ThreatQuotient

ThreatQ Connector for Microsoft 365 Defender

Version 1.2.1

March 15, 2024

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 **3**
Support **4**
Integration Details **5**
Introduction **6**
Prerequisites **7**
Time Zone 7
Permissions 7
Integration Dependencies 8
Installation **9**
Creating a Python 3.6 Virtual Environment 9
Installing the Connector 10
Configuration **12**
Usage **14**
Command Line Arguments 14
CRON 15
Known Issues / Limitations **16**
Change Log **17**

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	1.2.1
Compatible with ThreatQ Versions	>= 4.56.0
Python Version	3.6
Support Tier	ThreatQ Supported

Introduction

The ThreatQ Connector for Microsoft 365 Defender allows you to export indicators from ThreatQ directly to Microsoft Defender via Microsoft's 365 Defender API.

The connector utilizes the following endpoint:

- **Import Indicators** - `https://api.securitycenter.microsoft.com/api/indicators/import`

> There are several permission requirements in order to use this connector. See the Permissions section of the **Prerequisites** chapter for more details.

Prerequisites

Review the following requirements before attempting to install the connector.

Time Zone

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
```

Permissions

The connector requires the following permissions to call the Defender API:

PERMISSION TYPE	PERMISSION	PERMISSION DISPLAY NAME
Application	Ti.ReadWrite	'Read and write Indicators'
Application	Ti.ReadWrite	'Read and write All Indicators'
Delegated (work or school account)	Ti.ReadWrite	'Read and write Indicators'

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
MarkupSafe	>=2.0.1	N/A
certifi	>=2022.6.15	N/A
charset-normalizer	>=2.0.12	N/A
idna	>=3.3	N/A
jinja2	>=3.0.3	N/A
requests	>=2.27.1	N/A
urllib3	>=1.26.9	N/A
threatqsdk	>= 1.8.4	N/A
threatqcc	>= 1.4.2	N/A
python-dateutil	>= 2.8.2	N/A

Installation

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

Creating a Python 3.6 Virtual Environment

Run the following commands to create the virtual environment:

```
mkdir /opt/tqvenv/
sudo yum install -y python36 python36-libs python36-devel python36-pip
python3.6 -m venv /opt/tqvenv/<environment_name>
source /opt/tqvenv/<environment_name>/bin/activate
pip install --upgrade pip
pip install threatqsdk threatqcc python-dateutil setuptools==59.6.0
```

Proceed to Installing the Connector.

Installing the Connector

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.

1. Navigate to the ThreatQ Marketplace and download the .whl file for the integration.
2. Activate the virtual environment if you haven't already:

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

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

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

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

5. 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/
   ```

6. Perform an initial run using the following command:

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

7. 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-microsoft-defender -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.

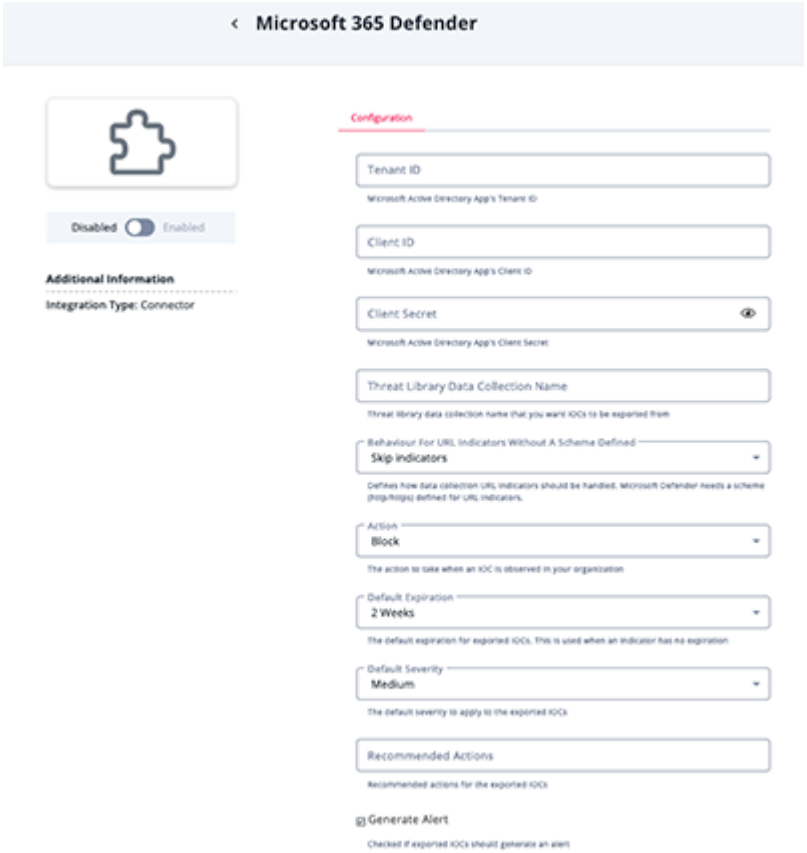
Configuration

> 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
Tenant ID	Your Microsoft Active Directory App's Tenant ID.
Client ID	Your Microsoft Active Directory App's Client ID.
Client Secret	Your Microsoft Active Directory App's Client Secret.
Threat Library Data Collection Name	Threat library data collection name that you want IOCs to be exported from.
Behavior for URL indicators without a scheme defined	Defines how data collection URL indicators should be handled.
Action	The action to take when an IOC is observed in your organization.
Default Expiration	The default expiration for exported IOCs. This parameter is used when an indicator has no expiration.
Default Severity	The default severity to apply to the exported IOCs.
Recommended Actions	The recommended actions for the exported IOCs.
Generate Alert	Generate an alert when IOCs are exported.



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:

```
/opt/tqvenv/<environment_name>/bin/tq-conn-microsoft-defender -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`	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.
`-n, --name`	Optional - Name of the connector (Option used in order to allow users to configure multiple connector instances on the same TQ box).

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-microsoft-defender -c /etc/tq_labs/ -ll /var/log/tq_labs/ -v3
   ```
4. Save and exit CRON.

Known Issues / Limitations

- The following is a list of limitations of the Import Indicators endpoint:
  - Rate limitations for this API are 30 calls per minute.
  - There is a limit of 15,000 active Indicators per tenant.
  - Maximum batch size for one API call is 500.
- Defender does not have a deduplication mechanism. This will result in duplicate indicators with different parameters being imported. **Example**: if an indicator with a specified action already exists on Defender, uploading the same indicator with a different action will result in the creation of a new indicator opposed to editing the existing entry.

Change Log

- **Version 1.2.1**
  - Related Malware and Adversary information is now added to the descriptions.
- **Version 1.2.0**
  - Added a retry mechanism for requests that return a 5xx status code. The retry will be attempted after a five second pause.
  - Set the maximum number of allowed 5xx status code request errors per session to 10.
- **Version 1.1.0**
  - Decreased the default expiration user field possible values.
  - TQ indicator expiration date is not taken into account anymore.
- **Version 1.0.0**
  - Initial release