ThreatQuotient for FireEye Application

May 14, 2018

Version 2.0

11400 Commerce Park Dr
Suite 200,
Reston, VA
20191, USA
https://www.threatq.com/
Support: support@threatq.com
Sales: sales@threatq.com

Contents

CONTENTS **2**

LIST OF FIGURES AND TABLES **3**

ABOUT THIS THREATQUOTIENT FOR FIREEYE APPLICATION **4**

HISTORY 4
REVIEW 4
DOCUMENT CONVENTIONS 4

1 INTRODUCTION **5**

1.1 APPLICATION FUNCTION 5
1.2 PREFACE 5
1.3 AUDIENCE 5
1.4 SCOPE 5
1.5 ASSUMPTIONS 6

2 IMPLEMENTATION OVERVIEW **7**

2.1 PREREQUISITES 7
2.2 SECURITY AND PRIVACY 8

3 FIREEYE APPLICATION INSTALLATION **9**

3.1 SETTING UP THE INTEGRATION 9
3.2 CONFIGURING THE CONNECTOR 11
3.3 CRON 12
3.3.1 Setting Up the CRONJOB *12*

APPENDIX A: SUPPLEMENTARY INFORMATION. **13**

UNINSTALLING THE CONNECTOR 13
DRIVER COMMAND LINE OPTIONS 13

TRADEMARKS AND DISCLAIMERS **14**

List of Figures and Tables

FIGURE 4: TIME ZONE CHANGE EXAMPLE 7
FIGURE 1: ENABLE CONFIGURATION FIREEYE 7
FIGURE 2: ENABLE WEB SERVICES API 7
FIGURE 3: VERIFY WEB SERVICES API ENABLED 7
FIGURE 4: TIME ZONE CHANGE EXAMPLE 7
FIGURE 5: INSTALLING .WHL FILE (INC EXAMPLE OUTPUT) 9
FIGURE 6: CREATING INTEGRATION DIRECTORIES EXAMPLE 9
FIGURE 7: RUNNING THE INTEGRATION 10
FIGURE 8: THREATQ UI CONFIGURATION 11
FIGURE 9: RUNNING OF THE INTEGRATION (EXAMPLE OUTPUT) 11
FIGURE 10: COMMAND LINE CRONTAB COMMAND 12
FIGURE 11: COMMAND LINE CRONTAB FIREEYE COMMAND 12

TABLE 1: DOCUMENT HISTORY INFORMATION 4
TABLE 2: DOCUMENT REVISION INFORMATION 4
TABLE 3: THREATQUOTIENT SOFTWARE & APP VERSION INFORMATION 5

About This ThreatQuotient for FireEye Application

Author ThreatQuotient Professional Services

History

Table 1: Document History Information

Version No.	Issue Date	Status	Reason for Change
0.1	21 Mar 2018	Initial Draft	Initial draft
0.2	22 Mar 2018	First Draft	ThreatQuotient internal review
1.0	22 Mar 2018	Release	Document Release
1.1	18 Apr 2018	Second Draft	ThreatQuotient internal review
1.2	04 May 2018	Third draft	ThreatQuotient internal review
1.3	08 May 2018	Second Release	Document Release minor changes

Review

Table 2: Document Revision Information

Reviewer's Details	Version No.	Date
Larry Selvy	0.1	22 Mar 2018
Les Adams	0.2	22 Mar 2018
Larry Selvy	1.0	22 Mar 2018
Les Adams	1.1	18 Apr 2018
Tony Michelizzi	1.2	04 May 2018
Les Adams	1.3	08 May 2018
Leon Brown	2.0	13 May 2018

Document Conventions

Alerts readers to take note. Notes contain suggestions or references to material not covered in the document.

Alerts readers to be careful. In this situation, you may do something that could result in equipment damage or loss of data.

Alerts the reader that they could save time by performing the action described in the paragraph.

Alerts the reader that the information could help them solve a problem. The information might not be troubleshooting or even an action.

1 Introduction

1.1 Application Function

The ThreatQuotient for FireEye Application is a uni-directional connector that pulls alerts from FireEye CMS and uploads the data as indicators and events to a ThreatQ instance. The events are tagged as "Malware" type events. This connector is meant to attach to a single FireEye CMS instance.

 The upload of data can take a quite some time. (>1 hour).

1.2 Preface

This guide provides the information necessary to implement the ThreatQuotient for FireEye Application. This document is not specifically intended as a site reference guide.

It is assumed that the implementation engineer has experience installing and commissioning ThreatQuotient Apps and integrations covered within the document, as well as experience necessary to troubleshoot at a basic level.

1.3 Audience

This document is intended for use by the following parties:

1. ThreatQ and FireEye Analysts/Engineers
2. ThreatQuotient Professional Services Project Team & Engineers

1.4 Scope

This document covers the implementation of the ThreatQuotient for FireEye Application only.

Table 3: ThreatQuotient Software & App Version Information

Software/App Name	File Name	Version
ThreatQ	Version 3.6.x or greater	
ThreatQuotient for FireEye Application	3.1.0	

1.5 Assumptions

The following criteria is assumed to be in place and functional to allow the implementation of the ThreatQuotient for FireEye Application into the managed estate:

- All ThreatQuotient equipment is online and in service.
- Infrastructure/transmission at all sites and between sites is in place to support the network traffic.
- All required firewall ports have been opened.
- All equipment is powered from permanent power supplies.
- A clock source of sufficient accuracy is connected to the network and the network and devices are using it as the primary clock source.

2 Implementation Overview

This document will show how to install the ThreatQuotient for FireEye Application.

2.1 Prerequisites

Throughout this implementation document, there will be referrals to several files and directories, some of which will be symbolic, and others may change depending on specifics of the environmental setup.

Ensure all ThreatQ devices are set to the correct time, time zone and date, and using a clock source available to all.

Figure 1: Time Zone Change Example

```
sudo ln -sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
```

The Web Services API must be enabled to allow the API to be functional. Please follow the steps below. At the time of this writing, the steps below are correct For further reference, please refer to the FireEye documentation found here: https://docs.fireeye.com/docs/docs_en/CM/sw/2018.02/API/API_2018.02_en.pdf.

Enabling the Web Services API

To enable the Web Services API on your appliance:

1. In a terminal window, log in to the command-line interface (CLI) on the appliance where you will run the Web Services API.
2. Enable the CLI configuration mode:

   Figure 2: Enable Configuration FireEye

   ```
   hostname > enable
   hostname # configure terminal
   ```
3. Enable the Web Services API:

   Figure 3: Enable Web Services API

   ```
   hostname (config) # wsapi enable
   ```
4. Verify that the Web Services API is enabled.

   Figure 4: Verify Web Services API enabled

   ```
   show wsapi
   ```

   Figure 5: Time Zone Change Example

   ```
   sudo ln -sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
   ```

2.2 Security and Privacy

For ThreatQuotient Professional Services engineers to configure the system, local network access is required to connect to the managed estate. Therefore, the implementation must occur at an office or data center location.

Passwords have not been provided in this document. Please contact your project team for this information, if required.

All engineers are reminded that all data belonging and pertaining to the business is strictly confidential and should not be disclosed to any unauthorized parties.

The data held within this document is classed as confidential due to its nature.

3 FireEye Application Installation

3.1 Setting up the Integration

Ensure the file `tqFireEye-3.2.0-py2-none-any.whl` has been added to the ThreatQ instance. Or the Threat Q instance has internet connectivity.

1. Install the .whl file using the following command.

Figure 6: Installing .whl File (Inc Example Output)

```
[root@localhost]# sudo pip install -i
https://<USERNAME>:<PASSWORD>@extensions.threatq.com/threatq/integrations tqFireEye
You are using pip version 7.1.0, however version 9.0.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
Collecting tqFireEye
  Downloading https://extensions.threatq.com/threatq/integrations-
dev/+f/8eb/f32d6cd5298b1/tqFireEye-3.2.0-py2-none-any.whl
https://extensions.threatq.com/root/pypi/+f/598/499a75be2e5e1/python_dateutil-
2.5.3-py2.py3-none-any.whl (201kB)
    100% |████████████████████████████████| 204kB 268kB/s
Collecting jinja2==2.8 (from threatqcc>=1.3.0->tqFireEye)
  Downloading https://extensions.threatq.com/root/pypi/+f/1cc/03ef32b64be19/Jinja2-
2.8-py2.py3-none-any.whl (263kB)
    100% |████████████████████████████████| 266kB 278kB/s
Requirement already satisfied (use --upgrade to upgrade): requests>=2.9.1 in
/usr/lib/python2.7/site-packages (from threatqsdk>=1.6.7->tqFireEye)
Requirement already satisfied (use --upgrade to upgrade): six>=1.5 in
/usr/lib/python2.7/site-packages (from python-dateutil==2.5.3->tqFireEye)
Requirement already satisfied (use --upgrade to upgrade): MarkupSafe in
/usr/lib64/python2.7/site-packages (from jinja2==2.8->threatqcc>=1.3.0->tqFireEye)
Installing collected packages: jinja2, threatqsdk, threatqcc, python-dateutil,
tqFireEye
  Found existing installation: Jinja2 2.7.2
    Uninstalling Jinja2-2.7.2:
      Successfully uninstalled Jinja2-2.7.2
  Found existing installation: python-dateutil 2.6.0
    Uninstalling python-dateutil-2.6.0:
      Successfully uninstalled python-dateutil-2.6.0
Successfully installed jinja2-2.8 python-dateutil-2.5.3 threatqcc-1.3.0 threatqsdk-
1.7.0 tqFireEye-3.2.0 [root@localhost]#
```

Once the application has been installed, a directory structure must be created for all configuration, logs and files, using the `mkdir` command. See example below:

Figure 7: Creating Integration Directories Example

```
$>cd /opt/
$>mkdir integrations
$>cd integrations
$>mkdir fireeye
$>cd fireeye
$>mkdir config
$>mkdir logs
$>mkdir files
```

A driver called `tq-fireeye` is installed.

2. Issue the commands shown in **Figure 8: Running the Integration** to initialize the integration.
   - **ThreatQ Host**: ThreatQ Hostname or IP Address
   - **Connector Name**: FireEye CMS – Auto Filled
   - **Client ID**: The Client ID can be found within the ThreatQ instance, under **Settings →Oauth Management**.
   - **E-Mail Address**: ThreatQ account associated with the FireEye integration
   - **Password**: ThreatQ account password associated with the FireEye integration
   - **Status**: Active

Figure 8: Running the Integration

```
[root@localhost fireeye]# sudo tqfireeye -c /path/to/config/directory/
-ll /path/to/log/directory/ -ds -v 3
ThreatQ Host: <IP ADDRESS>
Connector Name: FireEye CMS
Client ID: <CLIENT ID>
E-Mail Address: <EMAIL ADDRESS>
Password:
Status: Active
Connector configured.  Set information in UI.
2018-03-22 09:09:09 - tqFireEye.tq_driver CRITICAL: Connector has been created,
please use UI for final configuration
[root@localhost fireeye]#
```

The driver will run once, where it connects to the ThreatQ instance and installs the UI component of the connector.

3.2 Configuring the Connector

To edit the configuration, go to the **Incoming Feeds** page within ThreatQ, click the **ThreatQ Labs** tab, then expand the Feed Settings for the **FireEye CMS** section.

1. The following information will need to be entered as described below.
   - **Host**: This is the FireEye CMS host IP address or host name.
   - **Username**: This is the FireEye CMS User account.
   - **Password**: This is the password associated with the user account above.
   - **Alert Duration (hours):** Specifies the time interval to search in hours. Valid intervals are 1, 2, 6, 12, 24 and 48 hours. Default is 48 hours**.**
   - **Saved Search**: This is only required for the ThreatQ IoC -> FireEye Custom IoC List functionality.

Figure 9: ThreatQ UI Configuration

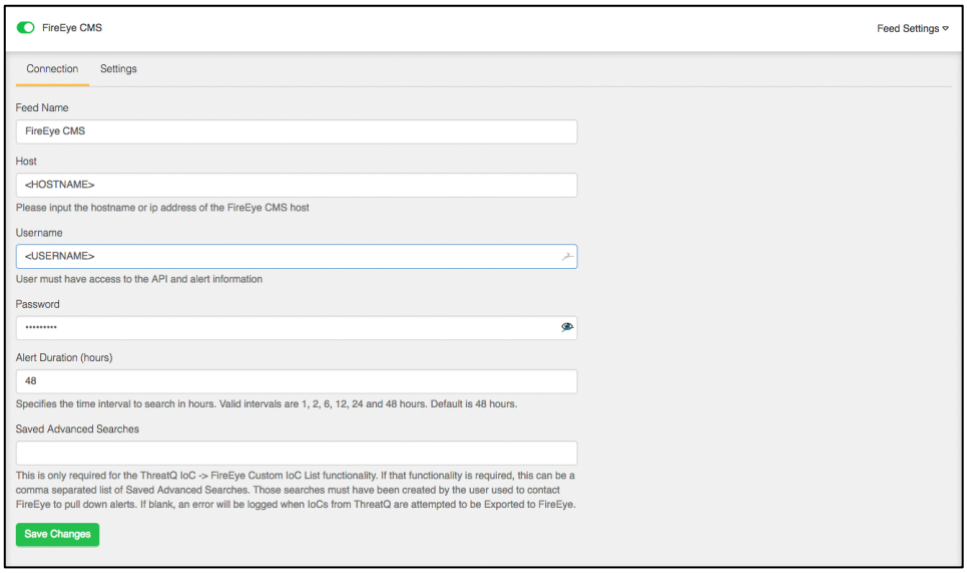

Figure 10: Running Of The Integration (Example Output)

```
$> sudo tqfireeye -c /opt/integration/fireeye/config/ -ll
/opt/integration/fireeye/logs/ -ds -v 3
0000-00-00 00:00:00 - tqFireEye.tq_driver DEBUG: Private Connection Established
0000-00-00 00:00:00 - tqFireEye.tq_driver INFO: Connection to FireEye CMS
Established
0000-00-00 00:00:00 - tqFireEye.tqFireEyeSDK.tqFireEyeAlerts INFO: Checking with
end time of 0000-00-00T00:00:00.000+00:00 for duration 00_hours
0000-00-00 00:00:00 - tqFireEye.tqFireEyeSDK.tqFireEyeConnection DEBUG: FireEye
Request: GET - alerts
for Alert 332 of type MALWARE_OBJECT
0000-00-00 00:00:00 - tqFireEye.tqFireEyeSDK.tqFireEyeAlerts INFO: Parsing related
indicators for Alert 000 of type MALWARE_OBJECT
0000-00-00 00:00:00 - tqFireEye.tqFireEyeSDK.tqFireEyeConnection DEBUG: FireEye
Request: POST - auth/logout
0000-00-00 00:00:00 - tqFireEye.tqFireEyeSDK.tqFireEyeConnection INFO: Logged out
of FireEye CMS Session
0000-00-00 00:00:00 - tqFireEye.tq_driver INFO: Completed processing: Uploaded 11
alerts [Some may not be unique]
0000-00-00 00:00:00 - tqFireEye.tq_driver INFO: Uploaded/Related 0 indicators [Not
these are not unique indicators]
0000-00-00 00:00:00 - tqFireEye.tq_driver INFO: Completed execution of the FireEye
CMS Connector in 00 seconds
```

3.3 CRON

To run this script on a recurring basis, use CRON or some other system schedule. The argument in the CRON script ***must*** specify the config and log locations.

This can be run multiple times a day and should not be run less than once every 2 hours, to avoid conflicts with long download times.

3.3.1 Setting Up the CRONJOB

1. Login via a CLI terminal session to your ThreatQ host.
2. Input the commands below.

Figure 11: Command Line Crontab 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. Input the commands below – this example shows every **4 Hours**.

Figure 12: Command Line Crontab FireEye Command

```
0 */4 * * * tqfireeye -c /path/to/config/directory/
-ll /path/to/log/directory/ -ds -v 3
```

To run this script on a recurring basis use CRON or some other on system schedule. CRON is shown here.


The argument in the CRON script ***must*** specify the config and log locations.


This can be run multiple times a day and should **not** be run more often than once/hr.

For further reference, see the ThreatQ Help Center.

Appendix A: Supplementary Information

Uninstalling the Connector

```
sudo pip uninstall tqfireeye
```

Driver command line options

The tqfireeye driver has several command line arguments that will help you and your customers execute. They are listed below. You can view these arguments by executing `/usr/bin/tqfireeye --help`.

```
usage: tqfireeye Connector [-h] [-ll LOGLOCATION][-c CONFIG] [-v VERBOSITY]
```

```
tqfireeye
```

optional arguments:

```
  -h, --help
```

Shows the help message and exit.

```
  -ll LOGLOCATION, --loglocation LOGLOCATION
```

This sets the logging location for this connector. The location should exist and be writable by the current user. A special value of 'stdout' means to log to the console (the default).

```
  -c CONFIG, --config CONFIG
```

This is the location of the configuration file for the connector. This location must have read and write permissions for the current user. If no config file is given, the current directory will be used. This file is also where some information from each run of the connector may be put (e.g. 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).

Trademarks and Disclaimers

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR THREATQUOTIENT REPRESENTATIVE FOR A COPY.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THIRD PARTY SUPPLIERS ARE PROVIDED “AS IS” WITH ALL FAULTS. THREATQUOTIENT AND THIRD-PARTY SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL THREATQUOTIENT OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF THREATQUOTIENT OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

ThreatQuotient and the ThreatQuotient Logo are trademarks of ThreatQuotient, Inc. and/or its affiliates in the U.S. and other countries.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.