is not admin or operations. *deliveryservices/{ID}/capacity* :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_delivery\_service\_health**(*delivery\_service\_id=None*)

Retrieves the health of all locations (cache groups) for a delivery service. Delivery service must be assigned to user if user is not admin or operations. [delivery-services/{ID}/health](#) :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_delivery\_service\_stats**(*query\_params=None*)

Retrieves statistics on the delivery services. [deliveryservice\\_stats](#) :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryservice\_ineligible\_servers**(*delivery\_service\_id=None*)

Retrieves properties of CDN EDGE or ORG servers not eligible for assignment to a delivery service. [deliveryservices/{ID}/servers/eligible](#) :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryservice\_regexes\_by\_id**(*delivery\_service\_id=None,*  
*query\_params=None*)

Get RegExes for a Delivery Service by Id. [deliveryservices/{ID}/regexes](#) :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :param query\_params: The url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryservice\_servers**(*delivery\_service\_id=None*)

Retrieves properties of CDN EDGE or ORG servers assigned to a delivery service. [deliveryservices/{ID}/servers](#) :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryservice\_ssl\_keys\_by\_xml\_id**(*xml\_id=None,*  
*query\_params=None*)

Get SSL keys for a Delivery Service by xmlId. [deliveryservices/xmlId/{XMLID}/sslkeys](#) :param xml\_id: The Delivery Service XML id :type xml\_id: str :param query\_params: The url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryservices**(*query\_params=None*)

Retrieves all delivery services (if admin or ops) or all delivery services assigned to user. [deliveryservices](#) :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryservices\_regexes**()

Get RegExes for all Delivery Services. [deliveryservices\\_regexes](#) :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_deliveryserviceserver**(*query\_params=None*)

Retrieves delivery service / server assignments. (Allows pagination and limits) *deliveryserviceserver* :param query\_params: The required url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_divisions**(*query\_params=None*)

Get all divisions. *divisions* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_federation\_delivery\_services**(*federation\_id=None*)

Retrieves delivery services assigned to a federation *federations/{ID}/deliveryservices* :param federation\_id: The federation id :type federation\_id: int :param federation\_id: The federation id :type federation\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_federation\_resolvers**(*query\_params=None*)

Get federation resolvers. *federation\_resolvers* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_federation\_resolvers\_by\_id**(*federation\_id=None*)

*federations/{ID}/federation\_resolvers* Retrieves federation resolvers assigned to a federation :param federation\_id: The federation id :type federation\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_federation\_users**(*federation\_id=None*)

Retrieves users assigned to a federation. *federations/{ID}/users* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_federations**()

Retrieves a list of federation mappings (aka federation resolvers) for a the current user *federations* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_federations\_for\_cdn**(*cdn\_name=None, query\_params=None*)

Retrieves a list of federations for a cdn. *cdns/{name}/federations* :param cdn\_name: The CDN name to find federation :type cdn\_name: String :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_jobs**(*query\_params=None*)

Get all content-invalidation jobs (tenancy permitting). *jobs* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_origins**(*query\_params=None*)

Get origins associated with the delivery service *origins* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_osversions**()

Get all OS versions for ISO generation and the directory where the kickstarter files are found. The values are retrieved from osversions.json found in either /var/www/files or in the location defined by the kickstart.files.location parameter (if defined). *osversions* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_parameters**(*query\_params=None*)

Get all Parameters. *parameters* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_parameters\_by\_profile\_id**(*profile\_id=None*)

Get all Parameters associated with a Profile by Id. *profiles/{ID}/parameters* :param profile\_id: The profile Id :type profile\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_parameters\_by\_profile\_name**(*profile\_name=None*)

Get all Parameters associated with a Profile by Name. *profiles/name/{name}/parameters* :param profile\_name: The profile name :type profile\_name: str :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_pending\_snapshot\_crconfig**(*cdn\_name=None*)

Retrieves a PENDING snapshot for a CDN which represents the current state of the CDN. The contents of this snapshot are NOT currently used by Traffic Monitor and Traffic Router. Once a snapshot is performed, this snapshot will become the CURRENT snapshot and will be used by Traffic Monitor and Traffic Router. *cdns/{name}/snapshot/new* :param cdn\_name: The CDN name :type cdn\_name: str :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**get\_physical\_locations**(*query\_params=None*)

Get Physical Locations. *phys\_locations* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_plugins**()

Retrieves the list of plugins. *plugins* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_profiles**(*query\_params=None*)

Get Profiles. *profiles* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_regions**(*query\_params=None*)

Get Regions. *regions* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_roles**()

Get Roles. *roles* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_server\_delivery\_services**(*server\_id=None*)

Retrieves all delivery services assigned to the server *servers/{ID}/deliveryservices* :param server\_id: The server id to retrieve :type server\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_server\_details**(*name=None*)

Get servers/details *servers/details* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

Deprecated since version 3.0: The endpoint this represents has been removed from APIv4 and clients should use get\_servers instead.

### **get\_server\_update\_status**(*server\_name=None*)

Gets the current update status of a server named *server\_name*. *servers/{hostname}/update\_status* :param server\_name: The (short) hostname of the server for which the update status will be fetched :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_servercheck\_extensions**()

Retrieves the list of extensions. *servercheck/extensions* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_servers**(*query\_params=None*)

Get Servers. *servers* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_staticdnsentries**(*query\_params=None*)

Get static DNS entries associated with the delivery service *staticdnsentries* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_statuses**(*query\_params=None*)

Retrieves a list of the server status codes available. *statuses* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

### **get\_system\_info**()

Get information on the traffic ops system. *system/info* :rtype: Tuple[Union[Dict[str,



Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_tenants**(*query\_params=None*)

Get all tenants. *tenants* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_topologies**(*query\_params=None*)

Get Topologies. *topologies* :type query\_params: Optional[Dict[str, Any]] :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], Response] :raises: Union[LoginError, OperationError]

**get\_traffic\_monitor\_cache\_stats**()

Retrieves cache stats from Traffic Monitor. Also includes rows for aggregates *caches/stats* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_types**(*query\_params=None*)

Get Data Types. *types* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_user\_by\_id**(*user\_id=None*)

Retrieves user by ID. *users/{ID}* :param user\_id: The user to retrieve :type user\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**get\_users**()

Retrieves all users. *users* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**property logged\_in**

Read-only property of to determine if user is logged in to Traffic Ops. :return: True if connected and logged in, False otherwise :rtype: bool

**login**(*username, password*)

Login to the Traffic Ops API. :param username: Traffic Ops User Name :type username: str :param password: Traffic Ops User Password :type password: str :return: None :rtype: None :raises: LoginError

**replace\_authenticated\_user**(*data=None*)

Updates the currently authenticated user. *user/current* :param data: The new user information which will replace the current user's user information. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**servers\_queue\_update**(*server\_id=None, data=None*)

Queue Updates by Server Id. *servers/{ID}/queue\_update* :param server\_id: The server Id :type server\_id: int :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**snapshot\_crconfig**(*query\_params=None*)

Snapshot CRConfig by CDN Name or ID. *snapshot* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**property to\_url**

The URL without the api portion. (read-only)

**Returns** The URL should match '[w+-.]+://[w+-.]+(:d+)?' e.g <https://to.somedomain.net> or <https://to.somedomain.net:443>

**Return type** str

**topologies\_queue\_update**(*name=None, data=None*)

Queue Updates by Topology name. *topologies/{name}/queue\_update* :param name: The Topology name :param data: The update action. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**update\_asn**(*query\_params=None*)

Update ASN *asns/{id}* :param asn\_id: The ID of the ASN to update :type asn\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_cachegroups**(*cache\_group\_id=None, data=None*)

Update a cache group *cachegroups/{ID}* :param cache\_group\_id: The cache group id to update :type cache\_group\_id: Integer :param data: The parameter data to use for cachegroup creation. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_cdn\_by\_id**(*cdn\_id=None, data=None*)

Update a CDN by Id. *cdns/{ID}* :param cdn\_id: The CDN id :type cdn\_id: int :param data: The parameter data to use for cdn update. :type data: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response] :raises: Union[LoginError, OperationError]

**update\_coordinates**(*query\_params=None, data=None*)

Update coordinates *coordinates* :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_deliveryservice\_by\_id**(*delivery\_service\_id=None, data=None*)

Update a Delivery Service by Id. *deliverservices/{ID}* :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int :param data: The request data structure for the API request :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]



**update\_deliveryservice\_regexes**(*delivery\_service\_id=None, regex\_id=None, query\_params=None*)

Update a regex for a delivery service *deliveryservices/{{ID}}/regexes/{{rID}}*  
 :param delivery\_service\_id: The delivery service Id :type delivery\_service\_id: int  
 :param regex\_id: The delivery service regex id :type regex\_id: int  
 :param query\_params: The required data to update delivery service regexes :type query\_params: Dict[str, Any] :rtype: Tuple[Dict[str, Any], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_deliveryservice\_safe**(*delivery\_service\_id=None, data=None*)

Allows a user to edit limited fields of a Delivery Service. *deliveryservices/{{ID}}/safe*  
 :param delivery\_service\_id: The Delivery Service Id :type delivery\_service\_id: int  
 :param data: The request data structure for the API request :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_division**(*division\_id=None, query\_params=None*)

Update a division by division id *divisions/{{ID}}* :param division\_id: The division id to update :type division\_id: int  
 :param query\_params: The required data to update delivery service regexes :type query\_params: Dict[str, Any]  
 :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_federation\_in\_cdn**(*cdn\_name=None, federation\_id=None, query\_params=None*)

Update a federation. *cdns/{{name}}/federations/{{ID}}* :param cdn\_name: The CDN name to find federation :type cdn\_name: String  
 :param federation\_id: The federation id :type federation\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_job**(*data=None, query\_params=None*)

Replaces a content-invalidation job with the one passed. :param data: The content-invalidation job with which the identified job will be replaced. :type data: Dict[str, Any]  
 :param query\_params: 'id' is a required parameter, identifying the job being updated. *jobs* :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_origins**(*query\_params=None*)

Updates origins associated with a delivery service *origins* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any]  
 :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_parameter**(*parameter\_id=None, query\_params=None*)

Update Parameter *parameters/{{ID}}* :param parameter\_id: The parameter id to update :type parameter\_id: int :rtype: Tuple[Dict[str, Any], requests.Response]  
 :raises: Union[LoginError, OperationError]

**update\_physical\_location**(*physical\_location\_id=None, query\_params=None*)  
 Update Physical Location by id *phys\_locations/{{ID}}* :param physical\_location\_id: The id to update :type physical\_location\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_profile\_by\_id**(*profile\_id=None, data=None*)  
 Update Profile by Id. *profiles/{{ID}}* :param profile\_id: The profile Id :type profile\_id: int :param data: The parameter data to edit :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_region**(*region\_id=None*)  
 Update a region *regions/{{ID}}* :param region\_id: The region to update :type region\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_role**(*data=None, query\_params=None*)  
 Get Roles. *roles* :param data: A new Role object which will replace the one identified. :type data: Dict[str, Any] :param query\_params: 'id' is a required parameter, defining the Role to be replaced. :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_server\_by\_id**(*server\_id=None, data=None*)  
 Update a Server by Id. *servers/{{ID}}* :param server\_id: The server Id :type server\_id: int :param data: The parameter data to edit :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_server\_status\_by\_id**(*server\_id=None, data=None*)  
 Update server\_status by Id. *servers/{{ID}}/status* :param server\_id: The server Id :type server\_id: int :status: <https://traffic-control-cdn.readthedocs.io/en/latest/api/server.html> :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_staticdnsentries**(*data=None, query\_params=None*)  
 Update static DNS entries associated with the delivery service *staticdnsentries* :param data: The update action. QueueUpdateRequest() can be used for this argument also. :type data: Dict[str, Any] :param query\_params: The optional url query parameters for the call :type query\_params: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_tenant**(*tenant\_id=None*)  
 Update a tenant *tenants/{{ID}}* :param tenant\_id: The tenant to update :type tenant\_id: int :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**update\_topology**(*name=None, data=None*)  
 Update a Topology *topologies* :type name: Optional[str] :param name: The name of the Topology :type name: str :type data: Optional[Dict[str, Any]] :param data: The new values for the Topology :type data: Dict[str, Any]

:rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], Response] :raises: Union[LoginError, OperationError]

**update\_user\_by\_id**(*user\_id=None, data=None*)

Update a user. *users* :param data: The user update data payload. :type data: Dict[str, Any] :rtype: Tuple[Union[Dict[str, Any], List[Dict[str, Any]]], requests.Response] :raises: Union[LoginError, OperationError]

**trafficops.api\_request**(*method\_name, api\_path, supported\_versions*)

This wrapper returns a decorator that routes the calls to the appropriate utility function that generates the RESTful API endpoint, performs the appropriate call to the endpoint and returns the data to the user.

#### Parameters

- **method\_name** (*str*) – A method name defined on the Class, this decorator is decorating, that will be called to perform the operation. E.g. ‘GET’, ‘POST’, ‘PUT’, ‘DELETE’, etc. The method\_name chosen must have the signature of <method>(self, api\_path, \*\*kwargs) e.g. def get(self, api\_path, \*\*kwargs): ...
- **api\_path** (*str*) – The path to the API end-point that you want to call which does not include the base url e.g. user/login, servers, etc. This string can contain substitution parameters as denoted by a valid field\_name replacement field specification as per str.format() e.g. cachegroups/{id} or cachegroups/{id:d}
- **supported\_versions** (*Tuple[str]*) – A tuple of API versions that this route supports

**Returns** rtype int: A new function that replaces the original function with a boilerplate execution process.

**Return type** Callable[str, Dict[str, Any]]

## utils

Useful utility methods

**trafficops.utils.log\_with\_debug\_info**(*logging\_level=20, msg="", parent=False, separator=':'*)

Uses inspect module(reflection) to gather debugging information for the source file name, function name, and line number of the calling function/method.

#### Parameters

- **logging\_level** (*int*) – The logging level from the logging module constants E.g. logging.INFO, logging.DEBUG, etc.
- **msg** (*Text*) – The message to log.
- **parent** (*bool*) – If True, use the caller’s parent information instead of the caller’s information in the message.
- **separator** (*Text*) – The string to use for the component separator

**Returns** ‘<file name>:<function name>:<line number>: <msg>’ e.g. ‘tosession.py:\_build\_endpoint:199: This is a message to log.’

**Return type** Text

### Versioning

The `trafficops.__version__` module contains only the `__version__` “constant” which gives the version of this *Apache-TrafficControl package* and **not** the version of *Apache Traffic Control* for which it was made. The two are versioned separately, to allow the client to grow in a version-controlled manner without being tied to the release cadence of Apache Traffic Control as a whole.

Version 3.0 is supported for use with Apache Traffic Control versions 5.1 and 6.1 (release pending at the time of this writing). New functionality will be added as the *Traffic Ops API* evolves, but changes to this client will remain non-breaking for existing code using it until the next major version is released.

### 6.1.3 toaccess

#### toaccess

This module provides a set of functions meant to provide ease-of-use functionality for interacting with the Traffic Ops API. It provides scripts named `tomethod` where *method* is the name of an HTTP method (in lowercase). Collectively they are referred to as **toaccess** Implemented methods thus far are:

- delete
- head
- get
- options
- patch
- post
- put

### Arguments and Flags

#### PATH

This is the request path. By default, whatever is passed is considered to be relative to `/api/api-version/` where `api-version` is `--api-version`. This behavior can be disabled by using `--raw-path`.

#### DATA

An optional positional argument that is a data payload to pass to the Traffic Ops server in the request body. If this is the absolute or relative path to a file, the contents of the file will instead be read and used as the request payload.

- h, --help**  
Print usage information and exit
- a API\_VERSION, --api-version API\_VERSION**  
Specifies the version of the Traffic Ops API that will be used for the request. Has no effect if `--raw-path` is used. (Default: 2.0)
- f, --full**  
Output the full HTTP exchange including request method line, request headers, request body (if any), response status line, and response headers (as well as the response body, if any). This is equivalent to using `--request-headers`, `--request-payload`, and `--response-headers` at the same time, and those options will have no effect if given. (Default: false)
- k, --insecure**  
Do not verify SSL certificates - typically useful for making requests to development or testing servers as they frequently have self-signed certificates. (Default: false)
- p, --pretty**  
Pretty-print any payloads that are output as formatted JSON. Has no effect on plaintext payloads. Uses tab characters for indentation. (Default: false)
- r, --raw-path**  
Request exactly `PATH`; do not preface the request path with `/api/api_version`. This effectively means that `--api-version` will have no effect. (Default: false)
- v, --version**  
Print version information and exit.
- request-headers**  
Output the request method line and any and all request headers. (Default: false)
- request-payload**  
Output the request body if any was sent. Will attempt to pretty-print the body as JSON if `--pretty` is used. (Default: false)
- response-headers**  
Output the response status line and any and all response headers. (Default: false)
- to-url URL**  
The FQDN and optionally the port and scheme of the Traffic Ops server. This will override `TO_URL`. The format is the same as for `TO_URL`. (Default: uses the value of `TO_URL`)
- to-password PASSWORD**  
The password to use when authenticating to Traffic Ops. Overrides `TO_PASSWORD`. (Default: uses the value of `TO_PASSWORD`)
- to-user USERNAME**  
The username to use when connecting to Traffic Ops. Overrides `TO_USER`. (Default: uses the value of `TO_USER`)

### Environment Variables

If defined, **toaccess** scripts will use the `TO_URL`, `TO_USER`, and `:envvar`TO_PASSWORD`` environment variables to define their connection to and authentication with the Traffic Ops server. Typically, setting these is easier than using the long options `--to-url`, `--to-user`, and `--to-password` on every invocation.

### Exit Codes

The exit code of a **toaccess** script can sometimes be used by the caller to determine what the result of calling the script was without needing to parse the output. The exit codes used are:

- 0** The command executed successfully, and the result is on STDOUT.
- 1** Typically this exit code means that an error was encountered when parsing positional command line arguments. However, this is also the exit code used by most Python interpreters to signal an unhandled exception.
- 2** Signifies a runtime error that caused the request to fail - this is **not** generally indicative of an HTTP client or server error, but rather an underlying issue connecting to or authenticating with Traffic Ops. This is distinct from an exit code of 32 in that the *format* of the arguments was correct, but there was some problem with the *value*. For example, passing `https://test:` to `--to-url` will cause an exit code of 2, not 32.
- 4** An HTTP client error occurred. The HTTP stack will be printed to stdout as indicated by other options - meaning by default it will only print the response payload if one was given, but will respect options like e.g. `--request-payload` as well as `-p/--pretty`.
- 5** An HTTP server error occurred. The HTTP stack will be printed to stdout as indicated by other options - meaning by default it will only print the response payload if one was given, but will respect options like e.g. `--request-payload` as well as `-p/--pretty`.
- 32** This is the error code emitted by Python's `argparse` module when the passed arguments could not be parsed successfully.

---

**Note:** The way exit codes 4 and 5 are implemented is by returning the status code of the HTTP request divided by 100 whenever it is at least 400. This means that if the Traffic Ops server ever started returning e.g. 700 status codes, the exit code of the script would be 7.

---

### Module Reference

`to_access.delete()`

Entry point for **todelete**

**Returns** The program's exit code

`to_access.get()`

Entry point for **toget**

**Returns** The program's exit code

`to_access.head()`

Entry point for **tohead**

**Returns** The program's exit code

`to_access.options()`

Entry point for **tooptions**

**Returns** The program's exit code

`to_access.output(r, pretty, request_header, response_header, request_payload,  
indent='\t')`

Prints the passed response object in a format consistent with the other parameters.

**Parameters**

- **r** – The requests response object being printed
- **pretty** – If True, attempt to pretty-print payloads as JSON
- **request\_header** – If True, print request line and request headers
- **response\_header** – If True, print response line and response headers
- **request\_payload** – If True, print the request payload
- **indent** – An optional number of spaces for pretty-printing indentation (default is the tab character)

`to_access.parse_arguments(program)`

A common-use function that parses the command line arguments.

**Parameters** **program** – The name of the program being run - used for usage informational output

**Returns** The Traffic Ops HTTP session object, the requested path, any data to be sent, an output format specification, whether or not the path is raw, and whether or not output should be prettified

`to_access.patch()`

Entry point for **topatch**

**Returns** The program's exit code

`to_access.post()`

Entry point for **topost**

**Returns** The program's exit code

`to_access.put()`

Entry point for **toput**

**Returns** The program's exit code

`to_access.request(method)`

All of the scripts wind up calling this function to handle their common functionality.

**Parameters** **method** – The name of the request method to use (case-insensitive)

**Returns** The program's exit code



## 6.1.4 Traffic Vault Util (Riak)

Deprecated since version ATCv6: When support for the Riak backend is removed, support for this tool will also be dropped.

The `traffic_vault_util` tool - located at `tools/traffic_vault_util.go` in the [Apache Traffic Control repository](#) - is used to view and modify the contents of a Traffic Vault Riak cluster. The tool contains basic operations to display the buckets, keys and values stored within Riak.

---

**Note:** This tool does not apply to the PostgreSQL Traffic Vault backend.

---

`traffic_vault_util` also has a small converter utility to perform a one-off conversion of key formats within the SSL bucket. This conversion is useful when moving from an older version of Traffic Ops to the current version. In the older version, SSL records were indexed by *Delivery Service* database ID. Currently, SSL records are indexed by *Delivery Service* `xml_id`.

### Usage

```
traffic_vault_util [--dry_run] --vault_ip IP --vault_action ACTION
[--vault_user USER] [--vault_password PASSWD] [--vault_port PORT]
[--insecure]
```

#### **--dry\_run**

An optional flag which, if given, will cause **traffic\_vault\_util** to not write changes, but merely print what *would* be done in a real run.

#### **--vault\_action ACTION**

Defines the action to be performed. Available actions are:

**list\_buckets** Lists the “buckets” in the Riak cluster used by Traffic Vault

**list\_keys** Lists all the keys in all the buckets in the Riak cluster used by Traffic Vault

**list\_values** Lists all the values of all the keys in all the buckets in the Riak cluster used by Traffic Vault

**convert\_ssl\_to\_xmlid** Changes the key of all records in all buckets that start with “ds” into the `xml_id` of the *Delivery Service* for which we assume the record was created.

#### **--vault\_ip IP**

Either the IP address or FQDN of the Traffic Vault instance with which **traffic\_vault\_util** will interact.

**Warning:** If this IP address or FQDN does not point to a real Riak cluster, **traffic\_vault\_util** will print an error message to STDOUT, but *will not terminate*. Instead, it will try forever to query the server to which it failed to connect, consuming large amounts of CPU usage all the while<sup>1</sup>.

---

<sup>1</sup> These problems are all tracked by [GitHub Issue #3261](#).



**--vault\_password** PASSWD

An optional flag used to specify the password of the user defined by `--vault_user` when authenticating with Traffic Vault's Riak cluster.

**Warning:** Although this flag is optional, the utility will not work without it. It will try, but it will fail<sup>Page 1608, 1</sup>.

**--vault\_port** PORT

An optional flag which, if given, sets the port to which `traffic_vault_util` will try to connect to Riak. Default: 8087

**--vault\_user** USER

An optional flag which, if given, specifies the name of the user as whom to connect to Riak.

**Warning:** Although this flag is optional, the utility will not work without it. It will try, but it will fail<sup>Page 1608, 1</sup>.

**--insecure**

An optional flag which, if given, specifies whether to utilize TLS certificate checks when establishing a connection. Defaults to false.



## 7.1 FAQ

### 7.1.1 Who is using Traffic Control?

- [Comcast Cable](#) is the original developer of Traffic Control and is using it for all it's IP video delivery, delivering images and software to it's X1 platform, and for [delivering third party content](#) to it's footprint.
- [Cox Communications](#)
- [Cisco](#) has a product called [Open Media Distribution](#) that is based on Traffic Control.
- [Concurrent](#) has a product that uses Traffic Control, see their [github page](#) for more info.
- [Augere Pakistan / QUBEE](#)
- [Qwilt](#) has a product called [Open Edge CDN](#) that is based on Traffic Control.

### 7.1.2 How do I get help with Traffic Control?

Join us at [#traffic-control](#) on the [ASF Slack](#) or send your questions to our [mailing list](#). Slack is usually the best forum for quick Q&A type issues – like when you are getting a Traffic Control CDN up and running or if you have a question about configuration. Any discussions that could potentially lead to decisions being made about the project – like a new feature – should happen on [the developer mailing list](#).

### 7.1.3 How do I get involved with the development of Traffic Control?

See the [CONTRIBUTING.md](#) file.

### 7.1.4 What is Rascal?

Rascal was the original name for Traffic Monitor. You will sometimes still see this name in the source, or in older documents.

### 7.1.5 What is the CCR?

CCR (Comcast Content Router) was the original name for Traffic Router. You will sometimes still see this name in the source, or in older documents.

### 7.1.6 What is Twelve Monkeys?

Twelve Monkeys was the the original internal name for Traffic Ops. You will sometimes still see this name in the source, or in older documents. It's also a good movie.

### 7.1.7 What license is Traffic Control released under?

See the [LICENSE](#) file.

## INDICES AND TABLES

### 8.1 Glossary

**302 content routing** *HTTP Content Routing*.

**ACME Account** An account previously created with an ACME provider.

**astats (stats\_over\_http)** An ATS plugin that allows you to monitor vitals of the ATS server.  
See *Cache Monitoring*.

#### Cache Group

**Cache Groups** A group of caching HTTP proxy servers that together create a combined larger cache using consistent hashing. Traffic Router treats all servers in a *Cache Group* as though they are in the same geographic location, though they are in fact only in the same general area. A *Cache Group* has one single set of geographical coordinates even if the *cache servers* that make up the *Cache Group* are actually in *Physical Locations*. The *cache servers* in a *Cache Group* are not aware of the other *cache servers* in the group - there is no clustering software or communications between *cache servers* in a *Cache Group*.

There are two basic types of *Cache Groups*: EDGE\_LOC and MID\_LOC (“LOC” being short for “location” - a holdover from when *Cache Groups* were called “Cache Locations”). Traffic Control is a two-tiered system, where the clients get directed to the Edge-tier (EDGE\_LOC) *Cache Group*. On cache miss, the *cache server* in the Edge-tier *Cache Group* obtains content from a Mid-tier (MID\_LOC) *Cache Group*, rather than the origin, which is shared with multiple Edge-tier *Cache Groups*. Edge-tier *Cache Groups* are usually configured to have a single “parent” *Cache Group*, but in general Mid-tier *Cache Groups* have many “children”.

---

**Note:** Often the Edge-tier to Mid-tier relationship is based on network distance, and does not necessarily match the geographic distance.

---

#### See also:

A *Cache Group* serves a particular part of the network as defined in the *Coverage Zone File* (or *Deep Coverage Zone File*, when applicable).

### See also:

For a more complete description of Cache Groups, see the [Cache Groups](#) overview section.

## Cache Server

### cache server

**cache servers** The main function of a CDN is to proxy requests from clients to [origin servers](#) and cache the results. To proxy, in the CDN context, is to obtain content using HTTP from an [origin server](#) on behalf of a client. To cache is to store the results so they can be reused when other clients are requesting the same content. There are three types of proxies in use on the Internet today:

- [reverse proxy](#): Used by Traffic Control for Edge-tier *cache servers*.
- [forward proxy](#): Used by Traffic Control for Mid-tier *cache servers*.
- transparent proxy: These are not used by Traffic Control. If you are interested you can learn more about transparent proxies on [wikipedia](#).

## Content Invalidation Job

### Content Invalidation Jobs

#### job

**jobs** *Content Invalidation Jobs* are a way to force [cache servers](#) to treat their cached content as stale (or even not in cache at all).

### See also:

For a more complete description of Content Invalidation Jobs, see the [Content Invalidation Jobs](#) overview section.

**content routing** Directing clients (or client systems) to a particular location or device in a location for optimal delivery of content See also [HTTP Content Routing](#) and [DNS Content Routing](#).

## Coverage Zone File

**Coverage Zone Map** The CZM (Coverage Zone Map) or CZF is a file that maps network prefixes to [Cache Groups](#). Traffic Router uses the CZM to determine what [Cache Group](#) is closest to the client. If the client IP address is not in this CZM, it falls back to geographic mapping, using a [MaxMind GeoIP2 database](#) to find the client's location, and the geographic coordinates from Traffic Ops for the [Cache Group](#). Traffic Router is inserted into the HTTP retrieval process by making it the authoritative DNS server for the domain of the CDN [Delivery Service](#). In the example of the [reverse proxy](#), the client was given the `http://www-origin-cache.cdn.com/foo/bar/fun.html` URL. In a Traffic Control CDN, URLs start with a routing name, which is configurable per [Delivery Service](#), e.g. `http://foo.mydeliveryservice.cdn.com/fun/example.html` with the chosen routing name `foo`.

1088: Example Coverage Zone File

```
{ "coverageZones": {
  "cache-group-01": {
    "network6": [
      "1234:5678::/64",
      "1234:5679::/64"
    ],
    "network": [
      "192.168.8.0/24",
      "192.168.9.0/24"
    ]
  }
}}
```

## Deep Coverage Zone File

**Deep Coverage Zone Map** The DCZF or DCZM (Deep Coverage Zone Map) maps network prefixes to “locations” - almost like the *Coverage Zone File*. Location names must be unique, and within the file are simply used to group *Edge-tier cache servers*. When a mapping is performed by Traffic Router, it will only look in the DCZF if the *Delivery Service* to which a client is being directed makes use of *Deep Caching*. If the client’s IP address cannot be matched by entries in this file, Traffic Router will first fall back to the regular *Coverage Zone File*. Then, failing that, it will perform geographic mapping using a database provided by the *Delivery Service*’s *Geolocation Provider*.

1089: Example Deep Coverage Zone File

```
{ "deepCoverageZones": {
  "location-01": {
    "network6": [
      "1234:5678::/64",
      "1234:5679::/64"
    ],
    "network": [
      "192.168.8.0/24",
      "192.168.9.0/24"
    ],
    "caches": [
      "edge"
    ]
  }
}}
```

## Delivery Service

**Delivery Services** *Delivery Services* are often referred to as a *reverse proxy* “remap rule” that exists on Edge-tier *cache servers*. In most cases, a *Delivery Service* is a one-to-one mapping to an FQDN that is used as a hostname to deliver the content. Many options and

settings regarding how to optimize the content delivery exist, which are configurable on a *Delivery Service* basis. Some examples of these *Delivery Service* settings are:

- Cache in RAM, cache on disk, or do not cache at all.
- Use DNS or HTTP Content routing.
- Limits on transactions per second and bandwidth.
- Protocol (HTTP or HTTPS).
- Token-based authentication settings.
- Header rewrite rules.

Since Traffic Control version 2.1, *Delivery Services* can optionally be linked to a *Profile*, and have *Parameters* associated with them. One example of a feature that uses *Delivery Service Parameters* is the *Use Multi-Site Origin Feature* configuration. *Delivery Services* are also for use in allowing multiple *Tenants* to coexist in a Traffic Control CDN without interfering with each other, and to keep information about their content separated.

**See also:**

See *Delivery Services* for a more in-depth explanation of *Delivery Services*.

### Delivery Service Request

### Delivery Service Requests

### DSR

**DSRs** A *Delivery Service Request* is the result of attempting to modify a *Delivery Service* when `dsRequests.enabled` is set to `true` in `traffic_portal_properties.json`. See *Delivery Service Requests* for more information.

**See also:**

See *Delivery Service Requests* for a more in-depth explanation of *Delivery Service Requests*, including their data model. See *Delivery Service Requests* for more information on how to use *Delivery Service Requests* in Traffic Portal.

**Delivery Service required capabilities** *Delivery Services required capabilities* are capabilities, which correlate to server capabilities, that are required in order to assign a server to a delivery service.

### Division

**Divisions** A group of *Regions*.

### Edge

### Edge-tier

### Edge-Tier

### Edge-tier cache

### Edge-tier caches

### Edge-tier cache server



**Edge-tier cache servers** Closest to the client or end-user. The edge tier is the tier that serves the client, edge caches are caches in the edge tier. In a Traffic Control CDN the basic function of the edge cache is that of a *reverse proxy*.

## Federation

**Federations** *Federations* allow for other (“federated”) CDNs (e.g. at a different ISP (Internet Service Provider)) to add a list of DNS resolvers and an FQDN to be used in a DNS CNAME record for a *Delivery Service*. When a request is made from one of the federated CDN’s clients, Traffic Router will return the CNAME record configured from the federation mapping. This allows the federated CDN to serve the content without the content provider changing the URL, or having to manage multiple URLs. For example, if the external CDN was actually another ATC-managed CDN, then a federation mapping to direct clients toward it should use the FQDN of a *Delivery Service* on the external CDN.

Federations only have meaning to DNS-routed *Delivery Services* - HTTP-routed Delivery services should instead treat the external FQDN as an *Origin* to achieve the same effect.

### See also:

Federations are currently only manageable by directly using the *Traffic Ops API*. The endpoints related to federations are *federations*, *federation\_resolvers*, *federations/{{ID}}/deliveryservices*, *federations/{{ID}}/deliveryservices/{{dsID}}*, *federations/{{ID}}/federation\_resolvers*, *federations/{{ID}}/users*, and *federations/{{ID}}/users/{{userID}}*.

## First-tier

### First-tier cache

### First-tier caches

### First-tier cache server

**First-tier cache servers** Closest to the client or end-user. The first tier in a *Topology* is the tier that serves the client, similar to the *Edge-tier*.

## forward proxy

**forward proxies** A forward proxy acts on behalf of the client such that the *origin server* is (potentially) unaware of the proxy’s existence. All Mid-tier *cache servers* in a Traffic Control based CDN are *forward proxies*. In a *forward proxy* scenario, the client is explicitly configured to use the the proxy’s IP address and port as a *forward proxy*. The client always connects to the *forward proxy* for content. The content provider does not have to change the URL the client obtains, and is (potentially) unaware of the proxy in the middle.

### See also:

[ATS documentation on forward proxy](#).

If a client uses a *forward proxy* to request the URL `http://www.origin.com/foo/bar/fun.html` the resulting chain of events follows.

1. To retrieve `http://www.origin.com/foo/bar/fun.html`, the client sends an HTTP request to the *forward proxy*.

1090: Client Requests Content from its *Forward Proxy*

```
GET http://www.origin.com/foo/bar/fun.html HTTP/1.1
Host: www.origin.com
```

---

**Note:** In this case, the client requests the entire URL instead of just the path as is the case when using a *reverse proxy* or when requesting content directly from the *origin server*.

---

2. The proxy verifies whether the response for `http://www-origin-cache.cdn.com/foo/bar/fun.html` is already in the cache. If it is not in the cache:
  1. The proxy sends the HTTP request to the *Origin*.

1091: The *Forward Proxy* Requests Content from the *origin server*

```
GET /foo/bar/fun.html HTTP/1.1
Host: www.origin.com
```

2. The *origin server* responds with the requested content.

1092: The *origin server*'s Response

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Server: Apache/2.2.15 (Red Hat)
Last-Modified: Sun, 14 Dec 2014
↳ 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE html><html><body>This is a
↳ fun file</body></html>
```

3. The proxy sends this on to the client, optionally adding a *Via:* header to indicate that the request was serviced by proxy.

1093: The *Forward Proxy*'s Response to the Client

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Last-Modified: Sun, 14 Dec 2014
↳ 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
```

(continues on next page)

(continued from previous page)

```
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8
Age: 0
Via: http/1.1 cache01.cdn.kabletown.
    ↪net (ApacheTrafficServer/4.2.1
    ↪[uScSsSfUpSeN:t cCSi p sS])
Server: ATS/4.2.1

<!DOCTYPE html><html><body>This is a
    ↪fun file</body></html>
```

If, however, the requested content *was* in the cache the proxy responds to the client with the previously retrieved result

#### 1094: The *Forward Proxy* Sends the Cached Response

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Last-Modified: Sun, 14 Dec 2014 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8
Age: 99711
Via: http/1.1 cache01.cdn.kabletown.net
    ↪(ApacheTrafficServer/4.2.1 [uScSsSfUpSeN:t cCSi p
    ↪sS])
Server: ATS/4.2.1

<!DOCTYPE html><html><body>This is a fun file</body>
    ↪</html>
```

**geo localization or geo routing** Localizing clients to the nearest caches using a geo database like the one from Maxmind.

**Health Protocol** The protocol to monitor the health of all the caches. See [Health Protocol](#).

**Inner-tier**

**Inner-tier cache**

**Inner-tier caches**

**Inner-tier cache server**

**Inner-tier cache servers** The tier between the First tier and the Last tier. The inner tier in a [Topology](#) is the tier that forwards requests from other caches to other caches, i.e. caches in this tier do not directly serve the end-user and do not make requests to [Origins](#).

**Last-tier**

**Last-tier cache**

**Last-tier caches**

**Last-tier cache server**

**Last-tier cache servers** The tier above the First and Inner tiers. The last tier in a *Topology* is the tier that forwards requests from other caches to *Origins*.

**localization** Finding location on the network, or on planet earth

**Mid**

**Mid-tier**

**Mid-Tier**

**Mid-tier cache**

**Mid-tier caches**

**Mid-tier cache server**

**Mid-tier cache servers** The tier above the edge tier. The mid tier does not directly serves the end-user and is used as an additional layer between the edge and the *Origin*. In a Traffic Control CDN the basic function of the mid cache is that of a *forward proxy*.

**Origin**

**Origins**

**origin server**

**origin servers**

**Origin Servers** The source of content for the CDN. Usually a redundant HTTP/1.1 webserver.

**ORT** The previous Traffic Control cache config app, replaced by *t3c*.

**Parameter**

**Parameters** Typically refers to a line in a configuration file, but in practice can represent any arbitrary configuration option.

**See also:**

The *Profiles and Parameters* overview section.

**parent**

**parents** The *parent(s)* of a *cache server* is/are the *cache server(s)* belonging to either the “parent” or “secondary parent” *Cache Group(s)* of the *Cache Group* to which the *cache server* belongs. For example, in general it is true that an *Edge-tier cache server* has one or more *parents* which are *Mid-tier cache servers*.

**Physical Location**

**Physical Locations** A pair of geographic coordinates (latitude and longitude) that is used by *Cache Groups* to define their location. This information is used by Traffic Router to route client traffic to the geographically nearest *Cache Group*.

## Profile

**Profiles** A *Profile* is, most generally, a group of *Parameters* that will be applied to a server. *Profiles* are typically re-used by all *Edge-tier cache servers* within a CDN or *Cache Group*. A *Profile* will, in addition to configuration *Parameters*, define the CDN to which a server belongs and the “*Type*” of the Profile - which determines some behaviors of Traffic Control components. The allowed “*Types*” of *Profiles* are **not** the same as *Types*, and are maintained as a PostgreSQL “Enum” in `traffic_ops/app/db/create_tables.sql`.

---

**Tip:** A *Profile* of the wrong type assigned to a Traffic Control component *will* (in general) cause it to function incorrectly, regardless of the *Parameters* assigned to it.

---

### See also:

The *Profiles and Parameters* overview section.

## Queue

### Queue Updates

**Queue Server Updates** *Queueing Updates* is an action that signals to various ATC components - most notably *cache servers* - that any configuration changes that are pending are to be applied now. Specifically, Traffic Monitor and Traffic Router are updated through a CDN *Snapshot*, and *not Queued Updates*. In particular, *ORT* will notice that the server on which it’s running has new configuration, and will request the new configuration from Traffic Ops.

Updates may be queued on a server-by-server basis (in Traffic Portal’s *Servers* view), a Cache Group-wide basis (in Traffic Portal’s *Cache Groups* view), or on a CDN-wide basis (in Traffic Portal’s *CDNs* view). Usually using the CDN-wide version is easiest, and unless there are special circumstances, and/or the user really knows what he or she is doing, it is recommended that the full CDN-wide *Queue Updates* be used.

This is similar to taking a CDN *Snapshot*, but this configuration change affects only servers, and not routing.

That seems like a vague difference because it is - in general the rule to follow is that changes to *Profiles* and *Parameters* requires only updates be queued, changes to the assignments of *cache servers* to *Delivery Services* requires both a *Snapshot* and a *Queue Updates*, and changes to only a *Delivery Service* itself (usually) entails a *Snapshot* only. These aren’t exhaustive rules, and a grasp of what changes require which action(s) will take time to form. In general, when doing both *Queueing Updates* as well as taking a CDN *Snapshot*, it is advisable to first *Queue Updates* and *then* take the *Snapshot*, as otherwise Traffic Router may route clients to *Edge-tier cache servers* that are not equipped to service their request(s). However, when modifying the assignment(s) of *cache servers* to one or more *Delivery Services*, a *Snapshot* ought to be taken before updates are queued.

**Warning:** Updates to *Parameters* with certain *Config File* values may require running *ORT* in a different mode, occasionally manually. Though the server may appear to no

longer have pending updates in these cases, until this manual intervention is performed the configuration *will not be correct*.

### Region

**Regions** A group of *Physical Locations*.

### reverse proxy

**reverse proxies** A *reverse proxy* acts on behalf of the *origin server* such that the client is (potentially) unaware it is not communicating directly with the *Origin*. All Edge-tier *cache servers* in a Traffic Control CDN are *reverse proxies*. To the end user a Traffic Control-based CDN appears as a *reverse proxy* since it retrieves content from the *origin server*, acting on behalf of that *origin server*. The client requests a URL that has a hostname which resolves to the *reverse proxy*'s IP address and, in compliance with the HTTP 1.1 specification (**RFC 2616**), the client sends a `Host:` header to the *reverse proxy* that matches the hostname in the URL. The proxy looks up this hostname in a list of mappings to find the *Origin* hostname; if the hostname of the `Host:` header is not found in the list, the proxy will send an error (usually either `404 Not Found` or `503 Service Unavailable` as appropriate) to the client. If the supplied hostname is found in this list of mappings, the proxy checks its cache, and when the content is not already present, connects to the *Origin* to which the requested `Host:` maps requests the path of the original URL, providing the *Origin* hostname in the `Host` header. The proxy then stores the URL in its cache and serves the contents to the client. When there are subsequent requests for the same URL, a caching proxy serves the content out of its cache - provided *Cache Control Headers and Revalidation* are satisfied - thereby reducing latency and network traffic.

#### See also:

The Apache Traffic Server documentation on reverse proxy.

To insert a *reverse proxy* into a typical HTTP 1.1 request and response flow, the *reverse proxy* needs to be told where the *origin server* can be reached (and which *Origin* to use for a given request when it's configured to proxy requests for multiple *Origins*). In ATS this is handled by adding rules to the [remap.config configuration file](#). The content owner must inform the clients, by updating the URL, to receive the content from the cache and not from the *origin server* directly. For example, clients might be instructed to request content from `http://www-origin-cache.cdn.com` which points to a *reverse proxy* for the actual *Origin* located at `http://www.origin.com`.

Now, if the client requests `/foo/bar/fun.html` from the *reverse proxy* the sequence of events is as follows. is given the URL `http://www-origin-cache.cdn.com/foo/bar/fun.html` (note the different hostname) and when attempting to obtain that URL, the following occurs:

1. The client sends a DNS request to the LDNS to resolve the name `www-origin-cache.cdn.com` to an IP address.
2. The LDNS finds an IP address for `www-origin-cache.cdn.com` e.g. `55.44.33.22`.
3. The client sends an HTTP request for `/foo/bar/fun.html` to the IP address.

1095: Client Requests Content from the *Reverse Proxy*

```
GET /foo/bar/fun.html HTTP/1.1
Host: www-origin-cache.cdn.com
```

4. The *reverse proxy* finds out the URL of the true *Origin* - in the case of ATS this is done by looking up `www-origin-cache.cdn.com` in its remap rules - and finds that it is `www.origin.com`.
5. The proxy checks its cache to see if the response for `GET /foo/bar/fun.html HTTP/1.1` from `www.origin.com` is already in the cache.
6. If the response is not in the cache:
  1. The proxy sends the request to the actual *Origin*

1096: *Reverse Proxy* Requests Content from the *origin server*

```
GET /foo/bar/fun.html HTTP/1.1
Host: www.origin.com
```

2. The *origin server* responds with the requested content

1097: Response from the *origin server*

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Server: Apache/2.2.15 (Red Hat)
Last-Modified: Sun, 14 Dec 2014
    ↪ 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE html><html><body>This is a
    ↪ fun file</body></html>
```

3. The proxy sends the response on to the client, optionally adding a `Via:` header to indicate that the request was serviced by proxy.

1098: Resulting Response from the *Reverse Proxy* to the Client

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Last-Modified: Sun, 14 Dec 2014
    ↪ 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
```

(continues on next page)



(continued from previous page)

```
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8
Age: 0
Via: http/1.1 cache01.cdn.kabletown.
    ↪net (ApacheTrafficServer/4.2.1
    ↪[uScSsSfUpSeN:t cCSi p sS])
Server: ATS/4.2.1

<!DOCTYPE html><html><body>This is a
    ↪fun file</body></html>
```

If, however, the response *was* already in the cache - and still valid according to the *Cache Control Headers and Revalidation* - the proxy responds to the client with the previously retrieved result.

#### 1099: The *Reverse Proxy* Provides a Cached Response

```
HTTP/1.1 200 OK
Date: Sun, 14 Dec 2014 23:22:44 GMT
Last-Modified: Sun, 14 Dec 2014 23:18:51 GMT
ETag: "1aa008f-2d-50a3559482cc0"
Content-Length: 45
Connection: close
Content-Type: text/html; charset=UTF-8
Age: 39711
Via: http/1.1 cache01.cdn.kabletown.net
    ↪(ApacheTrafficServer/4.2.1 [uScSsSfUpSeN:t cCSi p
    ↪sS])
Server: ATS/4.2.1

<!DOCTYPE html><html><body>This is a fun file</body>
    ↪</html>
```

## Role

**Roles** Permissions *Roles* define the operations a user is allowed to perform, and are currently an ordered list of permission levels.

### See also:

For a more complete description of Roles, see the *Roles* overview section.

## Server Capability

**Server Capabilities** A *Server Capability* (not to be confused with a “Capability”) expresses the capacity of a *cache server* to serve a particular kind of traffic. For example, a *Server Capability* could be created named “RAM” to be assigned to *cache servers* that have RAM-disks allocated for content caching. *Server Capabilities* can also be required by *Delivery Services*, which will prevent *cache servers* without that *Server Capability* from



being assigned to them. It also prevents *Mid-tier cache servers* without said *Server Capability* from being selected to serve upstream requests from those *Edge-tier cache servers* assigned to the requiring *Delivery Services*.

## **Service Category**

**Service Categories** A *Service Category* defines the type of content being delivered by a *Delivery Service*. For example, a *Service Category* could be created named “linear” and assigned to a *Delivery Service* that delivers linear content.

## **Snapshot**

### **Snapshots**

### **CDN Snapshot**

**CDN Snapshots** Previously called a “CRConfig” or “CRConfig.json” (and still called such in many places), this is a rather large set of routing information generated from a CDN’s configuration and topology.

## **Status**

**Statuses** A *Status* represents the current operating state of a server. The default *Statuses* made available on initial startup of Traffic Ops are related to the *Health Protocol* and are explained in that section.

**t3c** The Traffic Control cache config app, used to generate and apply cache configuration files.

### **See also:**

For usage and testing documentation, refer to *t3c*.

## **Tenant**

### **Tenants**

### **Tenancy**

**Tenancies** Users are grouped into *Tenants* (or *Tenancies*) to segregate ownership of and permissions over *Delivery Services* and their resources. To be clear, the notion of *Tenancy* **only** applies within the context of *Delivery Services* and does **not** apply permissions restrictions to any other aspect of Traffic Control.

## **Topology**

**Topologies** A structure composed of *Cache Groups* and parent relationships, which is assignable to one or more *Delivery Services*.

### **Topology Node**

### **Topology Nodes**

### **Parent Topology Node**

### **Parent Topology Nodes**

### **Child Topology Node**

**Child Topology Nodes** Each *Topology Node* is associated with a particular *Cache Group*. In addition, the Topology Node has 0, 1, or 2 Parent Topology Nodes and has 0, 1, or 2 Child Topology Nodes, according to your configuration.

### Type

**Types** A *Type* defines a type of some kind of object configured in Traffic Ops. Unfortunately, that is exactly as specific as this definition can be.

## PYTHON MODULE INDEX

### t

to\_access, [1604](#)

trafficops, [1579](#)

trafficops.\_\_version\_\_, [1604](#)

trafficops.utils, [1603](#)



## Symbols

- 7
  - command line option, [286](#)
- 8
  - command line option, [286](#)
- L
  - command line option, [286](#)
- P TO\_PASSWORD
  - command line option, [1577](#)
- S
  - command line option, [287](#)
- U TO\_USER
  - command line option, [1578](#)
- api-routes
  - traffic\_ops command line option, [100](#)
- api-version API\_VERSION
  - toaccess command line option, [1605](#)
- backendcfg BACKEND\_CONFIG\_PATH
  - traffic\_ops command line option, [101](#)
- cache-host-name NAME
  - command line option, [1577](#)
- cfg CONFIG
  - traffic\_vault\_migrate command line option, [349](#)
- cfg CONFIG\_FILE
  - reencrypt command line option, [225](#)
- cfg CONFIG\_PATH
  - traffic\_ops command line option, [100](#)
- compare
  - traffic\_vault\_migrate command line option, [345](#)
- database database\_name
  - sync\_ts\_databases command line option, [217](#)
- days N
  - sync\_ts\_databases command line option, [217](#)
- dbcfg DB\_CONFIG\_PATH
  - traffic\_ops command line option, [100](#)
- dir TSR00T
  - command line option, [1577](#)
- dir directory
  - command line option, [136](#)
- dry
  - traffic\_vault\_migrate command line option, [346](#)
- dry\_run
  - traffic\_vault\_util command line option, [1608](#)
- dump
  - traffic\_vault\_migrate command line option, [345](#)
- env ENVIRONMENT
  - admin command line option, [342](#)
- fill DIR
  - traffic\_vault\_migrate command line option, [346](#)
- fixtures FIXTURES
  - traffic\_vault\_migrate command line option, [349](#)
- fromCfgPath=CFG
  - traffic\_vault\_migrate command line option, [346](#)
- fromType=TYPE
  - traffic\_vault\_migrate command line option, [346](#)
- full
  - toaccess command line option, [1605](#)
- get-data DATA

```

    command line option, 1576
--help
    atc command line option, 385, 387
    command line option, 1577
    create_ts_databases command
        line option, 216
    reencrypt command line option,
        225
    sync_ts_databases command line
        option, 217
    toaccess command line option,
        1604
    traffic_vault_migrate command
        line option, 346
--http port
    command line option, 136
--includeSystemTests {no|yes}
    traffic_vault_migrate command
        line option, 349
--insecure
    toaccess command line option,
        1605
    traffic_vault_util command line
        option, 1609
--list-plugins
    command line option, 1577
--logCfg CFG
    traffic_vault_migrate command
        line option, 346
--logLevel=LEVEL
    traffic_vault_migrate command
        line option, 345
--log-location-error ERROR_LOCATION
    command line option, 1576
--log-location-info INFO_LOCATION
    command line option, 1577
--log-location-warning
    WARNING_LOCATION
    command line option, 1578
--new-key NEW_KEY
    reencrypt command line option,
        225
--noConfirm
    traffic_vault_migrate command
        line option, 346
--num-retries N
    command line option, 1577
--password password
    create_ts_databases command
        line option, 216
--plugins
    traffic_ops command line option,
        100
--pretty
    toaccess command line option,
        1605
--previous-key PREVIOUS_KEY
    reencrypt command line option,
        225
--raw-path
    toaccess command line option,
        1605
--replication N
    create_ts_databases command
        line option, 216
--request-headers
    toaccess command line option,
        1605
--request-payload
    toaccess command line option,
        1605
--response-headers
    toaccess command line option,
        1605
--revalidate-only
    command line option, 1578
--riakcfg RIAK_CONFIG_PATH
    traffic_ops command line option,
        100
--set-queue-status QUEUE_STATUS
    command line option, 1577
--set-reval-status REVAL_STATUS
    command line option, 1576
--source-password password
    sync_ts_databases command line
        option, 217
--source-url URL
    sync_ts_databases command line
        option, 217
--source-user username
    sync_ts_databases command line
        option, 217
--started filename
    command line option, 136
--target-password password
    sync_ts_databases command line

```

```

    option, 217
--target-url URL
    sync_ts_databases command line
    option, 217
--target-user username
    sync_ts_databases command line
    option, 217
--toCfgPath=CFG
    traffic_vault_migrate command
    line option, 346
--toType=TYPE
    traffic_vault_migrate command
    line option, 346
--to-password PASSWORD
    toaccess command line option,
    1605
--to-url URL
    toaccess command line option,
    1605
--to-user USERNAME
    toaccess command line option,
    1605
--traffic-ops-disable-proxy
    command line option, 1577
--traffic-ops-insecure
    command line option, 1577
--traffic-ops-password TO_PASSWORD
    command line option, 1577
--traffic-ops-timeout-milliseconds
    TIMEOUT
    command line option, 1578
--traffic-ops-url TO_URL
    command line option, 1578
--traffic-ops-user TO_USER
    command line option, 1578
--trafficvault
    admin command line option, 342
--url URL
    create_ts_databases command
    line option, 216
--user username
    create_ts_databases command
    line option, 216
--vault_action ACTION
    traffic_vault_util command line
    option, 1608
--vault_ip IP
    traffic_vault_util command line
    option, 1608
--vault_password PASSWD
    traffic_vault_util command line
    option, 1608
--vault_port PORT
    traffic_vault_util command line
    option, 1609
--vault_user USER
    traffic_vault_util command line
    option, 1609
--version
    command line option, 1578
    toaccess command line option,
    1605
    traffic_ops command line option,
    101
--wait
    atc command line option, 387
-a
    command line option, 286
-a API_VERSION
    toaccess command line option,
    1605
-a REVAL_STATUS
    command line option, 1576
-b
    command line option, 286
-c
    traffic_vault_migrate command
    line option, 345
-d
    command line option, 286
    traffic_vault_migrate command
    line option, 345
-d DATA
    command line option, 1576
-e ERROR_LOCATION
    command line option, 1576
-e LEVEL
    traffic_vault_migrate command
    line option, 345
-f
    toaccess command line option,
    1605
-f CFG
    traffic_vault_migrate command
    line option, 346
-f FILE

```

command line option, [286](#)

-g CFG  
traffic\_vault\_migrate command  
line option, [346](#)

-h  
atc command line option, [385](#), [387](#)  
command line option, [286](#), [1577](#)  
toaccess command line option,  
[1604](#)  
traffic\_vault\_migrate command  
line option, [346](#)

-i DIR  
traffic\_vault\_migrate command  
line option, [346](#)

-i INFO\_LOCATION  
command line option, [1577](#)

-k  
toaccess command line option,  
[1605](#)

-l  
command line option, [286](#), [1577](#)

-l CFG  
traffic\_vault\_migrate command  
line option, [346](#)

-m  
traffic\_vault\_migrate command  
line option, [346](#)

-n NAME  
command line option, [1577](#)

-o  
command line option, [286](#)

-o TYPE  
traffic\_vault\_migrate command  
line option, [346](#)

-p  
command line option, [286](#), [1577](#)  
toaccess command line option,  
[1605](#)

-q  
command line option, [286](#)

-q QUEUE\_STATUS  
command line option, [1577](#)

-r  
toaccess command line option,  
[1605](#)  
traffic\_vault\_migrate command  
line option, [346](#)

-r N

command line option, [1577](#)

-s  
command line option, [287](#), [1577](#)

-t TIMEOUT  
command line option, [1578](#)

-t TYPE  
traffic\_vault\_migrate command  
line option, [346](#)

-u TO\_URL  
command line option, [1578](#)

-v  
command line option, [287](#), [1578](#)  
toaccess command line option,  
[1605](#)

-w  
atc command line option, [387](#)

-w WARNING\_LOCATION  
command line option, [1578](#)

-y  
command line option, [1578](#)

302 content routing, [1613](#)

## A

ACME Account, [1613](#)

add\_ssl\_keys\_to\_deliveryservice()  
(*trafficops.TOSession* method), [1584](#)

admin command line option  
--env ENVIRONMENT, [342](#)  
--trafficvault, [342](#)  
command, [343](#)

api\_base\_url (*trafficops.RestApiSession*  
property), [1579](#)

api\_request() (in module *trafficops*), [1603](#)

api\_version (*trafficops.RestApiSession*  
property), [1579](#)

assign\_delivery\_services\_to\_federations()  
(*trafficops.TOSession* method), [1584](#)

assign\_deliveryservice\_servers\_by\_ids()  
(*trafficops.TOSession* method), [1584](#)

assign\_deliveryservice\_servers\_by\_names()  
(*trafficops.TOSession* method), [1584](#)

assign\_federation\_resolver\_to\_federations()  
(*trafficops.TOSession* method), [1584](#)

assign\_parameter\_to\_profile\_ids()  
(*trafficops.TOSession* method), [1585](#)

assign\_profile\_to\_parameter\_ids()  
(*trafficops.TOSession* method), [1585](#)



- `associate_paramater_to_profile()`  
(*trafficops.TOSession method*), 1585
- `associate_parameters_by_profile_id()`  
(*trafficops.TOSession method*), 1585
- `associate_parameters_by_profile_name()`  
(*trafficops.TOSession method*), 1585
- `astats (stats_over_http)`, 1613
- `atc` command line option
  - `--help`, 385, 387
  - `--wait`, 387
  - `-h`, 385, 387
  - `-w`, 387
  - CMD, 386
  - SERVICE, 386, 388, 389
- B**
- `base_url` (*trafficops.TOSession property*), 1585
- C**
- Cache Group, 1613
- Cache Groups, 1613
- Cache Server, 1614
- `cache server`, 1614
- `cache servers`, 1614
- `cachegroups_queue_update()` (*trafficops.TOSession method*), 1585
- CDN Snapshot, 1625
- CDN Snapshots, 1625
- `cdns_queue_update()` (*trafficops.TOSession method*), 1585
- Child Topology Node, 1625
- Child Topology Nodes, 1626
- `close()` (*trafficops.RestApiSession method*), 1579
- CMD
  - `atc` command line option, 386
- command
  - admin command line option, 343
- command line option
  - `-7`, 286
  - `-8`, 286
  - `-L`, 286
  - `-P TO_PASSWORD`, 1577
  - `-S`, 287
  - `-U TO_USER`, 1578
  - `--cache-host-name NAME`, 1577
  - `--dir TSROOT`, 1577
  - `--dir directory`, 136
  - `--get-data DATA`, 1576
  - `--help`, 1577
  - `--http port`, 136
  - `--list-plugins`, 1577
  - `--log-location-error ERROR_LOCATION`, 1576
  - `--log-location-info INFO_LOCATION`, 1577
  - `--log-location-warning WARNING_LOCATION`, 1578
  - `--num-retries N`, 1577
  - `--revalidate-only`, 1578
  - `--set-queue-status QUEUE_STATUS`, 1577
  - `--set-reval-status REVAL_STATUS`, 1576
  - `--started filename`, 136
  - `--traffic-ops-disable-proxy`, 1577
  - `--traffic-ops-insecure`, 1577
  - `--traffic-ops-password TO_PASSWORD`, 1577
  - `--traffic-ops-timeout-milliseconds TIMEOUT`, 1578
  - `--traffic-ops-url TO_URL`, 1578
  - `--traffic-ops-user TO_USER`, 1578
  - `--version`, 1578
  - `-a`, 286
  - `-a REVAL_STATUS`, 1576
  - `-b`, 286
  - `-d`, 286
  - `-d DATA`, 1576
  - `-e ERROR_LOCATION`, 1576
  - `-f FILE`, 286
  - `-h`, 286, 1577
  - `-i INFO_LOCATION`, 1577
  - `-l`, 286, 1577
  - `-n NAME`, 1577
  - `-o`, 286
  - `-p`, 286, 1577
  - `-q`, 286
  - `-q QUEUE_STATUS`, 1577
  - `-r N`, 1577
  - `-s`, 287, 1577
  - `-t TIMEOUT`, 1578
  - `-u TO_URL`, 1578
  - `-v`, 287, 1578
  - `-w WARNING_LOCATION`, 1578

-y, 1578  
 Content Invalidation Job, 1614  
 Content Invalidation Jobs, 1614  
 content routing, 1614  
 copy\_profile() (trafficops.TOSession method), 1586  
 Coverage Zone File, 1614  
 Coverage Zone Map, 1614  
 create() (trafficops.RestApiSession method), 1579  
 create\_asn() (trafficops.TOSession method), 1586  
 create\_cachegroups() (trafficops.TOSession method), 1586  
 create\_cdn() (trafficops.TOSession method), 1586  
 create\_cdn\_dns\_sec\_keys() (trafficops.TOSession method), 1586  
 create\_coordinates() (trafficops.TOSession method), 1586  
 create\_deliveryservice() (trafficops.TOSession method), 1586  
 create\_deliveryservice\_regexes() (trafficops.TOSession method), 1586  
 create\_division() (trafficops.TOSession method), 1586  
 create\_federation() (trafficops.TOSession method), 1587  
 create\_federation\_in\_cdn() (trafficops.TOSession method), 1587  
 create\_federation\_resolver() (trafficops.TOSession method), 1587  
 create\_federation\_user() (trafficops.TOSession method), 1587  
 create\_job() (trafficops.TOSession method), 1587  
 create\_origins() (trafficops.TOSession method), 1587  
 create\_parameter() (trafficops.TOSession method), 1587  
 create\_profile() (trafficops.TOSession method), 1587  
 create\_region() (trafficops.TOSession method), 1588  
 create\_role() (trafficops.TOSession method), 1588  
 create\_server() (trafficops.TOSession method), 1588

create\_servercheck() (trafficops.TOSession method), 1588  
 create\_staticdnsentries() (trafficops.TOSession method), 1588  
 create\_tenant() (trafficops.TOSession method), 1588  
 create\_to\_extension() (trafficops.TOSession method), 1588  
 create\_topology() (trafficops.TOSession method), 1588  
 create\_ts\_databases command line option  
   --help, 216  
   --password password, 216  
   --replication N, 216  
   --url URL, 216  
   --user username, 216  
 create\_user() (trafficops.TOSession method), 1588  
 create\_user\_with\_registration() (trafficops.TOSession method), 1589

## D

### DATA

toaccess command line option, 1604

Deep Coverage Zone File, 1615  
 Deep Coverage Zone Map, 1615  
 delete() (in module to\_access), 1606  
 delete() (trafficops.RestApiSession method), 1579  
 delete\_asn() (trafficops.TOSession method), 1589  
 delete\_cache\_group\_parameters() (trafficops.TOSession method), 1589  
 delete\_cachegroups() (trafficops.TOSession method), 1589  
 delete\_cdn\_by\_id() (trafficops.TOSession method), 1589  
 delete\_cdn\_dns\_sec\_keys() (trafficops.TOSession method), 1589  
 delete\_coordinates() (trafficops.TOSession method), 1589  
 delete\_deliveryservice\_by\_id() (trafficops.TOSession method), 1589  
 delete\_deliveryservice\_regex\_by\_regex\_id() (trafficops.TOSession method), 1589  
 delete\_deliveryservice\_servers\_by\_id()

- (trafficops.TOSession method)*, 1590
- delete\_deliveryservice\_ssl\_keys\_by\_xml() *(trafficops.TOSession method)*, 1590
- delete\_division() *(trafficops.TOSession method)*, 1590
- delete\_federation\_in\_cdn() *(trafficops.TOSession method)*, 1590
- delete\_federation\_resolver() *(trafficops.TOSession method)*, 1590
- delete\_federation\_user() *(trafficops.TOSession method)*, 1590
- delete\_job() *(trafficops.TOSession method)*, 1590
- delete\_origins() *(trafficops.TOSession method)*, 1591
- delete\_parameter() *(trafficops.TOSession method)*, 1591
- delete\_physical\_location() *(trafficops.TOSession method)*, 1591
- delete\_profile\_by\_id() *(trafficops.TOSession method)*, 1591
- delete\_profile\_parameter\_association\_by\_id() *(trafficops.TOSession method)*, 1591
- delete\_region() *(trafficops.TOSession method)*, 1591
- delete\_role() *(trafficops.TOSession method)*, 1591
- delete\_server\_by\_id() *(trafficops.TOSession method)*, 1591
- delete\_staticdnsentries() *(trafficops.TOSession method)*, 1591
- delete\_to\_extension() *(trafficops.TOSession method)*, 1592
- delete\_topology() *(trafficops.TOSession method)*, 1592
- Delivery Service, 1615
- Delivery Service Request, 1616
- Delivery Service Requests, 1616
- Delivery Service required capabilities, 1616
- Delivery Services, 1615
- Division, 1616
- Divisions, 1616
- DSR, 1616
- DSRs, 1616
- E**
- Edge, 1616
- Edge-Tier, 1616
- Edge-tier, 1616
- Edge-tier cache, 1616
- Edge-tier cache server, 1616
- Edge-tier cache servers, 1617
- Edge-tier caches, 1616
- environment variable
  - MOJO\_MODE, 103, 283, 342
  - SESSION\_TIMEOUT\_IN\_SECS, 349, 351
  - TO\_PASSWORD, 283, 284, 385, 1577, 1578, 1605
  - TO\_URL, 284, 349, 352, 385, 1578, 1605, 1606
  - TO\_USER, 283, 284, 385, 1578, 1605, 1606
  - TO\_USER\_ADMIN, 350, 352
  - TO\_USER\_DISALLOWED, 350, 352
  - TO\_USER\_EXTENSION, 350, 352
  - TO\_USER\_FEDERATION, 350, 353
  - TO\_USER\_OPERATIONS, 350, 353
  - TO\_USER\_PASSWORD, 350, 352
  - TO\_USER\_PORTAL, 350, 353
  - TO\_USER\_READ\_ONLY, 350, 353
- TODB\_DESCRIPTION, 349, 353
- TODB\_HOSTNAME, 349, 354
- TODB\_NAME, 349, 353
- TODB\_PASSWORD, 349, 354
- TODB\_PORT, 349, 354
- TODB\_SSL, 350, 354
- TODB\_TYPE, 350, 354
- TODB\_USER, 350, 354
- F**
- Federation, 1617
- Federations, 1617
- First-tier, 1617
- First-tier cache, 1617
- First-tier cache server, 1617
- First-tier cache servers, 1617
- First-tier caches, 1617
- forward proxies, 1617
- forward proxy, 1617
- G**
- generate\_deliveryservice\_ssl\_keys() *(trafficops.TOSession method)*, 1592
- generate\_deliveryservice\_url\_signature\_keys() *(trafficops.TOSession method)*, 1592
- generate\_iso() *(trafficops.TOSession method)*, 1592

geo localization or geo routing, [1619](#)

`get()` (in module `to_access`), [1606](#)

`get()` (*trafficops.RestApiSession* method), [1580](#)

`get_all_cachegroup_parameters()` (*trafficops.TOSession* method), [1592](#)

`get_all_deliveryservice_servers()` (*trafficops.TOSession* method), [1592](#)

`get_api_capabilities()` (*trafficops.TOSession* method), [1592](#)

`get_asns()` (*trafficops.TOSession* method), [1592](#)

`get_authenticated_user()` (*trafficops.TOSession* method), [1593](#)

`get_cache_stats()` (*trafficops.TOSession* method), [1593](#)

`get_cachegroup_parameters()` (*trafficops.TOSession* method), [1593](#)

`get_cachegroups()` (*trafficops.TOSession* method), [1593](#)

`get_capabilities()` (*trafficops.TOSession* method), [1593](#)

`get_cdn_dns_sec_keys()` (*trafficops.TOSession* method), [1593](#)

`get_cdn_health_by_name()` (*trafficops.TOSession* method), [1593](#)

`get_cdn_monitoring_info()` (*trafficops.TOSession* method), [1593](#)

`get_cdn_ssl_keys()` (*trafficops.TOSession* method), [1593](#)

`get_cdns()` (*trafficops.TOSession* method), [1593](#)

`get_cdns_capacity()` (*trafficops.TOSession* method), [1594](#)

`get_cdns_domains()` (*trafficops.TOSession* method), [1594](#)

`get_cdns_health()` (*trafficops.TOSession* method), [1594](#)

`get_cdns_routing()` (*trafficops.TOSession* method), [1594](#)

`get_change_logs()` (*trafficops.TOSession* method), [1594](#)

`get_change_logs_newcount()` (*trafficops.TOSession* method), [1594](#)

`get_coordinates()` (*trafficops.TOSession* method), [1594](#)

`get_current_snapshot_crconfig()` (*trafficops.TOSession* method), [1594](#)

`get_delivery_service_capacity()` (*trafficops.TOSession* method), [1594](#)

`get_delivery_service_health()` (*trafficops.TOSession* method), [1594](#)

`get_delivery_service_stats()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryservice_ineligible_servers()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryservice_regexes_by_id()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryservice_servers()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryservice_ssl_keys_by_xml_id()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryservices()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryservices_regexes()` (*trafficops.TOSession* method), [1595](#)

`get_deliveryserviceserver()` (*trafficops.TOSession* method), [1595](#)

`get_divisions()` (*trafficops.TOSession* method), [1596](#)

`get_federation_delivery_services()` (*trafficops.TOSession* method), [1596](#)

`get_federation_resolvers()` (*trafficops.TOSession* method), [1596](#)

`get_federation_resolvers_by_id()` (*trafficops.TOSession* method), [1596](#)

`get_federation_users()` (*trafficops.TOSession* method), [1596](#)

`get_federations()` (*trafficops.TOSession* method), [1596](#)

`get_federations_for_cdn()` (*trafficops.TOSession* method), [1596](#)

`get_jobs()` (*trafficops.TOSession* method), [1596](#)

`get_origins()` (*trafficops.TOSession* method), [1596](#)

`get_osversions()` (*trafficops.TOSession* method), [1597](#)

`get_parameters()` (*trafficops.TOSession* method), [1597](#)

`get_parameters_by_profile_id()` (*trafficops.TOSession* method), [1597](#)

`get_parameters_by_profile_name()` (*trafficops.TOSession* method), [1597](#)

`get_pending_snapshot_crconfig()`

(*trafficops.TOSession* method), 1597  
[get\\_physical\\_locations\(\)](#) (*trafficops.TOSession* method), 1597  
[get\\_plugins\(\)](#) (*trafficops.TOSession* method), 1597  
[get\\_profiles\(\)](#) (*trafficops.TOSession* method), 1597  
[get\\_regions\(\)](#) (*trafficops.TOSession* method), 1597  
[get\\_roles\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_server\\_delivery\\_services\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_server\\_details\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_server\\_update\\_status\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_servercheck\\_extensions\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_servers\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_staticdnsentries\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_statuses\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_system\\_info\(\)](#) (*trafficops.TOSession* method), 1598  
[get\\_tenants\(\)](#) (*trafficops.TOSession* method), 1599  
[get\\_topologies\(\)](#) (*trafficops.TOSession* method), 1599  
[get\\_traffic\\_monitor\\_cache\\_stats\(\)](#) (*trafficops.TOSession* method), 1599  
[get\\_types\(\)](#) (*trafficops.TOSession* method), 1599  
[get\\_user\\_by\\_id\(\)](#) (*trafficops.TOSession* method), 1599  
[get\\_users\(\)](#) (*trafficops.TOSession* method), 1599

## H

[head\(\)](#) (in module *to\_access*), 1606  
[head\(\)](#) (*trafficops.RestApiSession* method), 1580  
 Health Protocol, 1619

## I

Inner-tier, 1619

Inner-tier cache, 1619  
 Inner-tier cache server, 1619  
 Inner-tier cache servers, 1619  
 Inner-tier caches, 1619  
 InvalidJSONError, 1579  
[is\\_open](#) (*trafficops.RestApiSession* property), 1581

## J

job, 1614  
 jobs, 1614

## L

Last-tier, 1619  
 Last-tier cache, 1620  
 Last-tier cache server, 1620  
 Last-tier cache servers, 1620  
 Last-tier caches, 1620  
 localization, 1620  
[log\\_with\\_debug\\_info\(\)](#) (in module *trafficops.utils*), 1603  
[logged\\_in](#) (*trafficops.TOSession* property), 1599  
[login\(\)](#) (*trafficops.TOSession* method), 1599  
 LoginError, 1579

## M

Mid, 1620  
 Mid-Tier, 1620  
 Mid-tier, 1620  
 Mid-tier cache, 1620  
 Mid-tier cache server, 1620  
 Mid-tier cache servers, 1620  
 Mid-tier caches, 1620  
 module  
   [to\\_access](#), 1604  
   *trafficops*, 1579  
   *trafficops.\_\_version\_\_*, 1604  
   *trafficops.utils*, 1603  
 MOJO\_MODE, 103, 342

## O

OperationError, 1579  
[options\(\)](#) (in module *to\_access*), 1607  
[options\(\)](#) (*trafficops.RestApiSession* method), 1581  
 Origin, 1620  
 origin server, 1620  
 Origin Servers, 1620



origin servers, [1620](#)  
 Origins, [1620](#)  
 ORT, [1620](#)  
 output() (in module *to\_access*), [1607](#)

## P

Parameter, [1620](#)  
 Parameters, [1620](#)  
 parent, [1620](#)  
 Parent Topology Node, [1625](#)  
 Parent Topology Nodes, [1625](#)  
 parents, [1620](#)  
 parse\_arguments() (in module *to\_access*), [1607](#)  
 patch() (in module *to\_access*), [1607](#)  
 patch() (*trafficops.RestApiSession* method), [1581](#)  
 PATH  
     toaccess command line option, [1604](#)  
 Physical Location, [1620](#)  
 Physical Locations, [1620](#)  
 post() (in module *to\_access*), [1607](#)  
 post() (*trafficops.RestApiSession* method), [1581](#)  
 Profile, [1621](#)  
 Profiles, [1621](#)  
 put() (in module *to\_access*), [1607](#)  
 put() (*trafficops.RestApiSession* method), [1582](#)

## Q

Queue, [1621](#)  
 Queue Server Updates, [1621](#)  
 Queue Updates, [1621](#)

## R

reencrypt command line option  
     --cfg CONFIG\_FILE, [225](#)  
     --help, [225](#)  
     --new-key NEW\_KEY, [225](#)  
     --previous-key PREVIOUS\_KEY, [225](#)  
 Region, [1622](#)  
 Regions, [1622](#)  
 replace\_authenticated\_user() (*trafficops.TOSession* method), [1599](#)  
 request() (in module *to\_access*), [1607](#)  
 resp (*trafficops.InvalidJSONError* attribute), [1579](#)

resp (*trafficops.OperationError* attribute), [1579](#)

RestApiSession (class in *trafficops*), [1579](#)  
 reverse proxies, [1622](#)

reverse proxy, [1622](#)

## RFC

RFC 1035, [41](#), [492](#), [498](#), [509](#), [515](#), [1077](#), [1083](#), [1094](#), [1100](#)  
 RFC 1521, [1576](#)  
 RFC 2616, [4](#), [274](#), [1622](#)  
 RFC 2616#section-14.9, [4](#)  
 RFC 2616#section-9.1.2, [274](#)  
 RFC 3339, [18](#), [280](#), [281](#), [421–423](#), [456](#), [474](#), [546](#), [596–598](#), [677](#), [726](#), [730](#), [1005–1007](#), [1040](#), [1059](#), [1131](#), [1164–1166](#), [1247](#), [1543](#), [1547](#), [1558](#), [1562](#), [1565](#), [1569](#)  
 RFC 4033, [191](#)  
 RFC 4044, [191](#)  
 RFC 4045, [191](#)  
 RFC 4122, [960](#), [1553](#)  
 RFC 4511, [116](#)  
 RFC 6781, [191](#)  
 RFC 6781#section-4.1.1.1, [192](#)  
 RFC 7129, [191](#)  
 RFC 7231#section-4, [274](#)  
 RFC 7234, [55](#), [170](#)  
 RFC 7516, [672](#), [674](#), [675](#), [1242–1245](#)  
 RFC 7518, [672](#), [674](#), [675](#), [1242](#), [1244](#), [1245](#)  
 RFC 8006, [267](#), [1310](#), [1311](#), [1313](#), [1316](#), [1318](#)  
 RFC 8007, [267](#), [1316](#), [1318](#)  
 RFC 8008, [267](#), [1316](#), [1318](#)

Role, [1624](#)

Roles, [1624](#)

## S

Server Capabilities, [1624](#)

Server Capability, [1624](#)

server\_url (*trafficops.RestApiSession* property), [1582](#)

servers\_queue\_update() (*trafficops.TOSession* method), [1599](#)

## SERVICE

atc command line option, [386](#), [388](#), [389](#)

Service Categories, [1625](#)

- Service Category, [1625](#)
- session (*trafficops.RestApiSession* property), [1582](#)
- SESSION\_TIMEOUT\_IN\_SECS, [351](#)
- Snapshot, [1625](#)
- snapshot\_crconfig() (*trafficops.TOSession* method), [1599](#)
- Snapshots, [1625](#)
- Status, [1625](#)
- Statuses, [1625](#)
- sync\_ts\_databases command line option
  - database database\_name, [217](#)
  - days N, [217](#)
  - help, [217](#)
  - source-password password, [217](#)
  - source-url URL, [217](#)
  - source-user username, [217](#)
  - target-password password, [217](#)
  - target-url URL, [217](#)
  - target-user username, [217](#)
- T**
- t3c, [1625](#)
- Tenancies, [1625](#)
- Tenancy, [1625](#)
- Tenant, [1625](#)
- Tenants, [1625](#)
- to\_access module, [1604](#)
- TO\_PASSWORD, [283](#), [284](#), [385](#), [1577](#), [1578](#), [1605](#)
- TO\_URL, [349](#), [352](#), [385](#), [1578](#), [1605](#), [1606](#)
- to\_url (*trafficops.TOSession* property), [1600](#)
- TO\_USER, [283](#), [284](#), [385](#), [1578](#), [1605](#), [1606](#)
- TO\_USER\_ADMIN, [352](#)
- TO\_USER\_DISALLOWED, [352](#)
- TO\_USER\_EXTENSION, [352](#)
- TO\_USER\_FEDERATION, [353](#)
- TO\_USER\_OPERATIONS, [353](#)
- TO\_USER\_PASSWORD, [352](#)
- TO\_USER\_PORTAL, [353](#)
- TO\_USER\_READ\_ONLY, [353](#)
- toaccess command line option
  - api-version API\_VERSION, [1605](#)
  - full, [1605](#)
  - help, [1604](#)
  - insecure, [1605](#)
  - pretty, [1605](#)
  - raw-path, [1605](#)
  - request-headers, [1605](#)
  - request-payload, [1605](#)
  - response-headers, [1605](#)
  - to-password PASSWORD, [1605](#)
  - to-url URL, [1605](#)
  - to-user USERNAME, [1605](#)
  - version, [1605](#)
  - a API\_VERSION, [1605](#)
  - f, [1605](#)
  - h, [1604](#)
  - k, [1605](#)
  - p, [1605](#)
  - r, [1605](#)
  - v, [1605](#)
  - DATA, [1604](#)
  - PATH, [1604](#)
- TODB\_DESCRIPTION, [353](#)
- TODB\_HOSTNAME, [354](#)
- TODB\_NAME, [353](#)
- TODB\_PASSWORD, [354](#)
- TODB\_PORT, [354](#)
- TODB\_SSL, [354](#)
- TODB\_TYPE, [354](#)
- TODB\_USER, [354](#)
- Topologies, [1625](#)
- topologies\_queue\_update() (*trafficops.TOSession* method), [1600](#)
- Topology, [1625](#)
- Topology Node, [1625](#)
- Topology Nodes, [1625](#)
- TOSession (*class in trafficops*), [1582](#)
- traffic\_ops command line option
  - api-routes, [100](#)
  - backendcfg BACKEND\_CONFIG\_PATH, [101](#)
  - cfg CONFIG\_PATH, [100](#)
  - dbcfg DB\_CONFIG\_PATH, [100](#)
  - plugins, [100](#)
  - riakcfg RIAK\_CONFIG\_PATH, [100](#)
  - version, [101](#)
- traffic\_vault\_migrate command line option
  - cfg CONFIG, [349](#)
  - compare, [345](#)
  - dry, [346](#)
  - dump, [345](#)

```
--fill DIR, 346
--fixtures FIXTURES, 349
--fromCfgPath=CFG, 346
--fromType=TYPE, 346
--help, 346
--includeSystemTests {no|yes},
    349
--logCfg CFG, 346
--logLevel=LEVEL, 345
--noConfirm, 346
--toCfgPath=CFG, 346
--toType=TYPE, 346
-c, 345
-d, 345
-e LEVEL, 345
-f CFG, 346
-g CFG, 346
-h, 346
-i DIR, 346
-l CFG, 346
-m, 346
-o TYPE, 346
-r, 346
-t TYPE, 346
traffic_vault_util command line
    option
--dry_run, 1608
--insecure, 1609
--vault_action ACTION, 1608
--vault_ip IP, 1608
--vault_password PASSWD, 1608
--vault_port PORT, 1609
--vault_user USER, 1609
trafficops
    module, 1579
trafficops.__version__
    module, 1604
trafficops.utils
    module, 1603
Type, 1626
Types, 1626
```

## U

```
update_asn() (trafficops.TOSession
    method), 1600
update_cachegroups() (traffi-
    cops.TOSession method), 1600
```

```
update_cdn_by_id() (trafficops.TOSession
    method), 1600
update_coordinates() (traffi-
    cops.TOSession method), 1600
update_deliveryservice_by_id() (traf-
    ficops.TOSession method), 1600
update_deliveryservice_regexes()
    (trafficops.TOSession method), 1600
update_deliveryservice_safe() (traffi-
    cops.TOSession method), 1601
update_division() (trafficops.TOSession
    method), 1601
update_federation_in_cdn() (traffi-
    cops.TOSession method), 1601
update_job() (trafficops.TOSession
    method), 1601
update_origins() (trafficops.TOSession
    method), 1601
update_parameter() (trafficops.TOSession
    method), 1601
update_physical_location() (traffi-
    cops.TOSession method), 1601
update_profile_by_id() (traffi-
    cops.TOSession method), 1602
update_region() (trafficops.TOSession
    method), 1602
update_role() (trafficops.TOSession
    method), 1602
update_server_by_id() (traffi-
    cops.TOSession method), 1602
update_server_status_by_id() (traffi-
    cops.TOSession method), 1602
update_staticdnsentries() (traffi-
    cops.TOSession method), 1602
update_tenant() (trafficops.TOSession
    method), 1602
update_topology() (trafficops.TOSession
    method), 1602
update_user_by_id() (traffi-
    cops.TOSession method), 1603
```