

ThreatQuotient



ThreatQ Connector for Microsoft Exchange On-Prem

Version 1.1.0

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
Microsoft Exchange Server Credentials.....	7
Installing Cryptography	7
ThreatQ ACE Library	8
Integration Dependencies	9
Installation.....	10
Creating a Python 3.6 Virtual Environment	10
Installing the Connector.....	11
Configuration	13
Usage.....	16
Command Line Arguments.....	16
CRON	17
Known Issues / Limitations	18
Change Log	19

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.

Copyright © 2024 ThreatQuotient, Inc.

All rights reserved. This document and the software product it describes are licensed for use under a software license agreement. Reproduction or printing of this document is permitted in accordance with the license agreement.

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	1.1.0
------------------------------------	-------

Compatible with ThreatQ Versions	>= 4.40.0
---	-----------

Compatible with Microsoft Exchange Server Versions	2013
---	------

Microsoft Exchange Server Hosting Type	On-Prem
---	---------

Python Version	3.6
-----------------------	-----

Support Tier	ThreatQ Supported
---------------------	-------------------

Introduction

The ThreatQ Connector for Microsoft Exchange On-Prem is a unidirectional integration that integrates directly with the Microsoft Exchange mail server. The connector enables the ingestion of emails and attachments into ThreatQ and supports parsing emails and attachments as well as forwarded emails (spearphishing).

The connector ingests the following system objects:

- Attachments
- Events
- Indicators

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
```

Microsoft Exchange Server Credentials

The following Microsoft Exchange Server information is required to configure the connector:

- Username
- Password
- Email
- Exchange Server IP or Hostname


Installing Cryptography

If you are having an issue with installing cryptography, install it with these parameters:

```
pip install cryptography --global-option=build_ext --global-option="-L/usr/local/opt/openssl/lib" --global-option="-I/usr/local/opt/openssl/include"
```

ThreatQ ACE Library

The required ThreatQ ACE library is included in the connector's marketplace download. Use the following steps to install the library.

 When installing the library, be aware that any in-progress feed runs will be cancelled when initiating step 6.

1. Locate the threatq_ace-<version>-py3-none-any.whl library file bundled with the operation file.
2. SSH into your ThreatQ instance.
3. Navigate to tmp directory:

```
cd /tmp/
```

4. Upload the library file to this directory.
5. Install the library:


```
sudo -u apache /opt/threatq/python/bin/pip install threatq_ace-  
<version>-py3-none-any.whl
```

6. Restart Dynamo:


```
systemctl restart threatq-dynamo
```

7. Proceed with installing the connector.

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
exchangelib	4.6.2	Pinned
backports-datetime-fromisoformat	1.0.0	Pinned
msoffcrypto-tool	5.0.0	Pinned
openpyxl	3.1.2	Pinned
pikepdf	3.2.0	Pinned

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



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

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-tq-conn-ms-exchange-on-prem -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.

Example Output

```
/opt/tqenv/<environment_name>/bin/tq-conn-tq-conn-ms-exchange-on-prem
-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>
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 authentication method for the Exchange server	<p>Select your authentication method. Options include:</p> <ul style="list-style-type: none"> ◦ NTLM - will authenticate to the Exchange server using NTLM. ◦ Username and Password - will authenticate to the Exchange server using the provided username and password.
Email Address and Username are the Same	<p>If checked, the Email Address supplied below will be used for logging into the Exchange server. Otherwise a Username must be provided in the field provided.</p>
Server IP/Hostname	<p>The exchange server IP or hostname your organization uses</p>
Username	<p>The username for logging into the Exchange server.</p> <p>If you have enabled the Email Address and username for the server are the same option, enter the email address here.</p>
Email Address	<p>The email for the account you want to ingest email from.</p> <p>The email address will also be used as a username to log into the Exchange server if you have selected the option Email Address and username for the server are the same above.</p>

PARAMETER	DESCRIPTION
Password	The password for the account.
Mailbox Folders	Enter a comma-separated list of folders to pull emails from. The default setting is Inbox .
Mailbox Type	Select the mailbox type. Options include: <ul style="list-style-type: none"> ◦ Intelligence - This will import and parse all emails and attachments for indicators. ◦ Spearphish - This will only parse emails with spearphish attachments (.eml, .emlx, .pst, .ost).
Trim Number of Characters	<p>The number of characters to trim from the email body.</p> <div>  <p>The Trim Body parameter should be used with caution. Useful indicator data could be trimmed out resulting in ingested email files having an invalid format.</p> </div>
Use ACE Parser for Attachments	<p>Use the TQ ACE Parser library to parse attachments.</p> <div>  <p>This option requires the ACE Parser library to be installed on your ThreatQ instance. See ThreatQ ACE Library section for more details.</p> </div>
Default File Password	The password that will be used to decrypt encrypted attachments if a password cannot be parsed from a corresponding email.
Subject String	A string or regular expression that appears in the subject line of an email with an encrypted attachment and the email containing the attachment's password.
Password Prefix String	A string or regular expression that appears immediately before a password in the text body of an email containing an encrypted attachment's password.

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-tq-conn-ms-exchange-on-prem -v3
-ll /var/log/tq_labs/ -c /etc/tq_labs/
```

Command Line Arguments

This connector supports the following custom command line arguments:

ARGUMENT	DESCRIPTION
<code>-h, --help</code>	Review all additional options and their descriptions.
<code>-ll LOGLOCATION, --loglocation LOGLOCATION</code>	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).
<code>-c CONFIG, --config CONFIG</code>	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.)
<code>-v {1,2,3}, --verbosity {1,2,3}</code>	This is the logging verbosity level where 3 means everything. A special value of stdout means to log to the console (this happens by default).
<code>-ds, --disable_ssl</code>	Disables SSL verification.

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/tqenv/<environment_name>/bin/tq-conn-tq-conn-ms-  
exchange-on-prem -c /etc/tq_labs/ -ll /var/log/tq_labs/ -v3
```

4. Save and exit CRON.

Known Issues / Limitations

- The Trim Body parameter should be used with caution. Useful indicator data could be trimmed out resulting in ingested email files having an invalid format.

Change Log

- **Version 1.1.0**
 - Added the ability to parse encrypted xlsx and pdf email attachments using a default password or a password parsed from an email text body.
 - Added support for using the TQ ACE Parser library for parsing file attachments.
 - Added new configuration options:
 - Use ACE Parser for Attachments
 - Default File Password
 - Subject String Password
 - Prefix String
- **Version 1.0.1**
 - Resolved an issue regarding the Mailbox Folder.
- **Version 1.0.0**
 - Initial release