ThreatQuotient

A Securonix Company

Extrahop Connector

Version 1.3.2 rev-a

July 10, 2025

ThreatQuotient
20130 Lakeview Center Plaza Suite 400
Ashburn, VA 20147

ThreatQ Supported

Support
Email: support@threatq.com
Web: support.threatq.com
Phone: 703.574.9893

Contents

Warning and Disclaimer

ThreatQuotient, Inc. provides this document “as is”, without representation or warranty of any kind, express or implied, including without limitation any warranty concerning the accuracy, adequacy, or completeness of such information contained herein. ThreatQuotient, Inc. does not assume responsibility for the use or inability to use the software product as a result of providing this information.

Support

This integration is designated as **ThreatQ Supported**.

Support Email: support@threatq.com
Support Web: https://support.threatq.com
Support Phone: 703.574.9893

Integrations/apps/add-ons designated as **ThreatQ Supported** are fully supported by ThreatQuotient's Customer Support team.

ThreatQuotient strives to ensure all ThreatQ Supported integrations will work with the current version of ThreatQuotient software at the time of initial publishing. This applies for both Hosted instance and Non-Hosted instance customers.

> ThreatQuotient does not provide support or maintenance for integrations, apps, or add-ons published by any party other than ThreatQuotient, including third-party developers.

Integration Details

ThreatQuotient provides the following details for this integration:

Current Integration Version	1.3.2
Compatible with ThreatQ Versions	>= 5.6.0
Third-Party Application Hosting Type	Cloud, On-Prem
Python Version	3.6
Support Tier	ThreatQ Supported

Introduction

The Extrahop Connector allows a user to export indicators directly to Extrahop via Extrahop's REST API.

The indicator types pushed to Extrahop are:

- FQDN
- URL
- IP Address
- IPv6 Address

The Extrahop Connector creates a new Extrahop Collection whose name is the same as the configured ThreatQ data collection. Every time the connector runs, this Extrahop Collection is overwritten.

Prerequisites

Review the following requirements before attempting to install the connector.

Extrahop Credentials

The following credentials are required based on your Extrahop hosting method:

- Reveal(x) 360
  - Extrahop Reveal(x) 360 Hostname
  - Extrahop Reveal(x) 360 Client ID
  - Extrahop Reveal(x) 360 Client Secret
- Reveal(x) Enterprise
  - Extrahop Reveal(x) Enterprise Hostname
  - Extrahop Reveal(x) Enterprise API Key

Time Zone

> ⚠ The time zone steps are for ThreatQ v5 only. You can skip these steps if you are on a ThreatQ v6 instance.

You should ensure all ThreatQ devices are set to the correct time, time zone, and date (UTC is recommended), and using a clock source available to all.

To identify which time zone is closest to your present location, use the `timedatectl` command with the list-`timezones` command line option.

For example, enter the following command to list all available time zones in Europe:

```
timedatectl list-timezones | grep Europe
Europe/Amsterdam
Europe/Athens
Europe/Belgrade
Europe/Berlin
```

Enter the following command, as root, to change the time zone to UTC:

```
timedatectl set-timezone UTC
```

Integration Dependencies

> ⚠ The integration must be installed in a python 3.6 environment.

The following is a list of required dependencies for the integration. These dependencies are downloaded and installed during the installation process. If you are an Air Gapped Data Sync (AGDS) user, or run an instance that cannot connect to network services outside of your infrastructure, you

will need to download and install these dependencies separately as the integration will not be able to download them during the install process.

> Items listed in bold are pinned to a specific version. In these cases, you should download the version specified to ensure proper function of the integration.

DEPENDENCY	VERSION	NOTES
threatqsdk	>=1.8.6	N/A
threatqcc	>=1.4.2	N/A
tqTaxiiExport	>=2.1.1	N/A

Installation

Upgrading Users - Review the Change Log for updates to configuration parameters before updating. If there are changes to the configuration file (new/removed parameters), you must first delete the previous version's configuration file before proceeding with the install steps listed below. Failure to delete the previous configuration file will result in the connector failing.

ThreatQ v6 Process

1. Download the connector integration file from the ThreatQ Marketplace.
2. Transfer the connector whl file to the `/tmp/` directory on your instance.
3. SSH into your instance.
4. Move the connector whl file from its `/tmp/` location to the following directory: `/opt/tqvenv`
5. Navigate to the custom connector container:

```
kubectl exec -n threatq -it deployments/custom-connectors -- /bin/bash
```

6. Create your python 3 virtual environment:

```
python3.6 -m venv /opt/tqvenv/<environment_name>
```

7. Active the new environment:

```
source /opt/tqvenv/<environment_name>/bin/activate
```

8. Run the pip upgrade command:

```
pip install --upgrade pip
```

9. Install the required dependencies:

```
pip install setuptools==59.6.0 threatqsdk threatqcc
```

10. Install the connector:

```
pip install /opt/tqvenv/tq_conn_extrahop-<version>-py3-none-any.whl
```

11. Perform an initial run of the connector:

```
/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop --cron="0 */2 * *
*"
```

> The `--cron` argument above is used to generate a cron job for the connector. After running the command above, the cronjob will be created under the /etc/cron.d/ directory. This entry will initially be commented out upon creation - see the CRON chapter for more details.

12. Enter the following parameters when prompted:

PARAMETER	DESCRIPTION
ThreatQ Host	**Leave this field blank as it will be set dynamically.**
ThreatQ Client ID	This is the OAuth id that can be found at Settings Gear → User Management → API details within the user's details.
ThreatQ Username	This is the Email Address of the user in the ThreatQ System for integrations.
ThreatQ Password	The password for the above ThreatQ account.
Status	This is the default status for objects that are created by this Integration.

Example Output

```
/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop --cron="0 */2 * * *"
ThreatQ Host:
ThreatQ Client ID: <ClientID>
ThreatQ Username: <EMAIL ADDRESS>
ThreatQ Password: <PASSWORD>
Status: Review
Connector configured. Set information in UI
```

You will still need to configure and then enable the connector.

ThreatQ v5 Process

1. Navigate to the ThreatQ Marketplace and download the .whl file for the integration.
2. Create the following directory:

```
mkdir /opt/tqvenv/
```

3. Install python 3.6:

```
sudo yum install -y python36 python36-libs python36-devel python36-pip
```

4. Create a virtual environment:

```
python3.6 -m venv /opt/tqvenv/<environment_name>
```

5. Activate the virtual environment:

```
source /opt/tqvenv/<environment_name>/bin/activate
```

6. Run the pip upgrade command:

```
pip install --upgrade pip
```

7. Install the required dependencies:

```
pip install threatqsdk threatqcc setuptools==59.6.0
```

8. Transfer the whl file to the `/tmp` directory on your ThreatQ instance.
9. Install the connector on your ThreatQ instance:

```
pip install /tmp/tq_conn_extrahop-<version>-py3-none-any.whl
```

> A driver called `tq-conn-extrahop` will be installed. After installing, a script stub will appear in `/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop`.

10. Once the application has been installed, a directory structure must be created for all configuration, logs and files, using the `mkdir -p` command. Use the commands below to create the required directories:

```
mkdir -p /etc/tq_labs/ mkdir -p /var/log/tq_labs/
```

11. Perform an initial run using the following command:

```
/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop -ll /var/log/
tq_labs/ -c /etc/tq_labs/ -v3
```

12. Enter the following parameters when prompted:

PARAMETER	DESCRIPTION
ThreatQ Host	This is the host of the ThreatQ instance, either the IP Address or Hostname as resolvable by ThreatQ.
ThreatQ Client ID	This is the OAuth id that can be found at Settings Gear → User Management → API details within the user's details.

PARAMETER	DESCRIPTION
ThreatQ Username	This is the Email Address of the user in the ThreatQ System for integrations.
ThreatQ Password	The password for the above ThreatQ account.
Status	This is the default status for objects that are created by this Integration.

Example Output

```
/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop -ll /var/log/tq_labs/
-c /etc/tq_labs/ -v3
ThreatQ Host: <ThreatQ Host IP or Hostname>
ThreatQ Client ID: <ClientID>
ThreatQ Username: <EMAIL ADDRESS>
ThreatQ Password: <PASSWORD>
Status: Review
Connector configured. Set information in UI
```

You will still need to configure and then enable the connector.

Configuration

> ThreatQuotient does not issue API keys for third-party vendors. Contact the specific vendor to obtain API keys and other integration-related credentials.

To configure the integration:

1. Navigate to your integrations management page in ThreatQ.
2. Select the **Labs** option from the *Category* dropdown (optional).
3. Click on the integration entry to open its details page.
4. Enter the following parameters under the **Configuration** tab:

PARAMETER	DESCRIPTION
Select the type of deployment to use	Select your Extrahop deployment type. Options include: ◦ Extrahop Reveal(x) 360 ◦ Extrahop Reveal(x) Enterprise
Data Collection Name (Threat Library)	The Threat Library data collection that you want IoCs to be exported from. > Any ThreatQ data collection containing more than 21,000 indicators will be submitted to Extrahop in batches. See the Known Issues / Limitations chapter for more details.
Extrahop Reveal(x) 360 Hostname *(360 Deployment Only)*	Your Extrahop Reveal(x) 360 Hostname.
Extrahop Reveal(x) 360 Client ID *(360 Deployment Only)*	Your Extrahop Reveal(x) 360 Client ID.
Extrahop Reveal(x) 360 Client Secret *(360 Deployment Only)*	Your Extrahop Reveal(x) 360 Client Secret.

PARAMETER	DESCRIPTION
Extrahop Reveal(x) Enterprise Hostname *(Enterprise Deployment Only)*	Your Extrahop Reveal(x) Enterprise instance hostname or IP address.
Extrahop Reveal(x) Enterprise API Key *(Enterprise Deployment Only)*	Your Extrahop Reveal(x) Enterprise API Key. This can be found on your Extrahop instance under user (in top right) > API Access.

5. Review any additional settings, make any changes if needed, and click on **Save**.
6. Click on the toggle switch, located above the *Additional Information* section, to enable it.

Usage

Use the following command to execute the driver:

ThreatQ v6 Driver Command

```
/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop
```

ThreatQ v5 Driver Command

```
/opt/tqvenv/<environment_name>/bin/tq-conn-extrahop -v3 -ll /var/log/
tq_labs/ -c /etc/tq_labs/
```

Command Line Arguments

This connector supports the following custom command line arguments:

ARGUMENT	DESCRIPTION
`-h`, `--help`	Shows this help message and exits.
`-ll LOGLOCATION`, `--loglocation LOGLOCATION`	Sets the logging location for the connector. The location should exist and be writable by the current. A special value of 'stdout' means to log to the console (this happens by default).
`-c CONFIG`, `--config CONFIG`	This is the location of the configuration file for the connector. This location must be readable and writable by the current user. If no config file path is given, the current directory will be used. This file is also where some information from each run of the connector may be put (last run time, private oauth, etc.)
`-v {1,2,3}`, `--verbosity {1,2,3}`	This is the logging verbosity level where **3** means everything. The default setting is **1** (Warning).

ARGUMENT	DESCRIPTION
`-ep, --external-proxy`	This allows you to use the proxy that is specified in the ThreatQ UI. This specifies an internet facing proxy, NOT a proxy to the TQ instance.
`-n NAME, --name NAME`	This sets the name for this connector. In some cases, it is useful to have multiple connectors of the same type executing against a single TQ instance. For example, the Syslog Exporter can be run against multiple target and multiple exports, each with their own name and configuration.
--cron	ThreatQ v6 Only - creates a CRON entry for the connector based on a pre-loaded ThreatQ template. See the CRON section for more details.

Accessing Connector Logs

ThreatQ v6

ThreatQ version 6 aggregates the logs for all custom connectors to its output container. You can access the container's log using the following command:

```
kubectl logs -n threatq deployments/custom-connectors
```

ThreatQ v5

The connector log directory was created in 10 of the installation process and is identified using the `-ll` argument flag when executing the driver.

Accessing Connector Configuration

ThreatQ v6

The custom connector configuration file can be found in the following directory: `/etc/tq_labs/`.

ThreatQ v5

The custom connector configuration file was created in step 10 of the install process and identified using the `-c` argument flag when executing the driver.

CRON

ThreatQ v6 CRON

The addition of the `--cron` argument in the initial run of connector, performed during the install process, resulted in the creation of a cron job file for the connector in the following directory: `/etc/cron.d/`. The contents of the file will resemble the following structure:

```
#{schedule} root /bin/bash -c "source /etc/env-vars.sh; {venv_path}/bin/
{executable} --config=/etc/tq_labs > /proc/1/fd/1 2>/proc/1/fd/2"
```

The `{schedule}` will be replaced with the cron settings you entered with the `--cron` flag and the `{executable}` will be replaced for with the connector's driver command.

You will also see a `#` at the beginning of the file. This comments out the job. This allows you to configure the custom connector in the ThreatQ UI first. After you have configured the connector in ThreatQ, you can remove the `#` from the file content's in order to activate the cron job.

To summarize this process:

1. Install the connector and perform an initial run using the `--cron` argument to create the cron job.
2. Complete the connector's configuration settings in the ThreatQ UI.
3. Access the connector's cron file in the `/etc/cron.d/` directory and remove the # from the beginning of the file.

ThreatQ v5 CRON

Automatic CRON configuration has been removed from this script. To run this script on a recurring basis, use CRON or some other jobs scheduler. The argument in the CRON script must specify the config and log locations.

Add an entry to your Linux crontab to execute the connector at a recurring interval. Depending on how quickly you need updates, this can be run multiple times a day (no more than once an hour) or a few times a week.

In the example below, the command will execute the connector every two hours.

1. Log into your ThreatQ host via a CLI terminal session.
2. Enter the following command:

   ```
   crontab -e
   ```

   This will enable the editing of the crontab, using vi. Depending on how often you wish the cronjob to run, you will need to adjust the time to suit the environment.
3. Enter the commands below:

   Every 2 Hours Example

```
0 */2 * * * /opt/tqvenv/<environment_name>/bin/tq-conn-extrahop -c /
etc/tq_labs/ -ll /var/log/tq_labs/ -v3
```

4. Save and exit CRON.

Known Issues / Limitations

- Extrahop limits the amount of indicators a data collection can have based on the type of account with lower accounts only allowing 24,000 indicators per collection on the Extrahop platform. Version 1.3.2 of the integration will now submit data collections larger that 21,000 indicators in batches. These batches will create multiple data collections on the Extrahop platform using the following naming convention: {ThreatQ collection}_batch_{batch number}.

Change Log

- **Version 1.3.2 rev-a**
  - Guide Update - Added ThreatQ v6 installation steps.
- **Version 1.3.2**
  - Resolved an issue where customers were encountering 400 and 403 errors when using large data collections (24,000+ indicators). Data collections larger than 21,000 indicators will now be exported to Extrahop in batches. See the Known Issues / Limitations chapter for more details.
- **Version 1.3.1**
  - Updated the naming of deployment methods in the configuration page: Online is now **Reveal(x) 360** and On-Prem is now **Reveal(x) Enterprise**.
- **Version 1.3.0**
  - Added support for Extrahop Cloud deployments.
  - Add new configuration option to allow you to select your deployment methods.
  - Added additional configuration fields for Online Deployment method: Online Hostname, Online Client ID, and Online Client Secret.
  - Updated minimum ThreatQ version to 5.6.0.
- **Version 1.2.0**
  - Added Python 3 support.
  - Updated integration to use latest ThreatQ SDK.
- **Version 1.1.0**
  - Added support for Unicode characters.
- **Version 1.0.0**
  - Initial Release