# ThreatQuotient

## ThreatQuotient for FireEye EX Connector Guide

Version 1.0.0

Thursday, May 21, 2020


**ThreatQuotient**

11400 Commerce Park Dr., Suite 200

Reston, VA 20191

**Support**

Email: support@threatq.com

Web: Support.threatq.com

Phone: 703.574.9893

# 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.

# Contents

# Versioning

- Integration Version: 1.0.0
- ThreatQ Version: 4.34.0 or greater

| Operating System | OS Version | Python Version | Notes |
|---|---|---|---|
| RedHat/CentOS | 7 | 2.7.12 | N/A |
| Ubuntu | 16.04 | 2.7.12 | This has not been tested. |
| Windows | 2012R2/10 | 2.7.12 | This has not been tested. |

# Introduction

The FireEye EX connector integrates ThreatQ with a FireEye EX appliance. It is a bi-directional integration that is used for 1) uploading YARA rules from a saved search in ThreatQ to FireEye, and 2) searching in FireEye EX for alerts and creating events from those alerts in ThreatQ, including any indicators that have been found in the alerts.

The connector runs a saved search in ThreatQ, parses the YARA rules from the search, and sends each of the rules to FireEye EX. If the customer has enabled it, the connector also searches for email alerts, and if it finds any, adds them to ThreatQ as events. Any indicators in the alerts are also added to ThreatQ and related to the event.

# Installation

This package is available in `.tar.gz` and `.whl` formats, and can be installed from the ThreatQ integrations repository.

To install the `.tar.gz` or `.whl` formats:

```
pip install tq_conn_fireeye_ex
```

## Executing the Driver

This package comes with a driver called `tq-conn-fireeye-ex`. After installing with `pip` or `setup.py`, a script stub will appear in `/usr/bin/tq-conn-fireeye-ex`.

To execute the feed just use:

```
tq-conn-fireeye-ex -c /path/to/config/directory/ -ll

/path/to/log/directory/ -v VERBOSITY_LEVEL
```

The driver will run once, where it will connect to the TQ instance and will install the UI component of the connector. After installation, the user will need to go into the connector UI and configure the required fields.

If you have multiple FireEye EX appliances, you can configure separate custom connectors for each appliance. The connectors should be given different names when configured, and in the IP/Hostname parameter in the TQ UI you would need to provide the hostnames for each of the appliances.

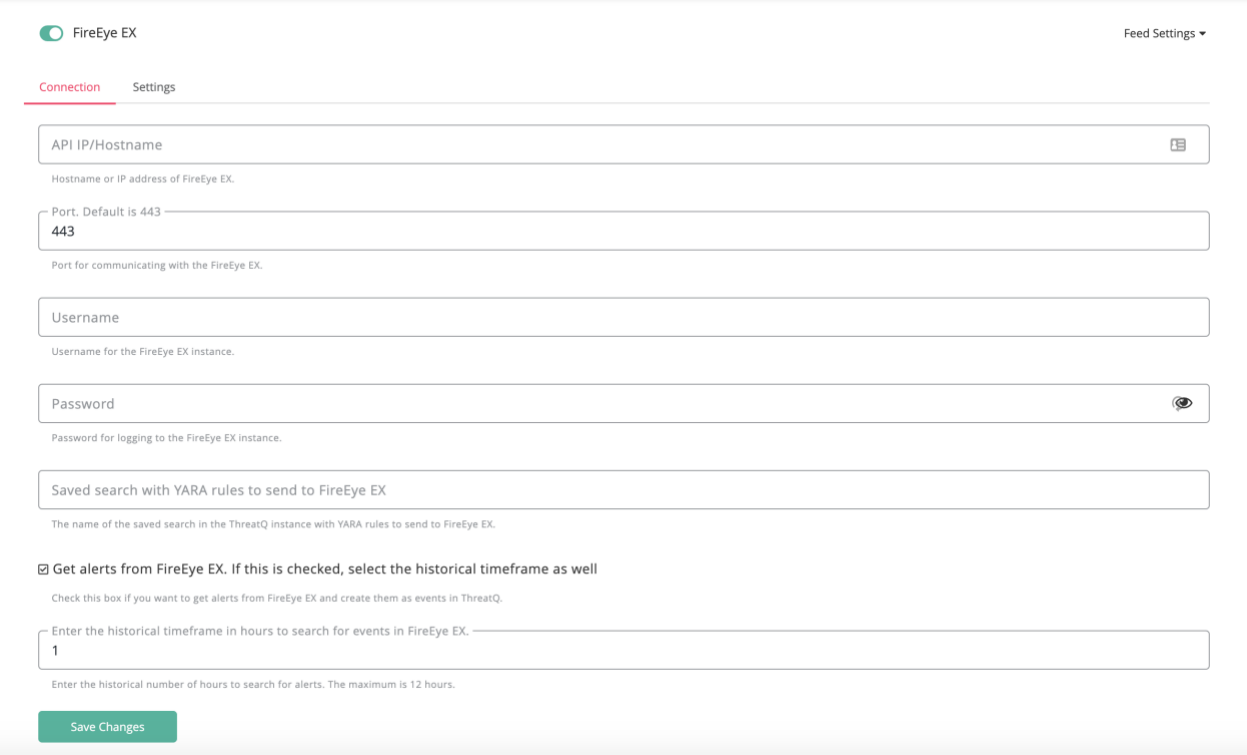
# Configuration

***Note:*** *ThreatQuotient does not issue API keys for third-party vendors. Contact the specific vendor to obtain API keys and other feed-related credentials.*

**To configure the connector:**

1. Click on the **Settings** icon and select **Incoming Feeds**.
2. Locate the connector under the **Labs** tab.
3. Click on the **Feed Settings** link for the connector.
4. Under the Connection tab, enter the following configuration parameters:

| Parameter | Description |
|---|---|
| IP/Hostname | Hostname or IP address of the FireEye EX |
| Port | Port number the FireEye EX API is listening on<br>The default port is 443. |
| Username | The provided username for FireEye EX |
| Password | The password for logging to FireEye EX |
| Saved Search | The name of the saved search in the ThreatQ instance with YARA rules (Optional) |
| Get alerts from FireEye EX (checkbox) | Check this box if you want to get alerts from FireEye EX and create them as events in ThreatQ |
| Enter the historical timeframe in hours to search for events in FireEye EX | Enter the historical number of hours to search for alerts. The maximum historical timeframe is 12 hours. |




5. Click on **Save Changes**.
6. Click on the toggle switch to the left of the connector name to enable the connector.

# 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 at the top of every hour.

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. To execute the connector at a scheduled frequency, you can configure a CRON entry to run the connector. Depending on how quickly you want updates, this can be run multiple times a day (no more than once an hour) or a few times a week.

   **Hourly Example**

```
 0 * * * * /usr/bin/tq-conn-fireeye-ex -c

/path/to/config/directory/ -ll /path/to/log/directory/ -v

VERBOSITY_LEVEL
```

4. Save and exit cron.

## 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. The default is 1 (Warning). |
| `-ep, --external-proxy` | This allows you to use the proxy that is specified in the ThreatQ UI |
| `-ds, --disable-ssl` | Adding this flag will disable SSL verification when contacting the FireEye EX API |

# Change Log

| Version | Details |
|---|---|
| 1.0.0 | • Initial Release |