9.6
IDS Sensor REST API Guide

For more documentation, see https://docs.extrahop.com.

Published: 2024-04-03

ExtraHop Networks
Seattle, WA 98101
877-333-9872 (US)
+44 (0)203 7016850 (EMEA)
+65-31585513 (APAC)
www.extrahop.com

Contents

Introduction to the ExtraHop REST API

The ExtraHop REST API enables you to automate administration and configuration tasks on your ExtraHop system. You can send requests to the ExtraHop API through a Representational State Transfer (REST) interface, which is accessed through resource URIs and standard HTTP methods.

When a REST API request is sent over HTTPS to an ExtraHop system, that request is authenticated and then authorized through an API key. After authentication, the request is submitted to the ExtraHop system and the operation completes.

Video the related training: Rest API Overview

Each ExtraHop system provides access to the built-in ExtraHop REST API Explorer, which enables you to view all of the available system resources, methods, properties, and parameters. The REST API Explorer also enables you to send API calls directly to your ExtraHop system.

Note: This guide is intended for an audience that has a basic familiarity with software development and the ExtraHop system.

ExtraHop API requirements

Before you can begin writing scripts for the ExtraHop REST API or performing operations through the REST API Explorer, you must meet the following requirements:

- Your ExtraHop system must be configured to allow API key generation for the type of user you are (remote or local).
- You must generate a valid API key.
- You must have a user account on the ExtraHop system with appropriate privileges set for the type of tasks you want to perform.

Access and authenticate to the ExtraHop REST API

Setup users and users with system and access administration privileges control whether users can generate API keys. For example, you can prevent remote users from generating keys or you can disable API key generation entirely. When this functionality is enabled, API keys are generated by users and can be viewed only by the user who generated the key.

Note: Administrators set up user accounts and assign privileges, but then users generate their own API keys. Users can delete API keys for their own account, and users with system and access administration privileges can delete API keys for any user. For more information, see Users and user groups.

After you generate an API key, you must append the key to your request headers. The following example shows a request that retrieves metadata about the firmware running on the ExtraHop system:

```
curl -i -X GET --header "Accept: application/json" \
--header "Authorization: ExtraHop apikey=2bc07e55971d4c9a88d0bb4d29ecbb29" \
"https://<hostname-or-IP-of-your-ExtraHop-system>/api/v1/extrahop"
```

Privilege levels

User privilege levels determine which ExtraHop system and administration tasks the user can perform through the ExtraHop REST API.

You can view the privilege levels for users through the `granted_roles` and `effective_roles` properties. The `granted_roles` property shows you which privilege levels are explicitly granted to the user. The `effective_roles` property shows you all privilege levels for a user, including those received outside of the granted role, such as through a user group.

The `granted_roles` and `effective_roles` properties are returned by the following operations:

- GET /users
- GET /users/{username}

The `granted_roles` and `effective_roles` properties support the following privilege levels. Note that the type of tasks for each ExtraHop system vary by the available resources listed in the REST API Explorer and depend on the modules enabled on the system and user module access privileges.

Privilege level	Actions allowed
"system": "full"	• Enable or disable API key generation for the ExtraHop system. • Generate an API key. • View the last four digits and description for any API key on the system. • Delete API keys for any user. • View and edit cross-origin resource sharing. • Perform any administration task available through the REST API. • Perform any ExtraHop system task available through the REST API.
"write": "full"	• Generate your own API key. • View or delete your own API key. • Change your own password, but you cannot perform any other administration tasks through the REST API.

Privilege level	Actions allowed
	• Perform any ExtraHop system task available through the REST API.
"write": "limited"	• Generate an API key. • View or delete their own API key. • Change your own password, but you cannot perform any other administration tasks through the REST API. • Perform all GET operations through the REST API. • Perform metric and record queries.
"write": "personal"	• Generate an API key. • View or delete your own API key. • Change your own password, but you cannot perform any other administration tasks through the REST API. • Perform all GET operations through the REST API. • Perform metric and record queries.
"metrics": "full"	• Generate an API key. • View or delete your own API key. • Change your own password, but you cannot perform any other administration tasks through the REST API. • Perform metric and record queries.
"metrics": "restricted"	• Generate an API key. • View or delete your own API key. • Change your own password, but you cannot perform any other administration tasks through the REST API.
"ndr": "full"	• View security detections • View and create investigations This is a module access privilege that can be granted to a user in addition to one of the following system access privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"
"ndr": "none"	• No access to NDR module content This is a module access privilege that can be granted to a user in addition to one of the following system access privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"

Privilege level	Actions allowed
"npm": "full"	• View performance detections • View and create dashboards • View and create alerts This is a module access privilege that can be granted to a user in addition to one of the following system access privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"
"npm": "none"	• No access to NPM module content This is a module access privilege that can be granted to a user in addition to one of the following system access privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"
"packets": "full"	• View and download packets through the `GET /packets/search` and `POST /packets/search` operations. This is an add-on privilege that can be granted to a user with one of the following privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"
"packets": "full_with_keys"	• View and download packets and session keys through the `GET /packets/search` and `POST /packets/search` operations. This is an add-on privilege that can be granted to a user with one of the following privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"
"packets": "slices_only"	• View and download the first 64 bytes of packets through the `GET /packets/search` and `POST /packets/search` operations.

Privilege level	Actions allowed
	This is an add-on privilege that can be granted to a user with one of the following privilege levels: • "write": "full" • "write": "limited" • "write": "personal" • "write": null • "metrics": "full" • "metrics": "restricted"

Manage API key access

Users with system and access administration privileges can configure whether users can generate API keys for the ExtraHop system. You can allow only local users to generate keys, or you can also disable API key generation entirely.

Users must generate an API key before they can perform operations through the ExtraHop REST API. Keys can be viewed only by the user who generated the key or system administrators with unlimited privileges. After a user generates an API key, they must append the key to their request headers.

1. Log in to the Administration settings on the ExtraHop system through `https://<extrahop-hostname-or-IP-address>/admin`.
2. In the Access Settings section, click **API Access**.
3. In the Manage API Access section, select one of the following options:
   - **Allow all users to generate an API key**: Local and remote users can generate API keys.
   - **Only local users can generate an API key**: Remote users cannot generate API keys.
   - **No users can generate an API key**: No API keys can be generated by any user.
4. Click **Save Settings**.

Generate an API key

You must generate an API key before you can perform operations through the ExtraHop REST API. Keys can be viewed only by the user who generated the key or by users with system and access administration privileges. After you generate an API key, add the key to your request headers or the ExtraHop REST API Explorer.

Before you begin
Make sure the ExtraHop system is configured to allow API key generation.

1. In the Access Settings section, click **API Access**.
2. In the Generate an API Key section, type a description for the new key, and then click **Generate**.
3. Scroll down to the API Keys section, and copy the API key that matches your description.

You can paste the key into the REST API Explorer or append the key to a request header.

Configure cross-origin resource sharing (CORS)

Cross-origin resource sharing (CORS) allows you to access the ExtraHop REST API across domain-boundaries and from specified web pages without requiring the request to travel through a proxy server.

You can configure one or more allowed origins or you can allow access to the ExtraHop REST API from any origin. Only users with system and access administration privileges can view and edit CORS settings.

1. In the **Access Settings** section, click **API Access**.
2. In the CORS Settings section, specify one of the following access configurations.
   - To add a specific URL, type an origin URL in the text box, and then click the plus (+) icon or press ENTER.

     The URL must include a scheme, such as `HTTP` or `HTTPS`, and the exact domain name. You cannot append a path; however, you can provide a port number.
   - To allow access from any URL, select the Allow API requests from any Origin checkbox.

     Note: Allowing REST API access from any origin is less secure than providing a list of explicit origins.
3. Click **Save Settings** and then click **Done**.

Set up an SSL certificate

Before making requests to an ExtraHop system with a self-signed certificate, you must set up an SSL certificate for each user who will access the ExtraHop system from a particular computer.

In each of the following examples, replace {HOST} with the hostname of your ExtraHop system.

Note: The SSL certificate applies only to the user performing the command. Each user must run the command with their login credentials to set up the SSL certificate.

Set up SSL through Windows PowerShell

```
Invoke-WebRequest "http://{HOST}/public.cer" -OutFile ($env:USERPROFILE +
"\ex.cer"); Import-Certificate ($env:USERPROFILE + "\ex.cer")
-CertStoreLocation Cert:\CurrentUser\Root
```

Set up SSL through OS X

```
curl -O http://{HOST}/public.cer; security add-trusted-cert -r trustRoot -k
~/Library/Keychains/login.keychain public.cer
```

Learn about the REST API Explorer

The REST API Explorer is a web-based tool that enables you to view detailed information about the ExtraHop REST API resources, methods, parameters, properties, and error codes. Code samples are available in Python, cURL, and Ruby for each resource. You also can perform operations directly through the tool.

Open the REST API Explorer

You can open the REST API Explorer from the Administration settings or through the following URL:

```
https://<extrahop-hostname-or-ip-address>/api/v1/explore/
```

1. Log in to the Administration settings on the ExtraHop system through `https://<extrahop-hostname-or-IP-address>/admin`.
2. From the Access Setting section, click **API Access**.
3. On the API Access page, click **REST API Explorer**.
   The REST API Explorer opens in your browser.

View operation information

From the REST API Explorer, you can click any operation to view configuration information for the resource.

The following table provides information about the sections available for resources in the REST API Explorer. Section availability varies by HTTP method. Not all methods have all of the sections listed in the table.

Section	Description
Body Parameters	Provides all of the fields for the request body and supported values for each field.
Parameters	Provides information about the available query parameters.
Responses	Provides information about the possible HTTP status codes for the resource. If you click **Send Request**, this section also includes the response from the server and the cURL, Python, and Ruby syntax required to send the specified request. **Tip:** Click **Model** to view descriptions of the fields returned in a response.

Identify objects on the ExtraHop system

Objects on the ExtraHop system can be identified by any unique value, such as the IP address, MAC address, name, or system ID. However, to perform API operations on a specific object, you must locate the object ID. You can easily locate the object ID through the following methods in the REST API Explorer.

- The object ID is provided in the headers returned from a POST request. For example, if you send a POST request to create a page, the response headers display a location URL.

The following request returned the location for the newly created tag as `/api/v1/tags/1` and the ID for the tag as `1`.

```
{
 "date": "Tue, 09 Nov 2021 18:21:00 GMT ",
 "via": "1.1 localhost",
 "server": "Apache",
 "content-type": "text/plain; charset=utf-8",
 "location": "/api/v1/tags/1",
 "cache-control": "private, max-age=0",
 "connection": "Keep-Alive",
 "keep-alive": "timeout=90, max=100",
 "content-length": "0"
}
```

- The object ID is provided for all objects returned from a GET request. For example, if you perform a GET request on all devices, the response body contains information for each device, including the ID.

  The following response body displays an entry for a single device, with an ID of 10212:

```
{
 "mod_time": 1448474346504,
 "node_id": null,
 "id": 10212,
 "extrahop_id": "test0001",
 "description": null,
 "user_mod_time": 1448474253809,
 "discover_time": 1448474250000,
 "vlanid": 0,
 "parent_id": 9352,
 "macaddr": "00:05:G3:FF:FC:28",
 "vendor": "Cisco",
 "is_l3": true,
 "ipaddr4": "10.10.10.5",
 "ipaddr6": null,
 "device_class": "node",
 "default_name": "Cisco5",
 "custom_name": null,
 "cdp_name": "",
 "dhcp_name": "",
 "netbios_name": "",
 "dns_name": "",
 "custom_type": "",
 "analysis_level": 1
},
```

- The object ID is provided in the URL for most objects. For example, in the ExtraHop system, click on **Assets**, and then **Devices**. Select any device and view the URL. In the following example, the URL for the device page shows Oid=10180.

```
https://10.10.10.205/extrahop/#/Devices?details=true&device
Oid=10180&from=6&interval_type=HR&until=0&view=l2stats
```

  To perform specific requests for that device, add 10180 to the id field in the REST API Explorer or to the body parameter in your request.

  The URL for dashboards displays a short_code, which appears after /Dashboard. When you add the short_code to the REST API Explorer or to your request, you must prepend a tilde to the short code.

In the following example, kmC9Y is the short_code. To perform requests for this dashboard, add ~kmC9Y as the value for the short_code field.

```
https://10.10.10.205/extrahop/#/Dashboard/kmC9Y/?from=6&interval_
type=HR&until=0
```

You can also find the short_code and dashboard ID in the Dashboard Properties for any dashboard, which can be accessed from the command menu ⋮. Some API operations, such as DELETE, require the dashboard ID.

ExtraHop API resources

You can perform operations on the following resources through the ExtraHop REST API. You also can view more detailed information about these resources, such as available HTTP methods, query parameters, and object properties in the REST API Explorer.

APIKey

An API key enables a user to perform operations through the ExtraHop REST API.

You can generate the initial API key for the setup user account through the REST API. All other API keys are generated through the API Access page in the Administration settings.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /apikeys	Retrieve all API keys.
POST /apikeys	Create the initial API key for the setup user account.
GET /apikeys/{keyid}	Retrieve information about a specific API key.

Operation details

`GET /apikeys`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "description": "string",
    "id": 0,
    "key": "string",
    "time_added": 0,
    "user_id": 0,
    "username": "string"
}
```

`GET /apikeys/{keyid}`

Specify the following parameters.

`keyid:` ***Number***

The unique identifier for the API key.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "description": "string",
    "id": 0,
    "key": "string",
    "time_added": 0,
    "user_id": 0,
    "username": "string"
}
```

`POST /apikeys`

Specify the following parameters.

`body:` ***Object***

The password of the setup user.

`password:` ***String***

The password for the setup user.

Specify the body parameter in the following JSON format.

```
{
    "password": "string"
}
```

Audit log

The audit log displays a record of all recorded system administration and configuration activity, such as the time of the activity, the user who performed the activity, the operation, operation details, and system component..

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /auditlog	Retrieve all audit log messages.

Operation details

`GET /auditlog`

Specify the following parameters.

`limit:` ***Number***

(Optional) The maximum number of log messages to return.

`offset:` ***Number***

(Optional) The number of log messages to skip in the results. Returns log messages starting from the offset value.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "body": {},
    "id": 0,
    "occur_time": 0,
    "time": 0
}
```

Auth

You can configure secure, single sign-on (SSO) authentication to the ExtraHop system through one or more security assertion markup language (SAML) identity providers.

When a user logs in to an ExtraHop system that is configured as a service provider (SP) for SAML SSO authentication, the ExtraHop system requests authorization from the appropriate identity provider (IdP).

The identity provider authenticates the user's credentials and then returns the authorization for the user to the ExtraHop system. The user is then able to access the ExtraHop system.

Operation	Description
GET /auth/identityproviders	Retrieve all identity providers.
POST /auth/identityproviders	Add an identity provider for remote authentication.
DELETE /auth/identityproviders/{id}	Delete a specific identity provider.
GET /auth/identityproviders/{id}	Retrieve a specific identity provider.
PATCH /auth/identityproviders/{id}	Update an existing identity provider.
GET /auth/identityproviders/{id}/privileges	Retrieve the privilege settings for a specific identity provider.
PATCH /auth/identityproviders/{id}/privileges	Update the privilege settings for a specific identity provider.
GET /auth/samlsp	Retrieve SAML security provider (SP) metadata for this ExtraHop system.

Operation details

`POST /auth/identityproviders`

Specify the following parameters.

`body:` ***Object***

Parameters for the identity provider.

`name:` ***String***

The name of the identity provider.

`enabled:` ***Boolean***

Indicates whether authentication through the identity provider is enabled on the ExtraHop system.

`entity_id:` ***String***

(Optional) The SAML 2.0 entityID.

`sso_url:` ***String***

(Optional) The SAML 2.0 Single Sign-On (SSO) URL.

`signing_certificate:` ***String***

(Optional) The SAML 2.0 X.509 signing certificate in PEM format.

`type:` ***String***

The type of identity provider.

The following values are valid:

- `saml`

`auto_provision_users:` ***Boolean***

Indicates whether a user can be created on the ExtraHop system from the identity provider.

Specify the body parameter in the following JSON format.

```
{
    "auto_provision_users": true,
    "enabled": true,
    "entity_id": "string",
```

```
    "name": "string",
    "signing_certificate": "string",
    "sso_url": "string",
    "type": "string"
}
```

`GET /auth/identityproviders`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "auto_provision_users": true,
    "enabled": true,
    "entity_id": "string",
    "id": 0,
    "name": "string",
    "signing_certificate": "string",
    "sso_url": "string",
    "type": "string"
}
```

`GET /auth/identityproviders/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the identity provider.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "auto_provision_users": true,
    "enabled": true,
    "entity_id": "string",
    "id": 0,
    "name": "string",
    "signing_certificate": "string",
    "sso_url": "string",
    "type": "string"
}
```

`PATCH /auth/identityproviders/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the identity provider.

`body:` ***Object***

The parameters for the identity provider.

`DELETE /auth/identityproviders/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the identity provider.

`GET /auth/identityproviders/{id}/privileges`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the identity provider.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "detectionsaccesslevel": {},
    "ndrlevel": {},
    "npmlevel": {},
    "packetslevel": {},
    "writelevel": {}
}
```

`PATCH /auth/identityproviders/{id}/privileges`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the identity provider.

`body:` ***Object***
An object that contains the privilege settings.

`GET /auth/samlsp`

Specify the following parameters.

`xml:` ***Boolean***
(Optional) Indicates whether to retrieve the SAML 2.0 XML metadata.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "acs_url": "string",
    "entity_id": "string",
    "xml": "string"
}
```

Cloud

This resource enables you to connect your on-premises sensors to Reveal(x) 360 For more information, see Connect to Reveal(x) 360 from self-managed sensors.

The following table displays all of the operations you can perform on this resource:

Operation	Description
POST /cloud/connect	Connect the ExtraHop system to Reveal(x) 360.

Operation details

`POST /cloud/connect`

Specify the following parameters.

`body:` ***Object***

The token you generated from Reveal(x) 360.

`cloud_token:` ***String***

The token you generated from Reveal(x) 360.

`nickname:` ***String***

A nickname to easily identify the sensor.

Specify the body parameter in the following JSON format.

```
{
    "cloud_token": "string",
    "nickname": "string"
}
```

Detections

The Detections resource enables you to retrieve detections that have been identified by the ExtraHop system.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /detections	Retrieve all detections.
GET /detections/formats	Retrieve all detection types.
GET /detections/formats/{id}	Retrieve a specific custom detection type.
POST /detections/formats	Create a new custom detection type.
DELETE /detections/formats/{id}	Delete a specific custom detection type.
PATCH /detections/formats/{id}	Update a specific custom detection type.
GET /detections/rules/hiding	Retrieve all tuning rules.
GET /detections/rules/hiding/{id}	Retrieve a specific tuning rule.
POST /detections/rules/hiding	Create a tuning rule.
DELETE /detections/rules/hiding/{id}	Delete a tuning rule.
PATCH /detections/rules/hiding/{id}	Update a tuning rule.
POST /detections/search	Retrieve detections that match the specified search criteria.
PATCH /detections/tickets	Update a ticket associated with detections.
GET /detections/{id}	Retrieve a specific detection.
GET /detections/{id}/investigations	Retrieve all investigations that a specific detection is in
PATCH /detections/{id}	Update a detection.
DELETE /detections/{id}/notes	Delete the notes for a given detection.
GET /detections/{id}/notes	Retrieve the notes for a given detection.
PUT /detections/{id}/notes	Create or replace notes for a given detection.

Operation	Description
GET /detections/{id}/related	Retrieve all detections related to a specific detection.

Operation details

`GET /detections/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the detection.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /detections`

Specify the following parameters.

`limit:` ***Number***

(Optional) Limit the number of detections returned to the specified maximum number. A random selection of detections is returned.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
```

```
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```


`POST /detections/search`

Specify the following parameters.

`body:` ***Object***

The detection search parameters.

`filter:` ***Object***

Detection-specific filters.

`category:` ***String***

Deprecated. Replaced by the categories field.

`categories:` ***Array of Strings***

Return detections from the specified categories.

`assignee:` ***Array of Strings***

Returns detections assigned to the specified user. Specify ".none" to search for unassigned detections or specify ".me" to search for detections assigned to the authenticated user.

`ticket_id:` ***Array of Strings***

Returns detections that are associated with the specified tickets. Specify ".none" to search for detections that are not associated with tickets.

`status:` ***Array of Strings***

Returns detections for tickets with the specified status. Specify ".none" to search for detections without a ticket status.

The following values are valid:

- `new`
- `in_progress`
- `closed`
- `acknowledged`

`resolution:` ***Array of Strings***

Returns detections for tickets with the specified resolution. Specify ".none" to search for detections without resolutions.

The following values are valid:

- `action_taken`
- `no_action_taken`

`types:` ***Array of Strings***

Returns detections with the specified types.

`risk_score_min:` ***Number***

Returns detections with risk scores greater than or equal to the specified value.

`recommended:` ***Boolean***

Returns detections recommended for triage. This field is valid only on a console.

`from:` ***Number***

Returns detections that occurred after the specified date, expressed in milliseconds since the epoch. Detections that started before the specified date are returned if the detection was ongoing at that time.

`limit:` ***Number***

Returns no more than the specified number of detections.

`offset:` ***Number***

The number of detections to skip for pagination.

`sort:` ***Array of Objects***

Sorts returned detections by the specified fields. By default, detections are sorted by most recent update time and then ID in ascending order.

`direction:` ***String***

The order in which returned detections are sorted.

The following values are valid:

- `asc`
- `desc`

`field:` ***String***

The field to sort detections by.

`until:` ***Number***

Return detections that ended before the specified date, expressed in milliseconds since the epoch.

`update_time:` ***Number***

Returns detections related to events that occurred after the specified date, expressed in milliseconds since the epoch. Note that the ExtraHop Machine Learning Service analyzes historical data to generate detections, and so there is a time delay between when the events that cause those detections occur and when the detections are generated. If you search for detections in the same update_time window multiple times, the later search might return detections that were not returned by the earlier search.

`mod_time:` ***Number***

Returns detections that were updated after the specified date, expressed in milliseconds since the epoch.

`create_time:` ***Number***

Returns detections that were created after the specified date, expressed in milliseconds since the epoch. For sensors, this returns detections that were generated after the specified date.

For consoles, this returns detections that were first synchronized to the console after the specified date.

`id_only:` ***Boolean***

(Optional) Returns only the IDs of the detections.

Specify the body parameter in the following JSON format.

```
{
    "create_time": 0,
    "filter": {
        "category": "string",
        "categories": [],
        "assignee": [],
        "ticket_id": [],
        "status": [],
        "resolution": [],
        "types": [],
        "risk_score_min": 0,
        "recommended": true
    },
    "from": 0,
    "id_only": true,
    "limit": 0,
    "mod_time": 0,
    "offset": 0,
    "sort": {
        "direction": "string",
        "field": "string"
    },
    "until": 0,
    "update_time": 0
}
```

`PATCH /detections/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the detection.

`body:` ***Object***

The detection parameters to update.

`ticket_id:` ***String***

The ID of the ticket associated with the detection.

`assignee:` ***String***

The assignee of the detection or the ticket associated with the detection.

`status:` ***String***

The status of the detection or the ticket associated with the detection.

The following values are valid:

- `new`
- `in_progress`
- `closed`
- `acknowledged`

`resolution:` ***String***

The resolution of the detection or the ticket associated with the detection.

The following values are valid:

- `action_taken`
- `no_action_taken`

`participants:` ***Array of Objects***

A list of devices and applications associated with the detection. You can modify specific fields for a participant, but you cannot add new participants to a detection.

`id:` ***Number***

The ID of the participant associated with the detection.

`usernames:` ***Array of Strings***

The usernames associated with the participant through the REST API.

`origins:` ***Array of Strings***

The origin IP addresses associated with the participant through the REST API.

Specify the body parameter in the following JSON format.

```
{
    "assignee": "string",
    "participants": {
        "id": 0,
        "usernames": [],
        "origins": []
    },
    "resolution": "string",
    "status": "string",
    "ticket_id": "string"
}
```

`PATCH /detections/tickets`

Specify the following parameters.

`body:` ***Object***

The detection ticketing values to update.

`ticket_id:` ***String***

The ID of the ticket associated with the detection.

`assignee:` ***String***

The assignee of the ticket associated with the detection.

`status:` ***String***

The status of the ticket associated with the detection.

The following values are valid:

- `new`
- `in_progress`
- `closed`
- `acknowledged`

`resolution:` ***String***

The resolution of the ticket associated with the detection.

The following values are valid:

- `action_taken`
- `no_action_taken`

Specify the body parameter in the following JSON format.

```
{
    "assignee": "string",
    "resolution": "string",
    "status": "string",
    "ticket_id": "string"
}
```

`GET /detections/{id}/related`

Specify the following parameters.

`id:` ***Number***

The ID of the detection to retrieve related detections for.

`from:` ***Number***

Returns detections that occurred after the specified date, expressed in milliseconds since the epoch. Detections that started before the specified date are returned if the detection was ongoing at that time.

`until:` ***Number***

Return detections that ended before the specified date, expressed in milliseconds since the epoch.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /detections/{id}/investigations`

Specify the following parameters.

`id:` ***Number***

The ID of the detection to retrieve related investigations for.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /detections/formats`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "author": "string",
    "categories": [],
    "display_name": "string",
    "is_user_created": true,
    "last_updated": 0,
    "mitre_categories": [],
    "properties": {},
    "released": 0,
    "status": "string",
    "type": "string"
}
```

`GET /detections/formats/{id}`

Specify the following parameters.

`id:` ***String***

The string identifier of the detection format.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "author": "string",
    "categories": [],
    "display_name": "string",
    "is_user_created": true,
    "last_updated": 0,
    "mitre_categories": [],
    "properties": {},
    "released": 0,
    "status": "string",
    "type": "string"
}
```

`POST /detections/formats`

Specify the following parameters.

`body:` ***Object***

The parameters of the detection format.

`type:` ***String***

A string identifier for the detection type. The string can only contain letters, numbers, and underscores. Although detection types are unique across built-in formats, and detection types are unique across custom formats, a built-in and custom format can share the same detection type.

`display_name:` ***String***

The display name of the detection type that appears on the Detections page in the ExtraHop system.

`mitre_categories:` ***Array of Strings***

(Optional) The IDs of the MITRE techniques associated with the detection.

`author:` ***String***

(Optional) The author of the detection format.

`categories:` ***Array of Strings***

(Optional) The list of categories the detection belongs to. For POST and PATCH operations, specify a list with a single string. You cannot specify more than one category for custom detection formats. The "perf" or "sec" category is automatically added to all detection formats.

Specify the body parameter in the following JSON format.

```
{
    "author": "string",
    "categories": [],
    "display_name": "string",
    "mitre_categories": [],
    "type": "string"
}
```

`DELETE /detections/formats/{id}`

Specify the following parameters.

`id:` ***String***

The string identifier of the detection format.

`PATCH /detections/formats/{id}`

Specify the following parameters.

`id:` ***String***

The string identifier of the detection format.

`body:` ***Object***

The parameters of the detection format.

`GET /detections/rules/hiding`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "author": "string",
    "create_time": 0,
    "description": "string",
    "detection_type": "string",
    "detections_hidden": 0,
    "enabled": true,
    "expiration": 0,
    "hide_past_detections": true,
    "id": 0,
    "offender": {},
    "participants_hidden": 0,
    "properties": [],
    "victim": {}
}
```

`GET /detections/rules/hiding/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the tuning rule.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "author": "string",
    "create_time": 0,
    "description": "string",
    "detection_type": "string",
    "detections_hidden": 0,
    "enabled": true,
    "expiration": 0,
    "hide_past_detections": true,
    "id": 0,
    "offender": {},
    "participants_hidden": 0,
    "properties": [],
    "victim": {}
}
```

`POST /detections/rules/hiding`

Specify the following parameters.

`body:` ***Object***

The tuning rule parameters.

`offender:` ***Object***

The offender that this tuning rule applies to. Specify a detection_hiding_participant object to apply the rule to a specific victim, or specify "Any" to apply the rule to any offender.

`object_type:` ***String***

The type of participant.

The following values are valid:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Number***

The ID for the device, device group, or network locality. This option is valid only if the object_type is "device", "device_group", or "network_locality".

`object_value:` ***Array or String***

The IP address or CIDR block of the participant. You can specify a single address or block in a string or multiple addresses or blocks in an array. This option is valid only if the object_type is "ipaddr".

`object_locality:` ***String***

The network locality type of the participant. Specify either "external" or "internal". This option is valid only if the object_type is "locality_type".

The following values are valid:

- `internal`
- `external`

`object_scanner:` ***Array or String***

The name of an external scanning service. You can specify a single service in a string or multiple values in an array. You can also specify "Any" to select any scanning service. This option is valid only if the object_type is "scanner_service".

`object_hostname:` ***Array or String***

The hostname of a participant. You can specify a single hostname in a string or multiple hostnames in an array. This option is valid only if the object_type is "hostname".

`victim:` ***Object***

The victim that this tuning rule applies to. Specify a detection_hiding_participant object to apply the rule to a specific victim, or specify "Any" to apply the rule to any victim.

`object_type:` ***String***

The type of participant.

The following values are valid:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`

- `scanner_service`

`object_id:` ***Number***

The ID for the device, device group, or network locality. This option is valid only if the object_type is "device", "device_group", or "network_locality".

`object_value:` ***Array or String***

The IP address or CIDR block of the participant. You can specify a single address or block in a string or multiple addresses or blocks in an array. This option is valid only if the object_type is "ipaddr".

`object_locality:` ***String***

The network locality type of the participant. Specify either "external" or "internal". This option is valid only if the object_type is "locality_type".

The following values are valid:

- `internal`
- `external`

`object_scanner:` ***Array or String***

The name of an external scanning service. You can specify a single service in a string or multiple values in an array. You can also specify "Any" to select any scanning service. This option is valid only if the object_type is "scanner_service".

`object_hostname:` ***Array or String***

The hostname of a participant. You can specify a single hostname in a string or multiple hostnames in an array. This option is valid only if the object_type is "hostname".

`expiration:` ***Number***

The time that the tuning rule expires, expressed in milliseconds since the epoch. A value of null or 0 indicates that the rule does not expire.

`description:` ***String***

(Optional) The description of the tuning rule.

`detection_type:` ***String***

The type of detection that this tuning rule applies to. View a list of valid fields for "type" by running the GET /detections/formats operation. Specify "all_performance" or "all_security" to apply the rule to all performance or all security detections.

`properties:` ***Array of Objects***

(Optional) The filter criteria for detection properties.

`property:` ***String***

The name of the property to filter.

`operator:` ***String***

The compare method applied when matching the operand value against the detection property value.

The following values are valid:

- `=`
- `!=`
- `~`
- `!~`
- `in`

`operand:` ***String or Number or Object***

The value that the filter attempts to match. The filter compares the value of the operand to the value of the detection property and applies the compare method

specified by the operator parameter. You can specify the operand as a string, integer, or object. For more information, see the REST API Guide.

Specify the body parameter in the following JSON format.

```
{
    "description": "string",
    "detection_type": "string",
    "expiration": 0,
    "offender": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    },
    "properties": {
        "property": "string",
        "operator": "string",
        "operand": "string"
    },
    "victim": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    }
}
```

`PATCH /detections/rules/hiding/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the tuning rule.

`body:` ***Object***

The tuning rule fields to update.

`enabled:` ***Boolean***

Indicates whether the tuning rule is enabled.

`expiration:` ***Number***

The time that the tuning rule expires, expressed in milliseconds since the epoch. A value of null or 0 indicates that the rule does not expire.

`description:` ***String***

The description of the tuning rule.

`offender:` ***Object***

The offender that this tuning rule applies to. Specify a detection_hiding_participant object to apply the rule to a specific victim, or specify "Any" to apply the rule to any offender.

`object_type:` ***String***

The type of participant.

The following values are valid:

- `device`
- `device_group`

- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Number***

The ID for the device, device group, or network locality. This option is valid only if the object_type is "device", "device_group", or "network_locality".

`object_value:` ***Array or String***

The IP address or CIDR block of the participant. You can specify a single address or block in a string or multiple addresses or blocks in an array. This option is valid only if the object_type is "ipaddr".

`object_locality:` ***String***

The network locality type of the participant. Specify either "external" or "internal". This option is valid only if the object_type is "locality_type".

The following values are valid:

- `internal`
- `external`

`object_scanner:` ***Array or String***

The name of an external scanning service. You can specify a single service in a string or multiple values in an array. You can also specify "Any" to select any scanning service. This option is valid only if the object_type is "scanner_service".

`object_hostname:` ***Array or String***

The hostname of a participant. You can specify a single hostname in a string or multiple hostnames in an array. This option is valid only if the object_type is "hostname".

`victim:` ***Object***

The victim that this tuning rule applies to. Specify a detection_hiding_participant object to apply the rule to a specific victim, or specify "Any" to apply the rule to any victim.

`object_type:` ***String***

The type of participant.

The following values are valid:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Number***

The ID for the device, device group, or network locality. This option is valid only if the object_type is "device", "device_group", or "network_locality".

`object_value:` ***Array or String***

The IP address or CIDR block of the participant. You can specify a single address or block in a string or multiple addresses or blocks in an array. This option is valid only if the object_type is "ipaddr".

`object_locality:` ***String***

The network locality type of the participant. Specify either "external" or "internal". This option is valid only if the object_type is "locality_type".

The following values are valid:

- `internal`
- `external`

`object_scanner:` ***Array or String***

The name of an external scanning service. You can specify a single service in a string or multiple values in an array. You can also specify "Any" to select any scanning service. This option is valid only if the object_type is "scanner_service".

`object_hostname:` ***Array or String***

The hostname of a participant. You can specify a single hostname in a string or multiple hostnames in an array. This option is valid only if the object_type is "hostname".

`properties:` ***Array of Objects***

The filter criteria for detection properties.

`property:` ***String***

The name of the property to filter.

`operator:` ***String***

The compare method applied when matching the operand value against the detection property value.

The following values are valid:

- `=`
- `!=`
- `~`
- `!~`
- `in`

`operand:` ***String or Number or Object***

The value that the filter attempts to match. The filter compares the value of the operand to the value of the detection property and applies the compare method specified by the operator parameter. You can specify the operand as a string, integer, or object. For more information, see the REST API Guide.

Specify the body parameter in the following JSON format.

```
{
    "description": "string",
    "enabled": true,
    "expiration": 0,
    "offender": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    },
    "properties": {
        "property": "string",
        "operator": "string",
        "operand": "string"
    },
    "victim": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
```

```
        "object_hostname": "array"
    }
}
```

`DELETE /detections/rules/hiding/{id}`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the tuning rule.

`GET /detections/{id}/notes`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the detection.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "author": "string",
    "note": "string",
    "update_time": 0
}
```

`DELETE /detections/{id}/notes`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the detection.

`PUT /detections/{id}/notes`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the detection.

`body:` ***Object***
The detection note parameters.

Operand values for detection property tuning rules

The `POST /detections/rules/hiding` operation enables you to create tuning rules that filter detections based on detection properties. You can specify filtering criteria for detection properties in objects. Each object should contain a unique value for the `operand` field that is valid for the specified `property` value.

Tip: You can retrieve valid property values through the `GET /detections/formats` operation. See the keys of the `properties` object in the response. In the following example, the `property` value is `s3_bucket`:

```
    "properties": {
      "s3_bucket": {
        "is_optional": true,
        "status": "active",
        "is_tunable": true,
        "data_type": "string"
```

```
        }
    }
```

The `is_tunable` field indicates whether you can create a tuning rule based on the property.

`registered_domain_name`

To hide rules by a registered domain name, specify the `property` value as `registered_domain_name` and the `operand` value as a domain name.

The following example rule hides DNS Tunnel detections for `example.com`.

```
{
    "detection_type": "dns_tunnel",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": "example.com",
            "operator": "=",
            "property": "registered_domain_name"
        }
    ]
}
```

`uris`

To hide rules by a URI, specify the `property` value as `uris` and the `operand` value as a URI.

The following example rule hides SQL Injection (SQLi) Attack detections for `http://example.com/test`.

```
{
    "detection_type": "sqli_attack",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": "http://example.com/test",
            "operator": "=",
            "property": "uris"
        }
      ]
}
```

`top_level_domain`

To hide rules by a top-level domain name, specify the `property` value as `top_level_domain` and the `operand` value as a top-level domain name.

The following example rule hides Suspicious Top-level Domain detections for the `org` top-level domain.

```
{
    "detection_type": "suspicious_tld",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": "org",
```

```
            "operator": "=",
            "property": "top_level_domain"
        }
    ]
}
```

Search with regular expressions (regex)

For certain `property` values, the string can be in regex syntax. Specify the `operand` value as an object that has a `value` parameter with the regex syntax you want to match and an `is_regex` parameter that is set to `true`. The following rule filters DNS Tunnel detections with domain names that end with `example.com`.

```
{
    "detection_type": "dns_tunnel",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": {
                "value": ".*?example.com",
                "is_regex": true
            },
            "operator": "=",
            "property": "registered_domain_name"
        }
    ]
}
```

Disable case sensitivity

By default, searches for string `property` values are case-sensitive. However, you can disable case sensitivity by specifying the operand value as an object that has a `case_sensitive` parameter that is set to `false`.

The following rule hides Hacking Tool Domain Access detections with the ArchStrike hacking tool.

```
{
    "detection_type": "hacking_tools",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": {
                "value": "archstrike",
                "case_sensitive": false
            },
            "operator": "=",
            "property": "hacking_tool"
        }
    ]
}
```

Email group

You can add individual or group email addresses to an email group and assign them to a system alert. When that alert is triggered, the system sends an email to all of the addresses in the email group.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /emailgroups	Retrieve all email groups.
POST /emailgroups	Create a new email group.
DELETE /emailgroups/{id}	Delete a email group by a unique identifier.
GET /emailgroups/{id}	Retrieve a specific email group by a unique identifier.
PATCH /emailgroups/{id}	Apply updates to a specific email group.

Operation details

`GET /emailgroups`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "email_addresses": [],
    "group_name": "string",
    "id": 0,
    "system_notifications": true
}
```

`POST /emailgroups`

Specify the following parameters.

`body:` ***Object***

Apply the specified property values to the new email group.

`group_name:` ***String***

The friendly name for the email group.

`email_addresses:` ***Array of Strings***

The list of email addresses in the email group.

`system_notifications:` ***Boolean***

Indicates whether that the group should receive system notifications.

Specify the body parameter in the following JSON format.

```
{
    "email_addresses": [],
    "group_name": "string",
    "system_notifications": true
}
```

`GET /emailgroups/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier of the email group.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "email_addresses": [],
    "group_name": "string",
    "id": 0,
    "system_notifications": true
}
```

`DELETE /emailgroups/{id}`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the email group.

`PATCH /emailgroups/{id}`

Specify the following parameters.

`body:` ***Object***
Apply the specified property value updates to the email group.

`id:` ***Number***
The unique identifier for the email group.

ExtraHop

This resource provides metadata about the ExtraHop system.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /extrahop	Retrieve metadata about the firmware running on the ExtraHop system.
POST /extrahop/cloudresources	Manually update resources on the ExtraHop system. These resources are automatically updated when the system is connected to ExtraHop Cloud Services.
GET /extrahop/cluster	Retrieve Explore cluster configuration settings.
PATCH /extrahop/cluster	Update Explore cluster configuration settings.
GET /extrahop/detections/access	Retrieve the detections access control settings.
PUT /extrahop/detections/access	Update detections access control settings.
GET /extrahop/edition	Retrieve the edition of the ExtraHop system. **Note:** This operation does not require an API key.
POST /extrahop/firmware	Upload a new firmware image to the ExtraHop system. For more information, see Upgrade ExtraHop firmware through the REST API.
POST /extrahop/firmware/download/url	Download a new firmware image onto the ExtraHop system from a URL.

Operation	Description
POST /extrahop/firmware/download/version	Download a new firmware image onto the ExtraHop system from ExtraHop Cloud Services.
POST /extrahop/firmware/latest/upgrade	Upgrade the ExtraHop system to the most recently uploaded firmware image.
GET /extrahop/firmware/next	Upgrade the ExtraHop system to the most recently uploaded firmware image.
GET /extrahop/firmware/previous	Retrieve information about a firmware version that you can roll back the ExtraHop system to.
POST /extrahop/firmware/previous/rollback	Roll back the ExtraHop system to the previous firmware version.
GET /extrahop/flowlogs/secret	Retrieve the flow log secret.
POST /extrahop/flowlogs/secret	Generate a new flow log secret.
GET /extrahop/idrac	Retrieve the iDRAC IP address of the ExtraHop system.
GET /extrahop/platform	Retrieve the platform name of the ExtraHop system.  **Note:** This operation does not require an API key.
GET /extrahop/processes	Retrieve a list of processes running on the ExtraHop system.
POST /extrahop/processes/{process}/restart	Restart a process running on the ExtraHop system.
GET /extrahop/services	Retrieve settings for all services.
PATCH /extrahop/services	Update the settings for services.
POST /extrahop/restart	Restart the ExtraHop system.
POST /extrahop/shutdown	Shut down the ExtraHop system.
POST /extrahop/sslcert	Regenerate the SSL certificate on the ExtraHop system. For more information, see Create a trusted SSL certificate through the REST API
PUT /extrahop/sslcert	Replace the SSL certificate on the ExtraHop system.
POST /extrahop/sslcert/signingrequest	Create an SSL certificate signing request. For more information, see Create a trusted SSL certificate through the REST API.
GET /extrahop/ticketing	Retrieve the ticketing integration status.
PATCH /extrahop/ticketing	Enable or disable ticketing integration.
GET /extrahop/version	Retrieve the version of the firmware running on the ExtraHop system.  **Note:** This operation does not require an API key.

Operation details

`GET /extrahop/version`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "version": "string"
}
```

`GET /extrahop/platform`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "platform": "string"
}
```

`GET /extrahop/edition`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "edition": "string"
}
```

`GET /extrahop`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "display_host": "string",
    "external_hostname": "string",
    "hostname": "string",
    "mgmt_ipaddr": "string",
    "platform": "string",
    "version": "string"
}
```

`GET /extrahop/idrac`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "ipaddr": "string"
}
```

`POST /extrahop/sslcert`

There are no parameters for this operation.

`PUT /extrahop/sslcert`

Specify the following parameters.

`body:` ***String***

The SSL certificate and optionally the private key. Enter as plain text, separated with a line break.

`POST /extrahop/sslcert/signingrequest`

Specify the following parameters.

`body:` ***Object***

Parameters for the SSL certificate signing request.

`subject_alternative_names:` ***Array of Objects***

A list of names that the certificate applies to, such as {"type": "dns", "name": "www.example.com"}.

`type:` ***String***

Type of Subject Alternative Name.

The following values are valid:

- `dns`
- `ip`

`name:` ***String***

Name of Subject Alternative Name.

`subject:` ***Object***

The subject of the SSL certificate. For a list of certificate subject fields, see below.

`common_name:` ***String***

The subject common name (CN).

`country_code:` ***String***

(Optional) The subject country (C).

`state_or_province_name:` ***String***

(Optional) The subject state or province (ST).

`locality_name:` ***String***

(Optional) The subject locality (L).

`organization_name:` ***String***

(Optional) The subject organization (O).

`organizational_unit_name:` ***String***

(Optional) The subject organizational unit (OU).

`email_address:` ***String***

(Optional) The subject e-mail address (emailAddress).

Specify the body parameter in the following JSON format.

```
{
    "subject": {
        "common_name": "string",
        "country_code": "string",
        "state_or_province_name": "string",
        "locality_name": "string",
        "organization_name": "string",
        "organizational_unit_name": "string",
        "email_address": "string"
    },
```

```
    "subject_alternative_names": {
        "type": "string",
        "name": "string"
    }
}
```

`GET /extrahop/ticketing`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "enabled": true,
    "external_ticketing_enabled": true,
    "internal_ticketing_enabled": true,
    "url_template": "string"
}
```

`PATCH /extrahop/ticketing`

Specify the following parameters.

`body:` ***Object***

Ticket tracking settings.

`enabled:` ***Boolean***

(Optional) Deprecated. Replaced by the external_ticketing_enabled and internal_ticketing_enabled fields.

`external_ticketing_enabled:` ***Boolean***

(Optional) Indicates whether investigations are tracked from an external ticketing system.

`internal_ticketing_enabled:` ***Boolean***

(Optional) Indicates whether investigations are tracked from within the ExtraHop System.

`url_template:` ***String***

(Optional) The URL template that links detections to external tickets. The template must include the $ticket_id variable. This field applies only if detection investigations are tracked from an external ticketing system.

Specify the body parameter in the following JSON format.

```
{
    "enabled": true,
    "external_ticketing_enabled": true,
    "internal_ticketing_enabled": true,
    "url_template": "string"
}
```

`PUT /extrahop/detections/access`

Specify the following parameters.

`body:` ***Object***

The detections access settings for the appliance.

`enabled:` ***Boolean***

Indicates whether detections access settings are enabled. When enabled, administrators can restrict detections access for specified users. You cannot disable detections access settings after the settings are enabled.

Specify the body parameter in the following JSON format.

```
{
    "enabled": true
}
```

`GET /extrahop/detections/access`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "enabled": true
}
```

`POST /extrahop/firmware`

Specify the following parameters.

`firmware:` ***Filename***

The .tar file that contains the firmware image. Note: You cannot upload a firmware image through the REST API explorer. For more information about how to upload an image through cURL or a Python script, see Upgrade ExtraHop firmware through the REST API.

`POST /extrahop/firmware/latest/upgrade`

Specify the following parameters.

`body:` ***Object***

(Optional) The installation options for upgrading the appliance.

`restart_after:` ***Boolean***

(Optional) Indicates whether to restart the appliance after the upgrade is complete.

`silent:` ***Boolean***

(Optional) Specifies whether to disable the ExtraHop Web UI during the upgrade process. If an upgrade fails, the appliance will automatically revert to the previous firmware version.

`force:` ***Boolean***

(Optional) Specifies whether to skip compatibility verification. Skip verification only if ExtraHop Support has reviewed and approved the upgrade.

Specify the body parameter in the following JSON format.

```
{
    "force": true,
    "restart_after": true,
    "silent": true
}
```

`POST /extrahop/firmware/download/url`

Specify the following parameters.

`body:` ***Object***

The download options.

`firmware_url:` ***String***

The URL of the firmware to download. HTTPS, HTTP, and FTP schemes are supported.

`upgrade:` ***Boolean***

(Optional) Specifies whether to upgrade the appliance after the firmware download is complete.

`force:` ***Boolean***

(Optional) Specifies whether to skip compatibility verification. Skip verification only if ExtraHop Support has reviewed and approved the upgrade.

Specify the body parameter in the following JSON format.

```
{
    "firmware_url": "string",
    "force": true,
    "upgrade": true
}
```

`POST /extrahop/restart`

There are no parameters for this operation.

`POST /extrahop/shutdown`

There are no parameters for this operation.

`GET /extrahop/services`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "admin": {
        "enabled": true
    },
    "keyreceiver": {
        "enabled": true
    },
    "snmp": {
        "enabled": true
    },
    "ssh": {
        "enabled": true
    }
}
```

`PATCH /extrahop/services`

Specify the following parameters.

`body:` ***Object***

The settings for services.

`admin:` ***Object***

(Optional) The settings of the Management GUI service, which provides browser-based access to the appliance.

`enabled:` ***Boolean***

Indicates whether the service is enabled.

`snmp:` ***Object***

(Optional) The settings of the SNMP service, which enables your network device monitoring software to collect information from the ExtraHop System.

`enabled:` ***Boolean***

Indicates whether the service is enabled.

`ssh:` ***Object***

(Optional) The settings of the SSH service, which enables users to securely log in to the ExtraHop command-line interface (CLI).

`enabled:` ***Boolean***

Indicates whether the service is enabled.

`keyreceiver:` ***Object***

(Optional) The settings of the SSL Session Key Receiver, which enables the appliance to receive and decrypt session keys from the session key forwarder.

`enabled:` ***Boolean***

Indicates whether the service is enabled.

Specify the body parameter in the following JSON format.

```
{
    "admin": {
        "enabled": true
    },
    "keyreceiver": {
        "enabled": true
    },
    "snmp": {
        "enabled": true
    },
    "ssh": {
        "enabled": true
    }
}
```

`GET /extrahop/processes`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "can_restart": true,
    "cpu": 0.0,
    "cpu_time": 0,
    "mem_percent": 0.0,
    "mem_res": 0,
    "mem_virt": 0,
    "process": "string",
    "start_time": 0
}
```

`POST /extrahop/processes/{process}/restart`

Specify the following parameters.

`process:` ***String***

The name of the process.

The following values are valid:

- `exadmin`
- `exalerts`
- `examf`
- `exapi`
- `exbridge`
- `excap`
- `exconfig`
- `exflowlogs`
- `exsnmpq`
- `exnotify`
- `exportal`
- `exremote`
- `exsearch`
- `exstatmirror`
- `extrend`
- `webserver`
- `hopcloud-api`

`GET /extrahop/cluster`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "ingest_enabled": true,
    "replication_policy": 0
}
```

`PATCH /extrahop/cluster`

Specify the following parameters.

`body:` ***Object***

The EXA cluster configuration settings.

`ingest_enabled:` ***Boolean***

(Optional) Indicates whether record ingest is enabled for the Explore cluster.

`replication_policy:` ***Number***

(Optional) The replication level that determines how many copies of each record are stored.

Specify the body parameter in the following JSON format.

```
{
    "ingest_enabled": true,
    "replication_policy": 0
}
```

`GET /extrahop/firmware/previous`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
```

```
    "backup_time": 0,
    "version": "string"
}
```

`POST /extrahop/firmware/previous/rollback`

There are no parameters for this operation.

`POST /extrahop/cloudresources`

Specify the following parameters.

`cloudresources:` ***Filename***

The resource bundle file.

`GET /extrahop/flowlogs/secret`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "secret": "string"
}
```

`POST /extrahop/flowlogs/secret`

There are no parameters for this operation.

`GET /extrahop/firmware/next`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "current_release": true,
    "release": "string",
    "versions": []
}
```

`POST /extrahop/firmware/download/version`

Specify the following parameters.

`body:` ***Object***

(Optional) The download options.

`version:` ***String***

The version of the firmware to download.

`upgrade:` ***Boolean***

(Optional) Specifies whether to upgrade the appliance after the firmware download is complete.

Specify the body parameter in the following JSON format.

```
{
    "upgrade": true,
    "version": "string"
```

```
}
```

Jobs

You can monitor the progress of some administration jobs started through the REST API. If a REST request creates a job, the job ID is returned in the `location` header of the response. The following operations create jobs:

- `POST /extrahop/firmware/latest/upgrade`
- `POST /extrahop/sslcert`

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /jobs	Retrieve the status of all jobs.
GET /jobs/{id}	Retrieve the status of a specific job.

Operation details

`GET /jobs/{id}`

Specify the following parameters.

`id:` ***String***

The unique identifier for the job.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "details": "string",
    "id": "string",
    "remote_jobs": [],
    "status": "string",
    "step_description": "string",
    "step_number": 0,
    "total_steps": 0,
    "type": "string"
}
```

`GET /jobs`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "details": "string",
    "id": "string",
    "remote_jobs": [],
    "status": "string",
    "step_description": "string",
    "step_number": 0,
    "total_steps": 0,
    "type": "string"
}
```

Job types

The `GET /jobs` operation returns the following values in the `type` field of the response.

extrahop_firmware_download

The ExtraHop system is downloading a new firmware image from either a URL or ExtraHop Cloud Services.

extrahop_firmware_upgrade

The ExtraHop system is upgrading to a new firmware version.

extrahop_firmware_download_upgrade

The ExtraHop system is downloading a firmware image and upgrading to a new firmware version. The image is retrieved from either a URL or ExtraHop Cloud Services.

Note: The `type` field is empty for some jobs.

License

This resource enables you to retrieve and set product keys or to retrieve and set a license.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /license	Retrieve the license applied to this ExtraHop system.
PUT /license	Apply and register a new license to the ExtraHop system.
GET /license/productkey	Retrieve the product key to this ExtraHop system.
PUT /license/productkey	Apply the specified product key to the ExtraHop system and register the license.

Operation details

`PUT /license`

Specify the following parameters.

`body:` ***String***

(Optional) The license text provided to you by ExtraHop Support, including the BEGIN and END lines.

`GET /license`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "dossier": "string",
    "edition": "string",
    "expires_at": 0,
    "expires_in": 0,
    "modules": {},
    "options": {},
    "platform": "string",
    "product_key": "string",
```

```
    "serial": "string"
}
```

`PUT /license/productkey`

Specify the following parameters.

`body:` ***Object***

(Optional) Apply the specified product key to the appliance.

`GET /license/productkey`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "product_key": "string"
}
```

Metrics

Metrics information is collected about every object identified by the ExtraHop system.

Note that metrics are retrieved through the POST method, which creates a query to collect the requested information through the API. For more information, see Extract metrics through the REST API.

The following table displays all of the operations you can perform on this resource:

Operation	Description
POST /metrics	Retrieves metrics for each specified object.
GET /metrics/next/{xid}	If you request activity group metrics from a consolewith the `POST /metrics`, `POST /metrics/total`, or `POST /metrics/totalbyobject` operation and the response contains the `xid` field, the `GET /metrics/next/{xid}` operation returns metrics from one of the sensors connected to the console. Repeat the `GET /metrics/next/{xid}` operation to return metrics from additional sensors. After all metrics are retrieved, the operation returns null.
POST /metrics/total	Retrieves combined metric totals for all specified objects.
POST /metrics/totalbyobject	Retrieves metric totals for each specified object.

For example, the following request body retrieves HTTP responses that two devices sent over the last 30 minutes.

```
{
  "cycle": "auto",
  "from": -1800000,
  "metric_category": "http_server",
  "metric_specs": [
```

```
    {
      "name": "rsp"
    }
  ],
  "object_ids": [
    180, 177
  ],
  "object_type": "device",
  "until": 0
}
```

For the `POST /metrics` operation, the previous example request body returns counts of HTTP responses that occurred during each time interval, labeled with the time of each event and the ID of the device that sent the responses, similar to the following example response:

```
{
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1709659320000,
  "from": 1709657520000,
  "until": 1709659320000,
  "stats": [
    {
      "oid": 177,
      "time": 1709657520000,
      "duration": 30000,
      "values": [
        4
      ]
    },
    {
      "oid": 177,
      "time": 1709657550000,
      "duration": 30000,
      "values": [
        4
      ]
    },
    {
      "oid": 180,
      "time": 1709657520000,
      "duration": 30000,
      "values": [
        4
      ]
    },
    {
      "oid": 180,
      "time": 1709657550000,
      "duration": 30000,
      "values": [
        4
      ]
    }
  ]
}
```

For the `POST /metrics/totalbyobject` operation, the same previous example request body retrieves the combined total for each device over the entire time period, similar to the following example response:

```
{
  "cycle": "30sec",
```

```
  "node_id": 0,
  "clock": 1709659620000,
  "from": 1709657820000,
  "until": 1709659620000,
  "stats": [
    {
      "oid": 180,
      "time": 1709659620000,
      "duration": 1830000,
      "values": [
        8
      ]
    },
    {
      "oid": 177,
      "time": 1709659620000,
      "duration": 1830000,
      "values": [
        8
      ]
    }
  ]
}
```


For the `POST /metrics/total` operation, the same previous example request body retrieves the combined total of both devices over the entire time period, similar to the following example response:


```
{
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1709659830000,
  "from": 1709658030000,
  "until": 1709659830000,
  "stats": [
    {
      "oid": -1,
      "time": 1709659830000,
      "duration": 1830000,
      "values": [
        16
      ]
    }
  ]
}
```


Note that the behavior of the `/metrics/total` and `/metrics/totalbyobject` endpoints depends on the type of the metric. For count metrics, the `values` field contains a sum total of values over the specified time interval, as shown in the example above. However, for dataset metrics, the `values` field contains a list of values and the frequency that those values appeared. For example, a query for server processing times with the `POST /metrics/total` operation returns a response similar to the following example:


```
{
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1494541440000,
  "from": 1494539640000,
  "until": 1494541440000,
  "stats": [
    {
      "oid": -1,
      "time": 1494541380000,
      "duration": 1800000,
```

```
      "values": [
        [
          {
            "value": 2.271,
            "freq": 5
          },
          {
            "value": 48.903,
            "freq": 1
          }
        ]
      ]
    }
  ]
}
```

If there are more than 1,000 distinct dataset values over the specified time period, similar values are consolidated to reduce the response to 1,000 values. For example, if there are less than 1,000 values, the response might contain the following entries:

```
{
  "value": 2.571,
  "freq": 4
},
{
  "value": 2.912,
  "freq": 2
}
```

However, if the response contains more than 1,000 values, those entries might be consolidated into the following entry:

```
{
  "value": 2.571,
  "freq": 6
}
```

If the `calc_type` field is specified and the response contains more than 1,000 values, the percentile or mean is calculated according to the consolidated dataset.

Operation details

`POST /metrics`

Specify the following parameters.

`body:` ***Object***

The description of the metric request.

`from:` ***Number***

The beginning timestamp for the request. Return only metrics collected after this time. Time is expressed in milliseconds since the epoch. 0 indicates the time of the request. A negative value is evaluated relative to the current time. The default unit for a negative value is milliseconds, but other units can be specified with a unit suffix. See the REST API Guide for supported time units and suffixes.

`until:` ***Number***

The ending timestamp for the request. Return only metrics collected before this time. Follows the same time value guidelines as the from parameter.

`cycle:` ***String***

The aggregation period for metrics.

The following values are valid:

- `auto`
- `1sec`
- `30sec`
- `5min`
- `1hr`
- `24hr`

`object_type:` ***String***

Indicates the object type of unique identifiers specified in the object_ids property.

The following values are valid:

- `network`
- `device`
- `application`
- `vlan`
- `device_group`
- `system`

`object_ids:` ***Array of Numbers***

The list of numeric values that represent unique identifiers. Unique identifiers can be retrieved through the /networks, /devices, /applications, /vlans, /devicegroups, /activitygroups, and /appliances resources. For system health metrics, specify the ID of the sensor or console and set the object_type parameter to "system".

`metric_category:` ***String***

The group of metrics that are searchable in the metric catalog.

`metric_specs:` ***Array of Objects***

An array of metric specification objects.

`name:` ***String***

The field name for the metric. When filtering in the metric catalog on a metric_category, each result is a potential metric_spec name. When a result is selected from the catalog, the "Metric" field value is a valid option for this field.

`key1:` ***String***

(Optional) Filter detail metrics. Detail metrics break down data through keys, which are strings or IP addresses. For example, the metric "HTTP Requests by Method" accepts a key1 value of "GET." Keys can also be regular expressions that are delimited with forward slashes ("/GET/").

`key2:` ***String***

(Optional) Enable additional filtering on detail metrics.

`calc_type:` ***String***

(Optional) The type of calculation to perform.

The following values are valid:

- `mean`
- `percentiles`

`percentiles:` ***Array of Numbers***

(Optional) The list of percentiles, sorted in ascending order, which should be returned. This parameter is only required if the calc_type parameter is set to "percentiles". If the calc_type parameter is set to mean, the percentiles property cannot be set.

Specify the body parameter in the following JSON format.

```
{
    "cycle": "string",
    "from": 0,
    "metric_category": "string",
    "metric_specs": {
        "name": "string",
        "key1": "string",
        "key2": "string",
        "calc_type": "string",
        "percentiles": []
    },
    "object_ids": [],
    "object_type": "string",
    "until": 0
}
```

`POST /metrics/total`

Specify the following parameters.

`body:` ***Object***

The description of the metric request.

`from:` ***Number***

The beginning timestamp for the request. Return only metrics collected after this time. Time is expressed in milliseconds since the epoch. 0 indicates the time of the request. A negative value is evaluated relative to the current time. The default unit for a negative value is milliseconds, but other units can be specified with a unit suffix. See the REST API Guide for supported time units and suffixes.

`until:` ***Number***

The ending timestamp for the request. Return only metrics collected before this time. Follows the same time value guidelines as the from parameter.

`cycle:` ***String***

The aggregation period for metrics.

The following values are valid:

- `auto`
- `1sec`
- `30sec`
- `5min`
- `1hr`
- `24hr`

`object_type:` ***String***

Indicates the object type of unique identifiers specified in the object_ids property.

The following values are valid:

- `network`
- `device`
- `application`
- `vlan`
- `device_group`
- `system`

`object_ids:` ***Array of Numbers***

The list of numeric values that represent unique identifiers. Unique identifiers can be retrieved through the /networks, /devices, /applications, /vlans, /devicegroups, /activitygroups, and /appliances resources. For system health metrics, specify the ID of the sensor or console and set the object_type parameter to "system".

`metric_category:` ***String***

The group of metrics that are searchable in the metric catalog.

`metric_specs:` ***Array of Objects***

An array of metric specification objects.

`name:` ***String***

The field name for the metric. When filtering in the metric catalog on a metric_category, each result is a potential metric_spec name. When a result is selected from the catalog, the "Metric" field value is a valid option for this field.

`key1:` ***String***

(Optional) Filter detail metrics. Detail metrics break down data through keys, which are strings or IP addresses. For example, the metric "HTTP Requests by Method" accepts a key1 value of "GET." Keys can also be regular expressions that are delimited with forward slashes ("/GET/").

`key2:` ***String***

(Optional) Enable additional filtering on detail metrics.

`calc_type:` ***String***

(Optional) The type of calculation to perform.

The following values are valid:

- `mean`
- `percentiles`

`percentiles:` ***Array of Numbers***

(Optional) The list of percentiles, sorted in ascending order, which should be returned. This parameter is only required if the calc_type parameter is set to "percentiles". If the calc_type parameter is set to mean, the percentiles property cannot be set.

Specify the body parameter in the following JSON format.

```
{
    "cycle": "string",
    "from": 0,
    "metric_category": "string",
    "metric_specs": {
        "name": "string",
        "key1": "string",
        "key2": "string",
        "calc_type": "string",
        "percentiles": []
    },
    "object_ids": [],
    "object_type": "string",
    "until": 0
}
```

`POST /metrics/totalbyobject`

Specify the following parameters.

`body:` ***Object***

The description of the metric request.

`from:` ***Number***

The beginning timestamp for the request. Return only metrics collected after this time. Time is expressed in milliseconds since the epoch. 0 indicates the time of the request. A negative value is evaluated relative to the current time. The default unit for a negative value is milliseconds, but other units can be specified with a unit suffix. See the REST API Guide for supported time units and suffixes.

`until:` ***Number***

The ending timestamp for the request. Return only metrics collected before this time. Follows the same time value guidelines as the from parameter.

`cycle:` ***String***

The aggregation period for metrics.

The following values are valid:

- `auto`
- `1sec`
- `30sec`
- `5min`
- `1hr`
- `24hr`

`object_type:` ***String***

Indicates the object type of unique identifiers specified in the object_ids property.

The following values are valid:

- `network`
- `device`
- `application`
- `vlan`
- `device_group`
- `system`

`object_ids:` ***Array of Numbers***

The list of numeric values that represent unique identifiers. Unique identifiers can be retrieved through the /networks, /devices, /applications, /vlans, /devicegroups, /activitygroups, and /appliances resources. For system health metrics, specify the ID of the sensor or console and set the object_type parameter to "system".

`metric_category:` ***String***

The group of metrics that are searchable in the metric catalog.

`metric_specs:` ***Array of Objects***

An array of metric specification objects.

`name:` ***String***

The field name for the metric. When filtering in the metric catalog on a metric_category, each result is a potential metric_spec name. When a result is selected from the catalog, the "Metric" field value is a valid option for this field.

`key1:` ***String***

(Optional) Filter detail metrics. Detail metrics break down data through keys, which are strings or IP addresses. For example, the metric "HTTP Requests by Method" accepts a key1 value of "GET." Keys can also be regular expressions that are delimited with forward slashes ("/GET/").

`key2:` ***String***

(Optional) Enable additional filtering on detail metrics.

`calc_type:` ***String***

(Optional) The type of calculation to perform.

The following values are valid:

- `mean`
- `percentiles`

`percentiles:` ***Array of Numbers***

(Optional) The list of percentiles, sorted in ascending order, which should be returned. This parameter is only required if the calc_type parameter is set to "percentiles". If the calc_type parameter is set to mean, the percentiles property cannot be set.

Specify the body parameter in the following JSON format.

```
{
    "cycle": "string",
    "from": 0,
    "metric_category": "string",
    "metric_specs": {
        "name": "string",
        "key1": "string",
        "key2": "string",
        "calc_type": "string",
        "percentiles": []
    },
    "object_ids": [],
    "object_type": "string",
    "until": 0
}
```

`GET /metrics/next/{xid}`

Specify the following parameters.

`xid:` ***Number***

The unique identifier returned by a metric query.

Supported time units

For most parameters, the default unit for time measurement is milliseconds. However, the following parameters return or accept alternative time units such as minutes and hours:

- Device
  - active_from
  - active_until
- Device group
  - active_from
  - active_until
- Metrics
  - from
  - until
- Record Log
  - from
  - until

- context_ttl

The following table displays supported time units:

Time unit	Unit suffix
Year	y
Month	M
Week	w
Day	d
Hour	h
Minute	m
Second	s
Millisecond	ms

To specify a time unit other than milliseconds for a parameter, append the unit suffix to the value. For example, to request devices active in the last 30 minutes, specify the following parameter value:

```
GET /api/v1/devices?active_from=-30m
```

The following example specifies a search for HTTP records created between 1 and 2 hours ago:

```
{
    "from": "-2h",
    "until": "-1h",
    "types": ["~http"]
}
```

Network locality entry

You can manage a list that specifies the network locality of IP addresses.

For example, you can create an entry in the network locality list that specifies that an IP address or CIDR block is internal or external.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /networklocalities	Retrieve all network locality entries.
POST /networklocalities	Create a network locality entry.
DELETE /networklocalities/{id}	Delete a network locality entry. **Note:** This operation is not available on sensors connected to Reveal(x) 360. However, this operation is available in the Reveal(x) 360 REST API.
GET /networklocalities/{id}	Retrieve a specific network locality entry.
PATCH /networklocalities/{id}	Apply updates to a specific network locality entry.

Operation	Description
	Note: This operation is not available on sensors connected to Reveal(x) 360. However, this operation is available in the Reveal(x) 360 REST API.

Operation details

`GET /networklocalities`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "description": "string",
    "external": true,
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`POST /networklocalities`

Specify the following parameters.

`body:` ***Object***

Apply the specified property values to the new network locality entry.

`name:` ***String***

(Optional) The name of the network locality. If this field is not specified, the network locality is named in the following format: "locality_ID", where ID is the unique identifier of the network locality.

`network:` ***String***

(Optional) Deprecated. Specify CIDR blocks or IP addresses with the networks field.

`networks:` ***Array of Strings***

(Optional) An array of CIDR blocks or IP addresses that define the network locality.

`external:` ***Boolean***

Indicates whether the network is internal or external.

`description:` ***String***

(Optional) An optional description of the network locality entry.

Specify the body parameter in the following JSON format.

```
{
    "description": "string",
    "external": true,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`GET /networklocalities/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the network locality entry.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "description": "string",
    "external": true,
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`DELETE /networklocalities/{id}`

Specify the following parameters.

`id:` ***Number***

The unique identifier for the network locality entry.

`PATCH /networklocalities/{id}`

Specify the following parameters.

`body:` ***Object***

Apply the specified property value updates to the network locality entry.

`network:` ***String***

(Optional) Deprecated. Specify CIDR blocks or IP addresses with the networks field.

`networks:` ***Array of Strings***

(Optional) An array of CIDR blocks or IP addresses that define the network locality.

`name:` ***String***

(Optional) The name of the network locality.

`external:` ***Boolean***

(Optional) Indicates whether the network is internal or external.

`description:` ***String***

(Optional) An optional description of the network locality entry.

Specify the body parameter in the following JSON format.

```
{
    "description": "string",
    "external": true,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`id:` ***Number***

The unique identifier for the network locality entry.

Node

A node is a sensor that is connected to a console.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /nodes	Retrieve all sensors connected to this console.
GET /nodes/{id}	Retrieve a specific sensor that is connected to this console.
PATCH /nodes/{id}	Update a specific sensor that is connected to this console.

Operation details

`GET /nodes`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "add_time": 0,
    "display_name": "string",
    "enabled": true,
    "firmware_version": "string",
    "hostname": "string",
    "id": 0,
    "license_status": "string",
    "nickname": "string",
    "ntp_sync": true,
    "product_key": "string",
    "status_code": "string",
    "status_message": "string",
    "time_added": 0,
    "time_offset": 0,
    "uuid": "string"
}
```

`GET /nodes/{id}`

Specify the following parameters.

`id:` ***Number***

The ID of the sensor.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "add_time": 0,
    "display_name": "string",
    "enabled": true,
    "firmware_version": "string",
    "hostname": "string",
    "id": 0,
    "license_status": "string",
    "nickname": "string",
    "ntp_sync": true,
    "product_key": "string",
    "status_code": "string",
    "status_message": "string",
    "time_added": 0,
    "time_offset": 0,
```

```
    "uuid": "string"
}
```

`PATCH /nodes/{id}`

Specify the following parameters.

`body:` ***Object***

Apply the specified updates to the Discover node.

`id:` ***Number***

The unique identifier for the Discover node.

Open Data Stream

An open data stream (ODS) is a channel through which you can send specified metric data from a sensor to an external, third-party system. For example, you might want to store or analyze metric data with a remote tool, such as Splunk, MongoDB, or Amazon Web Services (AWS).

Sending data through an open data stream is a two-step procedure. First, you configure a connection to the target system that will receive the data. Second, you write a trigger that specifies what data to send to the target system and when to send it. For more information, see Open Data Streams.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /odstargets	Retrieve all Open Data Stream targets.
GET /odstargets/http	Retrieve all HTTP Open Data Stream targets.
POST /odstargets/http	Create a new HTTP Open Data Stream target.
DELETE /odstargets/http/{name}	Delete an HTTP Open Data Stream target.
GET /odstargets/http/{name}	Retrieve a specific HTTP Open Data Stream target.
GET /odstargets/kafka	Retrieve all Kafka Open Data Stream targets.
POST /odstargets/kafka	Create a new Kafka Open Data Stream target.
DELETE /odstargets/kafka/{name}	Delete a Kafka Open Data Stream target.
GET /odstargets/kafka/{name}	Retrieve a specific Kafka Open Data Stream target.
GET /odstargets/mongodb	Retrieve all MongoDB Open Data Stream targets.
POST /odstargets/mongodb	Create a new MongoDB Open Data Stream target.
DELETE /odstargets/mongodb/{name}	Delete a MongoDB Open Data Stream target.
GET /odstargets/mongodb/{name}	Retrieve a specific MongoDB Open Data Stream target.
GET /odstargets/raw	Retrieve all Raw Open Data Stream targets.
POST /odstargets/raw	Create a new Raw Open Data Stream target.
DELETE /odstargets/raw/{name}	Delete a Raw Open Data Stream target.
GET /odstargets/raw/{name}	Retrieve a specific Raw Open Data Stream target.
GET /odstargets/syslog	Retrieve all Syslog Open Data Stream targets.
POST /odstargets/syslog	Create a new Syslog Open Data Stream target.

Operation	Description
DELETE /odstargets/syslog/{name}	Delete a Syslog Open Data Stream target.
GET /odstargets/syslog/{name}	Retrieve a specific Syslog Open Data Stream target.

Operation details

`GET /odstargets`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/http`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/http/{name}`

Specify the following parameters.

`name`: ***String***

The name of the target.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/kafka`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "brokers": [],
    "compression": "string",
    "name": "string",
    "partition_strategy": "string",
    "protocol": "string",
    "skip_cert_verification": true,
    "tls_ca_certs": "string",
    "tls_client_cert": "string",
    "tls_client_key": "string"
}
```

`GET /odstargets/kafka/{name}`

Specify the following parameters.

`name`: ***String***

The name of the target.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "brokers": [],
```

```
    "compression": "string",
    "name": "string",
    "partition_strategy": "string",
    "protocol": "string",
    "skip_cert_verification": true,
    "tls_ca_certs": "string",
    "tls_client_cert": "string",
    "tls_client_key": "string"
}
```

`GET /odstargets/mongodb`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/mongodb/{name}`

Specify the following parameters.

`name` : ***String***

The name of the target.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/raw`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/raw/{name}`

Specify the following parameters.

`name` : ***String***

The name of the target.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{}
```

`GET /odstargets/syslog`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "batch_min_bytes": 0,
    "concurrent_connections": 0,
    "host": "string",
    "localtime": true,
    "name": "string",
    "port": 0,
    "protocol": "string",
    "skip_cert_verification": true,
    "tcp_length_prefix_framing": true,
    "tls_ca_certs": "string",
```

```
    "tls_client_cert": "string",
    "tls_client_key": "string"
}
```

`GET /odstargets/syslog/{name}`

Specify the following parameters.

`name:` ***String***

The name of the target.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "batch_min_bytes": 0,
    "concurrent_connections": 0,
    "host": "string",
    "localtime": true,
    "name": "string",
    "port": 0,
    "protocol": "string",
    "skip_cert_verification": true,
    "tcp_length_prefix_framing": true,
    "tls_ca_certs": "string",
    "tls_client_cert": "string",
    "tls_client_key": "string"
}
```

`POST /odstargets/http`

Specify the following parameters.

`body:` ***Object***

`name:` ***String***

The name for the target.

`host:` ***String***

The hostname or IP address of the remote HTTP server.

`port:` ***Number***

The TCP port number of the HTTP server.

`protocol:` ***String***

The protocol to transmit data over.

The following values are valid:

- `http`
- `https`

`skip_cert_verification:` ***Boolean***

(Optional) Indicates whether to bypass TLS certificate verification for encrypted data. This parameter is valid only if `protocol` is set to `https`.

`pipeline:` ***Boolean***

Indicates whether multiple concurrent HTTP connections are enabled, which can improve throughput speed.

`additional_header:` ***String***

(Optional) Specifies an additional HTTP header to include in each request. Headers must be specified in the following format: "<key>:<value>". For example: "additional_header": "Accept: text/html".

`authentication:` ***Object***

An object that contains HTTP authentication credentials.

`auth_type:` ***String***

The type of HTTP authentication.

The following values are valid:

- `none`
- `basic`
- `aws`
- `azure_storage`
- `azure_ad`
- `crowdstrike`

`username:` ***String***

(Optional) The name of the user. This option is required if `auth_type` is set to `basic` or if `auth_type` is set to `azure_ad` and `grant_type` is set to `resource_owner`.

`password:` ***String***

(Optional) The password of the user. This option is required if `auth_type` is set to `basic` or if `auth_type` is set to `azure_ad` and `grant_type` is set to `resource_owner`.

`access_key:` ***String***

(Optional) The access key ID. This option is required for AWS and Azure Storage authentication.

`secret_key:` ***String***

(Optional) The secret access key. This option is required for AWS authentication.

`service:` ***String***

(Optional) The service code of the AWS service, such as "AmazonEC2". This option is required for AWS authentication.

`region:` ***String***

(Optional) The name of the AWS region, such as "us-west-1". This option is required for AWS authentication.

`grant_type:` ***String***

(Optional) The OAuth 2.0 grant type. This option is required for Azure AD authentication.

The following values are valid:

- `client`
- `resource_owner`

`client_id:` ***String***

(Optional) The client ID. This option is required for Azure AD and Crowdstrike authentication.

`client_secret:` ***String***

(Optional) The client Secret Key. This option is required for Azure AD and Crowdstrike authentication.

`resource:` ***String***

(Optional) The Azure AD resource URI. This option is required for Azure AD authentication.

`token_endpoint:` ***String***

(Optional) The Azure AD /token endpoint. For example: "https://login.microsoftonline.com/<tenant_id>/oauth2/token". This option is required for Azure AD authentication.

Specify the body parameter in the following JSON format.

```
{
    "additional_header": "string",
    "authentication": {
        "auth_type": "string",
        "username": "string",
        "password": "string",
        "access_key": "string",
        "secret_key": "string",
        "service": "string",
        "region": "string",
        "grant_type": "string",
        "client_id": "string",
        "client_secret": "string",
        "resource": "string",
        "token_endpoint": "string"
    },
    "host": "string",
    "name": "string",
    "pipeline": true,
    "port": 0,
    "protocol": "string",
    "skip_cert_verification": true
}
```

`POST /odstargets/kafka`

Specify the following parameters.

`body:` ***Object***

`name:` ***String***

The name for the target.

`brokers:` ***Array of Objects***

An array of one or more objects that contain information about Kafka Brokers.

`host:` ***String***

The hostname or IP address of the remote Kafka broker.

`port:` ***Number***

The TCP port number of the Kafka broker.

`compression:` ***String***

(Optional) The compression method to apply to transmitted data.

The following values are valid:

- `none`
- `gzip`
- `snappy`

`partition_strategy:` ***String***

(Optional) The partitioning method to apply to transmitted data.

The following values are valid:

- `hash_key`
- `manual`
- `random`
- `round_robin`

`protocol:` ***String***

The protocol to transmit data over.

The following values are valid:

- `tcp`
- `tls`

`tls_client_cert:` ***String***

(Optional) The TLS client certificate that is sent to the Kafka server during the TLS handshake. Specify this option if client authentication is enabled on the Kafka server.

`tls_client_key:` ***String***

(Optional) The private key of the TLS client certificate specified by the tls_client_cert parameter. Specify this option if client authentication is enabled on the Kafka server.

`skip_cert_verification:` ***Boolean***

(Optional) Indicates whether to bypass TLS certificate verification for encrypted data. This parameter is valid only if protocol is set to tls.

`tls_ca_certs:` ***String***

(Optional) The trusted certificates to validate the Kafka server certificate with, in PEM format. Specify this option if your Kafka server certificate has not been signed by a valid Certificate Authority (CA). If this option is not specified, the server certificate is validated with the built-in list of valid CA certificates. This option is valid only if the protocol is TLS.

`authentication:` ***Object***

(Optional) An object that contains Kafka authentication credentials.

`auth_type:` ***String***

The type of SASL authentication.

The following values are valid:

- `scram`

`username:` ***String***

The username of the SASL user.

`password:` ***String***

The password of the SASL user.

`algorithm:` ***String***

The hashing algorithm for SASL authentication.

The following values are valid:

- `sha256`
- `sha512`

Specify the body parameter in the following JSON format.

```
{
    "authentication": {
        "auth_type": "string",
```

```
        "username": "string",
        "password": "string",
        "algorithm": "string"
    },
    "brokers": {
        "host": "string",
        "port": 0
    },
    "compression": "string",
    "name": "string",
    "partition_strategy": "string",
    "protocol": "string",
    "skip_cert_verification": true,
    "tls_ca_certs": "string",
    "tls_client_cert": "string",
    "tls_client_key": "string"
}
```

`POST /odstargets/mongodb`

Specify the following parameters.

`body:` ***Object***

`name:` ***String***

The name for the target.

`host:` ***String***

The hostname or IP address of the remote MongoDB server.

`port:` ***Number***

The TCP port number of the MongoDB server.

`encrypt:` ***Boolean***

(Optional) Indicates whether data is encrypted with TLS.

`skip_cert_verification:` ***Boolean***

(Optional) Indicates whether to bypass TLS certificate verification for encrypted data. This parameter is valid only if `encrypt` is set to `true`.

`authentication:` ***Array of Objects***

(Optional) An array of objects that contain MongoDB authentication credentials.

`database:` ***String***

The name of the MongoDB database.

`user:` ***String***

The name of the user that has permission to modify the specified database.

`password:` ***String***

The password of the user.

Specify the body parameter in the following JSON format.

```
{
    "authentication": {
        "database": "string",
        "user": "string",
        "password": "string"
    },
    "encrypt": true,
    "host": "string",
    "name": "string",
    "port": 0,
```

```
    "skip_cert_verification": true
}
```

`POST /odstargets/raw`

Specify the following parameters.

`body:` ***Object***

`name:` ***String***

The name for the target.

`host:` ***String***

The hostname or IP address of the remote server.

`port:` ***Number***

The TCP or UDP port number of the remote server.

`protocol:` ***String***

The protocol to transmit data over.

The following values are valid:

- `tcp`
- `udp`

`compression:` ***Boolean***

(Optional) Indicates whether gzip compression is applied to transmitted data.

`gzip_threshold_bytes:` ***Number***

(Optional) The number of bytes that specifies the threshold for creating a new message. Every 30 seconds, the sensor or console sends messages that exceed the specified size to prevent messages from growing too large. This option is valid only if `compression` is set to `true`.

`gzip_threshold_seconds:` ***Number***

(Optional) The number of seconds that specifies the threshold for creating a new message. Every 30 seconds, the sensor or console sends messages that have been written for more than the specified time period to prevent messages from growing too large. This option is valid only if `compression` is set to `true`.

Specify the body parameter in the following JSON format.

```
{
    "compression": true,
    "gzip_threshold_bytes": 0,
    "gzip_threshold_seconds": 0,
    "host": "string",
    "name": "string",
    "port": 0,
    "protocol": "string"
}
```

`POST /odstargets/syslog`

Specify the following parameters.

`body:` ***Object***

`name:` ***String***

The name for the target.

`host:` ***String***

The hostname or IP address of the remote Syslog server.

`port:` ***Number***

The TCP or UDP port number of the remote Syslog server.

`tcp_length_prefix_framing:` ***Boolean***

(Optional) Indicates whether to prepend the number of bytes in a message to the beginning of the message. If this parameter is set to false, the end of each message is delimited by a trailing newline.

`batch_min_bytes:` ***Number***

(Optional) The minimum number of bytes to send at a time to the syslog server.

`concurrent_connections:` ***Number***

(Optional) The number of concurrent connections to send messages over.

`localtime:` ***Boolean***

(Optional) Indicates whether timestamps reference the local time zone of the sensor or console. If this parameter is set to false, timestamps reference GMT.

`protocol:` ***String***

The protocol to transmit data over.

The following values are valid:

- `tcp`
- `udp`
- `tls`

`tls_client_cert:` ***String***

(Optional) The TLS client certificate that is sent to the Syslog server during the TLS handshake. Specify this option if client authentication is enabled on the Syslog server.

`tls_client_key:` ***String***

(Optional) The private key of the TLS client certificate specified by the tls_client_cert parameter. Specify this option if client authentication is enabled on the Syslog server.

`skip_cert_verification:` ***Boolean***

(Optional) Indicates whether to bypass TLS certificate verification for encrypted data. This parameter is valid only if protocol is set to tls.

`tls_ca_certs:` ***String***

(Optional) The trusted certificates to validate the Syslog server certificate with, in PEM format. Specify this option if your Syslog server certificate has not been signed by a valid Certificate Authority (CA). If this option is not specified, the server certificate is validated with the built-in list of valid CA certificates. This option is valid only if the protocol is TLS and skip_cert_verification is false.

Specify the body parameter in the following JSON format.

```
{
    "batch_min_bytes": 0,
    "concurrent_connections": 0,
    "host": "string",
    "localtime": true,
    "name": "string",
    "port": 0,
    "protocol": "string",
    "skip_cert_verification": true,
    "tcp_length_prefix_framing": true,
    "tls_ca_certs": "string",
    "tls_client_cert": "string",
    "tls_client_key": "string"
}
```

`DELETE /odstargets/http/{name}`

Specify the following parameters.

`name:` ***String***
The name of the target.

`DELETE /odstargets/kafka/{name}`

Specify the following parameters.

`name:` ***String***
The name of the target.

`DELETE /odstargets/mongodb/{name}`

Specify the following parameters.

`name:` ***String***
The name of the target.

`DELETE /odstargets/raw/{name}`

Specify the following parameters.

`name:` ***String***
The name of the target.

`DELETE /odstargets/syslog/{name}`

Specify the following parameters.

`name:` ***String***
The name of the target.

Pairing

This resource enables you to generate a token required to connect a sensor to a console.

The following table displays all of the operations you can perform on this resource:

Operation	Description
POST /pairing/token	Generate a token required to connect the sensor to a console.

Operation details

`POST /pairing/token`

There are no parameters for this operation.

Record Log

Records are structured flow and transaction information about events on your network.

After you connect the ExtraHop system to a record store, you can generate and send record information to the recordstore, and you can query records to retrieve information about any object on your network. For more information, see Query for records through the REST API.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /records/cursor/{cursor}	Deprecated. Replaced by `POST /records/cursor`.
POST /records/cursor	Retrieve records starting at a specified cursor.
POST /records/search	Perform a record log query.

Operation details

`POST /records/search`

Specify the following parameters.

`body:` ***Object***

The record log query.

`from:` ***Number***

The beginning timestamp of the time range the query will search, expressed in milliseconds since the epoch. A negative value specifies that the search will begin with records created at a time in the past. For example, specify -600000ms to begin the search with records created 10 minutes before the time of the request. Negative values can be specified with a time unit other than milliseconds, such as seconds or hours. See the REST API Guide for supported time units and suffixes.

`until:` ***Number***

The ending timestamp of the time range the query will search, expressed in milliseconds since the epoch. A 0 value specifies that the search will end with records created at the time of the request. A negative value specifies that the search will end with records created at a time in the past. For example, specify -300000ms to end the search with records created 5 minutes before the time of the request. Negative values can be specified with a time unit other than milliseconds, such as seconds or hours. See the REST API Guide for supported time units and suffixes.

`types:` ***Array of Strings***

(Optional) An array of one or more record formats. The query returns only records that match the specified formats. If no value is specified, the query returns records of any type. Valid values for this field are displayed in the Record Type field on the Record Formats page. For example: "~cifs".

`limit:` ***Number***

The maximum number of records returned by the query. The maximum value cannot exceed 10000. The default value is 100.

`offset:` ***Number***

The number of records to skip in the query results. The query will return records starting from the offset value. This parameter is often combined with the limit and sort parameters. The default value is 0. For ExtraHop recordstores, the maximum value is 10,000; to retrieve records returned after the first 10,000, see POST /records/cursor/. For third-party recordstores, there is no maximum value.

`sort:` ***Array of Objects***

The list of one or more sort objects that specify sort priorities. The returned records are sorted in the order the objects are listed. The parameters are defined under the sort_item

section below. If no sort_item values are provided, records are sorted by timestamp in descending order.

`field:` ***String***

The field name that returned records are sorted by.

`direction:` ***String***

The order in which returned records are sorted. The default order is descending. After all other sorting criteria are applied, or if no sorting criteria was specified, the default order is descending by timestamp.

The following values are valid:

- `asc`
- `desc`

`filter:` ***Object***

The object containing the parameters that specify the filter criteria. The parameters are defined under the filter section below. If no filter values are provided, the query returns all records that match the time range and any specified record formats.

`field:` ***String***

The name of the field in the record to be filtered. The query compares the contents of the field parameter to the value of the operand parameter. If the specified field name is ".any", the union of all field values will be searched. If the specified field name is ".ipaddr" or ".port", the client, server, sender, and receiver roles are included in the search. Field names are located in record formats that can be viewed in the ExtraHop system.

`operator:` ***String***

The compare method applied when matching the operand value against the field contents. All filter objects require an operator.

The following values are valid:

- `>`
- `<`
- `<=`
- `>=`
- `=`
- `!=`
- `startswith`
- `~`
- `!~`
- `and`
- `or`
- `not`
- `exists`
- `not_exists`
- `in`
- `not_in`

`operand:` ***String or Number or Object***

The value that the query attempts to match. The query compares the value of the operand to the contents of the field parameter and applies the compare method specified by the operator parameter. You can explicitly specify the operand data type as described in the REST API Guide.

`rules:` ***Array of Objects***

The list of one or more filter objects within a single filter object. Filter objects can be embedded recursively. Only "and", "or", and "not" operators are allowed for this parameter.

`context_ttl:` ***Number***

The amount of time to keep the search context active. The specified value is interpreted as a duration into the future. The default unit is milliseconds, but other units can be specified with a unit suffix. See the REST API Guide for supported time units and suffixes. If a non-null value is specified, the response includes a cursor ID that is accepted by POST /records/cursor/. This parameter is not supported for third-party recordstores.

Specify the body parameter in the following JSON format.

```
{
    "context_ttl": 0,
    "filter": {
        "field": "string",
        "operator": "string",
        "operand": "string",
        "rules": []
    },
    "from": 0,
    "limit": 0,
    "offset": 0,
    "sort": {
        "field": "string",
        "direction": "string"
    },
    "types": [],
    "until": 0
}
```

`POST /records/cursor`

Specify the following parameters.

`body:` ***Object***

The cursor ID that specifies the next page of results in the query.

`cursor:` ***String***

The unique identifier of the cursor that specifies the next page of results in the query.

Specify the body parameter in the following JSON format.

```
{
    "cursor": "string"
}
```

`context_ttl:` ***Number***

(Optional) The amount of time to keep the search context active, expressed in milliseconds.

`GET /records/cursor/{cursor}`

Specify the following parameters.

`cursor:` ***String***

The cursor ID.

`context_ttl:` ***Number***

(Optional) The amount of time to keep the search context active, expressed in milliseconds.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "cursor": "string",
    "from": 0,
    "records": {},
    "total": 0,
    "until": 0,
    "warnings": {}
}
```

Operand values in record queries

The `operand` field in the `POST /records/search` method specifies the value that a record query attempts to match. You can specify either the value only or both the data type and the value. If you specify only the value, the query will refer to the record format associated with the `field` parameter to determine the data type of the value.

For example, if you want to search for an IP address, you can specify an IP address data type, and then provide the actual address as the value.

The following example explicitly specifies the data type and value of the operand:

```
{
    "from": -1000,
    "filter": {
        "field" : "senderAddr",
        "operator": "=",
        "operand" : { "type" : "ipaddr4", "value": "1.2.3.4" }
    }
}
```

The following example specifies only the value of the operand:

```
{
    "from": -1000,
    "filter": {
        "field" : "senderAddr",
        "operator": "=",
        "operand" : "1.2.3.4"
    }
}
```

You can explicitly specify the following data types in the `operand` field:

- application
- boolean
- device

  Note: You must specify the discovery ID of the device in the value field. You can find the discovery ID of a device through the `POST /devices/search` operation.
- device_filter
- device_group
- flowinterface
- flownetwork
- ipaddr4
- ipaddr6
- number
- network_locality
- object

- string

The `operand` field supports CIDR notation when filtering by IP addresses; the `operator` field must be set to "=" or "!=".

You can specify multiple filters by including the `rules` option, as shown in the following example:

```
{
  "filter": {
    "operator": "and",
    "rules": [
      {
        "field": "method",
        "operand": "SMB2_READ",
        "operator": "="
      },
      {
        "field": "reqL2Bytes",
        "operand": "100",
        "operator": ">"
      }
    ]
  },
  "types": [
    "~cifs"
  ],
    "from": "-30m"
}
```

Query records with a device group filter

To filter records by device group in the REST API, you must send a `POST` request to the `/records/search` endpoint with a record query filter that meets the following criteria:

- The `field` must specify devices, such as `client`, `server`, `sender`, or `receiver`.
- The `operator` must be either `in` or `not_in`.
- The `operandtype` must be `device_group`.
- The `operandvalue` must be a string representation of the numerical device group ID. You can retrieve device group IDs by running the GET /devicegroup operation and viewing the contents of the `id` field in the response.

For example, the following query searches for records in which the client device was a member of a device group with an ID of 200:

```
{
    "from": "-30m",
    "filter": {
        "field": "client",
        "operator": "in",
        "operand": {
            "type": "device_group",
            "value": "200"
        }
    }
}
```

You can also filter records by device group criteria without creating a device group by specifying the operand type as `device_filter`. For example, the following query searches for records in which the client device is running Windows 10:

```
{
    "from": "-30m",
```

```
    "filter": {
        "field": "client",
        "operator": "in",
        "operand": {
            "type": "device_filter",
            "value": {
                "field": "software",
                "operand": "windows_10",
                "operator": "="
            }
        }
    }
}
```

Note: Operand values with type `device_filter` for record search are formatted the same as device search filters. For more information, see Operand values for device groups.

Query records with a network locality filter

To filter records by device group in the REST API, you must send a POST request to the `/records/search` endpoint with a record query filter that meets the following criteria:

- The field must be a record field that specifies an IP address such as `clientAddr`, `serverAddr`, `senderAddr`, or `receiverAddr`.
- The operator must be either `in` or `not_in`.
- The operand type must be `network_locality`.
- The operand value must be a string representation of a numerical network locality ID. You can view locality IDs with the `GET /networklocalities` operation.

For example, the following query searches for records where the client device is in a network locality with an ID of `123`:

```
{
    "from": "-30m",
    "filter": {
        "field": "clientAddr",
        "operand": {
            "type": "network_locality",
            "value": "123"
        },
        "operator": "in"
    }
}
```

Supported time units

For most parameters, the default unit for time measurement is milliseconds. However, the following parameters return or accept alternative time units such as minutes and hours:

- Device
  - active_from
  - active_until
- Device group
  - active_from
  - active_until
- Metrics
  - from
  - until

- Record Log
  - from
  - until
  - context_ttl

The following table displays supported time units:

Time unit	Unit suffix
Year	y
Month	M
Week	w
Day	d
Hour	h
Minute	m
Second	s
Millisecond	ms

To specify a time unit other than milliseconds for a parameter, append the unit suffix to the value. For example, to request devices active in the last 30 minutes, specify the following parameter value:

```
GET /api/v1/devices?active_from=-30m
```

The following example specifies a search for HTTP records created between 1 and 2 hours ago:

```
{
    "from": "-2h",
    "until": "-1h",
    "types": ["~http"]
}
```

Running config

The running configuration file is a JSON document that contains core system configuration information for the ExtraHop system.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /runningconfig	Retrieve the current running configuration file.
PUT /runningconfig	Replace the current running configuration file. Configuration file changes are not automatically saved.
POST /runningconfig/save	Save the current changes to the running configuration file.
GET /runningconfig/saved	Retrieve the saved running configuration file.

Operation details

`GET /runningconfig/saved`

There are no parameters for this operation.

`POST /runningconfig/save`

There are no parameters for this operation.

`GET /runningconfig`

Specify the following parameters.

`section:` ***String***
(Optional) (Optional) The specific section of the running configuration file that you want to retrieve.

`PUT /runningconfig`

Specify the following parameters.

`body:` ***String***
(Optional) The running configuration file.

SSL decrypt key

This resource enables you to add a decryption key for your network traffic.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /ssldecryptkeys	Retrieve all SSL decryption keys.
POST /ssldecryptkeys	Create a new SSL decryption key.
DELETE /ssldecryptkeys/{id}	Remove an SSL key from the ExtraHop system.
GET /ssldecryptkeys/{id}	Retrieve an SSL PEM and metadata.
PATCH /ssldecryptkeys/{id}	Update an existing SSL decryption key.
GET /ssldecryptkeys/{id}/protocols	Retrieve all protocols assigned to an SSL decryption key.
POST /ssldecryptkeys/{id}/protocols	Create a new protocol for an SSL decryption key.
DELETE /ssldecryptkeys/{id}/protocols/{protocol}	Delete a protocol from an SSL decryption key.

Operation details

`GET /ssldecryptkeys`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "cert_pem": "string",
    "enabled": true,
    "id": "string",
```

```
    "name": "string"
}
```

`POST /ssldecryptkeys`

Specify the following parameters.

`body:` ***Object***

Set the specified property values on the new SSL decryption key.

`enabled:` ***Boolean***

Indicate whether this SSL decryption key is active.

`name:` ***String***

The friendly name for the SSL decryption key.

`certificate:` ***String***

The SSL certificate associated with this decryption key.

`private_key:` ***String***

The SSL private key that decrypts traffic.

Specify the body parameter in the following JSON format.

```
{
    "certificate": "string",
    "enabled": true,
    "name": "string",
    "private_key": "string"
}
```

`PATCH /ssldecryptkeys/{id}`

Specify the following parameters.

`body:` ***Object***

Apply the specified property updates to the SSL decryption key.

`id:` ***String***

The hexadecimal representation of the SHA-1 hash of the SSL decryption key. The string must not include delimiters.

`GET /ssldecryptkeys/{id}`

Specify the following parameters.

`id:` ***String***

The hexadecimal representation of the SHA-1 hash of the SSL decryption key. The string must not include delimiters.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "cert_pem": "string",
    "enabled": true,
    "id": "string",
    "name": "string"
}
```

```
DELETE /ssldecryptkeys/{id}
```

Specify the following parameters.

`id:` ***String***
The hexadecimal representation of the SHA-1 hash of the SSL decryption key. The string must not include delimiters.

```
GET /ssldecryptkeys/{id}/protocols
```

Specify the following parameters.

`id:` ***String***
The hexadecimal representation of the SHA-1 hash of the SSL decryption key. The string must not include delimiters.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "port": 0,
    "protocol": "string"
}
```

```
POST /ssldecryptkeys/{id}/protocols
```

Specify the following parameters.

`body:` ***Object***
The body of the protocol.
`protocol:` ***String***
The name of the protocol, in lowercase.
`port:` ***Number***
The port in which to listen for traffic.

Specify the body parameter in the following JSON format.

```
{
    "port": 0,
    "protocol": "string"
}
```

`id:` ***String***
The unique identifier for the SSL decrypt key.

```
DELETE /ssldecryptkeys/{id}/protocols/{protocol}
```

Specify the following parameters.

`protocol:` ***String***
The name of the protocol, in lowercase.
`id:` ***String***
The hexadecimal representation of the SHA-1 hash of the SSL decryption key. The string must not include delimiters.
`port:` ***Number***
(Optional) Remove only the protocols that are assigned on this port.

Support pack

A support pack is a file that contains configuration adjustments provided by ExtraHop Support.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /supportpacks	Retrieve metadata about all support packs.
POST /supportpacks	Upload and run a support pack.
POST /supportpacks/execute	Run a new support pack.
GET /supportpacks/queue/{id}	Check on the status of an in-progress, running support pack.
GET /supportpacks/{filename}	Downlolad an existing support pack by filename.

Operation details

`GET /supportpacks/queue/{id}`

Specify the following parameters.

`id:` ***String***
The unique identifier for the running support pack.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "created_time": 0,
    "filename": "string",
    "size": "string"
}
```

`GET /supportpacks/{filename}`

Specify the following parameters.

`filename:` ***String***
The name of the support pack to download.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "created_time": 0,
    "filename": "string",
    "size": "string"
}
```

`POST /supportpacks/execute`

`GET /supportpacks`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
```

```
    "created_time": 0,
    "filename": "string",
    "size": "string"
}
```

`POST /supportpacks`

Specify the following parameters.

`file:` ***Filename***

The filename for the support pack.

Tag

Device tags enable you to associate a device or group of devices by some characteristic.

For example, you might tag all of your HTTP servers or tag all of the devices that are in a common subnet. For more information, see Tag a device through the REST API.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /tags	Retrieve all tags.
POST /tags	Create a a new tag.
DELETE /tags/{id}	Delete a specific tag.
GET /tags/{id}	Retrieve a specific tag.
PATCH /tags/{id}	Apply updates to a specific tag.
GET /tags/{id}/devices	Retrieve all devices that are assigned to a specific tag.
POST /tags/{id}/devices	Assign and unassign a specific tag to devices.
DELETE /tags/{id}/devices/{child-id}	Unassign a device from a specific tag.
POST /tags/{id}/devices/{child-id}	Assign a device to a specific tag.

Operation details

`GET /tags`

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "id": 0,
    "mod_time": 0,
    "name": "string"
}
```

`POST /tags`

Specify the following parameters.

`body:` ***Object***

Apply the specified property values to the new tag.

`name:` ***String***
The string value for the tag.

Specify the body parameter in the following JSON format.

```
{
    "name": "string"
}
```

`GET /tags/{id}`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the tag.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "id": 0,
    "mod_time": 0,
    "name": "string"
}
```

`DELETE /tags/{id}`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the tag.

`PATCH /tags/{id}`

Specify the following parameters.

`body:` ***Object***
Apply the specified property value updates to the tag.
`id:` ***Number***
The unique identifier for the tag.

`GET /tags/{id}/devices`

Specify the following parameters.

`id:` ***Number***
The unique identifier for the tag.

`POST /tags/{id}/devices`

Specify the following parameters.

`body:` ***Object***
Lists of unique identifies for device to assign and unassign.
`assign:` ***Array of Numbers***
IDs of resources to assign

`unassign:` ***Array of Numbers***

IDs of resources to unassign

Specify the body parameter in the following JSON format.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Number***

The unique identifier for the tag.

`POST /tags/{id}/devices/{child-id}`

Specify the following parameters.

`child-id:` ***Number***

The unique identifier for the device.

`id:` ***Number***

the unique identifier for the tag.

`DELETE /tags/{id}/devices/{child-id}`

Specify the following parameters.

`child-id:` ***Number***

The unique identifier for the device.

`id:` ***Number***

The unique identifier for the tag.

Threat Collection

The Threat Collection resource enables you to upload free and commercial threat collections offered by the security community to your Reveal(x) system.

- You must upload threat collections individually to your Command appliance or Reveal(x) 360, and to all connected sensors.
- Custom threat collections must be formatted in Structured Threat Information eXpression (STIX) as TAR.GZ files. Reveal(x) currently supports STIX version 1.0 - 1.2.
- You can directly upload threat collections to Reveal(x) 360 systems for self-managed sensors. Contact ExtraHop Support to upload a threat collection to ExtraHop-managed sensors.
- The maximum number of observables that a threat collection can contain depends on your platform and license. Contact your ExtraHop representative for more information.

Note: This topic applies only to ExtraHop Reveal(x) Premium and Ultra.

For information about uploading STIX files through the ExtraHop system, see Upload STIX files through the REST API.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /threatcollections	Retrieve all threat collections.
POST /threatcollections	Create a new threat collection.

Operation	Description
DELETE /threatcollections/{id}	Delete a threat collection.
PUT /threatcollections/{id}	Upload a new threat collection. ExtraHop currently supports STIX versions 1.0 - 1.2. **Note:** If a threat collection with the same name already exists on the ExtraHop system, the existing threat collection is overwritten.
GET /threatcollections/{id}/observables	Retrieve the number of STIX observables loaded from a threat collection, such as IP address, hostname, or URI.

Operation details

`GET /threatcollections`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "id": 0,
    "last_updated": 0,
    "name": "string",
    "observables": 0,
    "user_key": "string"
}
```

`POST /threatcollections`

Specify the following parameters.

`user_key:` ***String***
(Optional) The user-supplied identifier for the threat collection. If this parameter is not specified, the threat collection name is set for this value, without spaces or punctuation.

`name:` ***String***
The name for the threat collection.

`file:` ***Filename***
The filename for the threat collection.

`PUT /threatcollections/~{userKey}`

Specify the following parameters.

`userKey:` ***String***
The user-supplied identifier for the threat collection.

`name:` ***String***
(Optional) The name for the threat collection.

`file:` ***Filename***
(Optional) The filename for the threat collection.

`DELETE /threatcollections/{id}`

Specify the following parameters.

`id:` ***String***
The unique identifier for the threat collection.

`GET /threatcollections/{id}/observables`

Specify the following parameters.

`id:` ***String***
The unique identifier for the threat collection.

User group

The user group resource enables you to manage and update groups of users and their dashboard sharing associations.

The following table displays all of the operations you can perform on this resource:

Operation	Description
GET /usergroups	Retrieve all user groups.
POST /usergroups	Create a new user group.
POST /usergroups/refresh	Query LDAP for the most recent user memberships for all remote user groups.
DELETE /usergroups/{id}	Delete a specific user group.
GET /usergroups/{id}	Retrieve a specific user group.
PATCH /usergroups/{id}	Update a specific user group.
DELETE /usergroups/{id}/associations	Delete all dashboard sharing associations with a specific user group.
GET /usergroups/{id}/members	Retrieve all members of a specific user group.
PATCH /usergroups/{id}/members	Assign or unassign users from a user group.
PUT /usergroups/{id}/members	Replace user group assignments.
POST /usergroups/{id}/refresh	Query LDAP for the most recent user membership of a specific remote user group.

Operation details

`GET /usergroups`

There are no parameters for this operation.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "display_name": "string",
    "enabled": true,
    "id": "string",
    "is_remote": true,
    "last_sync_time": 0,
    "name": "string",
    "rights": []
}
```

`POST /usergroups`

Specify the following parameters.

`body:` ***Object***

The properties of the user group.

`name:` ***String***

The name for the user group.

`enabled:` ***Boolean***

Indicates whether the user group is enabled.

Specify the body parameter in the following JSON format.

```
{
    "enabled": true,
    "name": "string"
}
```

`POST /usergroups/refresh`

There are no parameters for this operation.

`PATCH /usergroups/{id}`

Specify the following parameters.

`body:` ***Object***

The property value updates for the specific user group.

`id:` ***String***

The unique identifier for the user group.

`GET /usergroups/{id}`

Specify the following parameters.

`id:` ***String***

The unique identifier for the user group.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "display_name": "string",
    "enabled": true,
    "id": "string",
    "is_remote": true,
    "last_sync_time": 0,
    "name": "string",
    "rights": []
}
```

`DELETE /usergroups/{id}`

Specify the following parameters.

`id:` ***String***

The unique identifier for the user group.

`DELETE /usergroups/{id}/associations`

Specify the following parameters.

`id:` ***String***
The unique identifier for the user group.

`POST /usergroups/{id}/refresh`

Specify the following parameters.

`id:` ***String***
The unique identifier for the user group.

`GET /usergroups/{id}/members`

Specify the following parameters.

`id:` ***String***
The unique identifier for the user group.

If the request is successful, the ExtraHop system returns an object in the following format.

```
{
    "users": {}
}
```

`PATCH /usergroups/{id}/members`

Specify the following parameters.

`id:` ***String***
The unique identifier for the user group.

`body:` ***String***
An object that specifies which users to asssign or unassign. Each key must be a username and each value must be either "member" or null. For example {"Alice": "member", "Bob": null} assigns Alice to the group and unassigns Bob from the group.

`PUT /usergroups/{id}/members`

Specify the following parameters.

`id:` ***String***
The unique identifier for the user group.

`body:` ***String***
An object that specifies which users are assigned to the group. Each key must be a username and each value must be "member". For example {"Alice": "member", "Bob": "member"} assigns Alice and Bob as the only members of the group.