ThreatQuotient

Email Alerts Connector User Guide

Version 1.2.1 rev-a

January 23, 2025

ThreatQuotient
20130 Lakeview Center Plaza Suite 400
Ashburn, VA 20147

Not Actively Supported

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 **Not Actively Supported**.

Integrations, apps, and add-ons designated as **Not Actively Supported** are not supported by ThreatQuotient’s Customer Support team.

While you can report issues to ThreatQ’s Customer Support team regarding the integration/app/add-on, you are solely responsible for ensuring proper functionality and version compatibility of Not Supported designations with the applicable ThreatQuotient software.

If unresolvable functional or compatibility issues are encountered, you may be required to uninstall the integration/app/add-on from your ThreatQuotient environment in order for ThreatQuotient to fulfill support obligations.

> ⚠ For ThreatQuotient Hosted instance customers, the Service Level Commitment and Service Level Credit in the ThreatQuotient Service Level Schedule will not apply to issues caused by Not Actively Supported integrations/apps/add-ons.

Integration Details

ThreatQuotient provides the following details for this integration:

Current Integration Version	1.2.1
Compatible with ThreatQ Versions	>= 5.6.0
ThreatQ Alerts Library Version	1.6.0
Python Version	3.6
Support Tier	Not Actively Supported

Introduction

The Email Alerts Connector for ThreatQ operates as an alert/notification system for ThreatQ. The connector will send emails to a specified email address when the following changes occur:

- Watchlist changes
- Data collection updates

> ⚠ The Email Alerts Connector requires the ThreatQ Alerts Service Library.

Prerequisites

Review the following requirements before attempting to install the connector.

Time Zone

> ⚠ The time zone steps are for ThreatQ v5 only. ThreatQ v6 users should skip these steps.

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
```

ThreatQ Alerts Service Library

The Email Alerts Connector requires ThreatQ Alerts Service, with a version equal to or greater than 1.6.0, on your ThreatQ instance.

> ⚠ Both the Email Alerts Connector and ThreatQ Alerts Service library must be installed in the same virtual environment.

The ThreatQ Alerts Service Library can be downloaded from the ThreatQ Marketplace.

Integration Dependencies

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
requests	N/A	N/A
threatqsdk	=>1.8.8	N/A
threatqcc	=>1.4.2	N/A
tq-alert-service	>=1.6.0	N/A

Installation

The following provides you with steps on installing a Python 3 Virtual Environment and installing the connector.

> ⚠ The Email Alerts Connector and the required ThreatQ Service Alerts Library must be installed in the same virtual environment.

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 threatqsdk threatqcc tq-alert-service requests
   ```
10. Install the connector:
    ```
    pip install /opt/tqvenv/tq_conn_email_alerts-<version>-py3-none-any.whl
    ```
11. Perform an initial run of the connector:
    ```
    /opt/tqvenv/<environment_name>/bin/tq-conn-email-alerts --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-email-alerts --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 tq-alert-service requests
```

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_devo_ioc_exporter-<version>-py3-none-any.whl
```

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

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-email-alerts -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-email-alerts -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
ThreatQ Hostname/IP	The hostname/IP address that you access ThreatQ from in your browser. This is used to link notifications directly to your ThreatQ instance
Email	The email address you want to use to send emails.
Skip Login	Specify if a login to the SMTP server is required to send emails.
Username	The username for the email account if different than the Email parameter.
Password	The password to authenticate with the specified email list in the Username parameter.
Port	The port used by your SMTP server to send emails.
Use SSL	Specify whether use SSL to connect to the SMTP server.
Recipients	A comma-delimited list of email addresses to receive the notifications.
Enable Watchlist Notifications	Select if you want to enable notifications on your watchlist.

PARAMETER	DESCRIPTION
Watchlist Users	A comma-delimited list of users (names or emails) you want to check the watchlist for changes.
Enable Data Collection Notifications	Select if you want to enable notifications on a data collection changes.
Data Collection Name	The name of the Data Collection you want to receive notifications on.
Data Collection Object Selection	Check one or more objects that you want to to be alerted on
Notification Frequency (Saved Search)	Select when you receive notifications on the Saved Search/Data Collection list in the Saved Search Name parameter. Options include: ◦ Only notify when new items are created. ◦ Notify anytime an item is added or updated with context.

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(s) to execute the driver:

ThreatQ v6 Driver Commands

Tasks, Watchlists, or Data Collections

```
/opt/tqvenv/<environment_name>/bin/tq-conn-email-alerts
```

Send Generic Email with Subject and Body

```
/opt/tqvenv/environment_name/bin/tq-conn-email-alerts -p '{"email_subject":
"This is a test subject", "email_body": "Test email body"}' -v3
```

ThreatQ v5 Driver Commands

Tasks, Watchlists, or Data Collections

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

Send Generic Email with Subject and Body

```
/opt/tqvenv/<environment_name>/bin/tq-conn-email-alerts -ll /var/log/
tq_labs/ -c /etc/tq_labs/ -p '{"email_subject": "This is a test subject",
"email_body": "Test email body"}' -v3
```

Command Line Arguments

This connector supports the following custom command line arguments:

ARGUMENT	DESCRIPTION
`-h`, `--help`	Review all additional options and their descriptions.
`-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.
`-p`, `--payload`	Provides the option to send free form emails by passing a JSON payload with email subject and body. For example: *tq-conn-email-alerts -c /etc/threatq/ -ll /var/log/threatq/ -p '{"email_subject": "This is a test subject", "email_body": "Test email body"}' -v3*
--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-email-alerts -
c /etc/tq_labs/ -ll /var/log/tq_labs/ -v3
```

4. Save and exit CRON.

Email Examples

The following are example emails sent by the connector.

ThreatQ Watchlist: Indicator Added

This message is sent when Enable Watchlist Notifications is enabled and an object is added to the watchlist of one of the Watchlist users

Body

An item in your ThreatQ watchlist has been updated!

Details

Type: Indicator Item: 7.8.9.0 Changed At: 2021-10-28 20:32:57

ThreatQ Link: https://threatq.hostname.com/indicators/8/details

ThreatQ Watchlist: Indicator Updated

This message is sent when Enable Watchlist Notifications is enabled and an object is added to the watchlist of one of the Watchlist users

Body

An item in your ThreatQ watchlist has been updated!

Details

Type: Indicator Item: 7.8.9.0 Changed At: 2021-10-28 20:32:57

ThreatQ Link: https://threatq.hostname.com/indicators/8/details

What has changed? Comment added: 2

ThreatQ Data Collection Updated: 3 New Items

This message is sent when Enable Data Collection Notifications is enabled and a new object matches the filter of the specified data collection. Each item in the list is linked to its object details page in ThreatQ

Body

The following items have been updated in your data collection

Indicators 7.8.9.0 1.2.3.4 5.6.7.8

Change Log

- **Version 1.2.1 rev-a**
    - Guide Update - added ThreatQ v6 documentation.
- **Version 1.2.1**
    - Upgraded the integration to python 3.6.
    - Updated the minimum ThreatQ version to 5.6.0.
- **Version 1.2.0**
    - Added the ability to send free form emails by passing a JSON payload to the executable.
        - Updated **Usage** and **CRON** chapters.
- **Version 1.1.2**
    - Updated installation steps. (rev-a)
    - Fixed an issue regarding honoring the configured data collection by name.
- **Version 1.1.1**
    - Added a new configuration option, **Data Collection Object Selection**, to allow data collection notifications to be sent based on updates to a specific object type. See the Configuration chapter for more details.
    - Added Email examples to the Usage chapter of this guide.
- **Version 1.1.0**
    - Added a configuration option to allow sending emails without a login.
    - Added a configuration parameter, **Username**, to allow specifying a separate username if different from the one supplied in the Email parameter.
    - Added a configuration option to connect to SMTP using SSL (Previously SMTP over SSL was required)
- **Version 1.0.0**
    - Initial Release