THALES

Thales Cybels Threat Intelligence connector for ThreatQ

Version 1.0.0

1 RELEASE NOTES

1.1 VERSION 1.0.0

Publication Date:

Connector updates in order to be compatible with ThreatQ marketplace.

2 INSTALLATION

This procedure applies to install Thales Cybels Threat Intelligence connector for ThreatQ.

2.1 SETTING UP THE INTEGRATION

1. Copy **tq-conn-thales-<Version x.x.x>-py3-none-any.whl** into your ThreatQ instance.

2. Install the .whl file in ThreatQ python3 virtualenv.

```
# source /opt/threatq/python/bin/activate
# pip install /file/path/to/app/tq-conn-thales-<Version x.x.x>-py3-none-any.whl
# deactivate
```

3. Create directory structure for all configuration, logs.

```
# sudo mkdir -p /opt/connectors/tq_conn_thales/conf
# sudo mkdir -p /opt/connectors/tq_conn_thales/logs
# sudo mkdir -p /opt/connectors/tq_conn_thales/data
```

4. Execute the following commands to initialize the integration and fill the configuration form

```
# /opt/threatq/python/bin/tq-conn-thales -ll /opt/connectors/tq_conn_thales/logs/ -n
"Thales" -c /opt/connectors/tq_conn_thales/conf/ --verbosity 3
```

Answer following questions

- **ThreatQ Host**: ThreatQ Host URL
- **Client ID**: OAuth Client ID of ThreatQ API

  *Retrieve Client ID on ThreatQ GUI under menu **User Settings → My Account → API Credentials***
- **E-Mail Address**: ThreatQ Maintenance Account

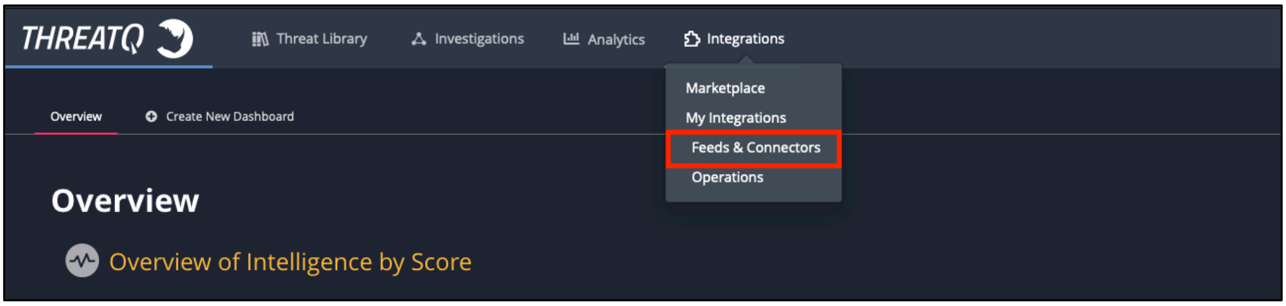
  We recommend to create an account dedicated to this usage
- **Password**: ThreatQ Maintenance Account password

Configuration example:

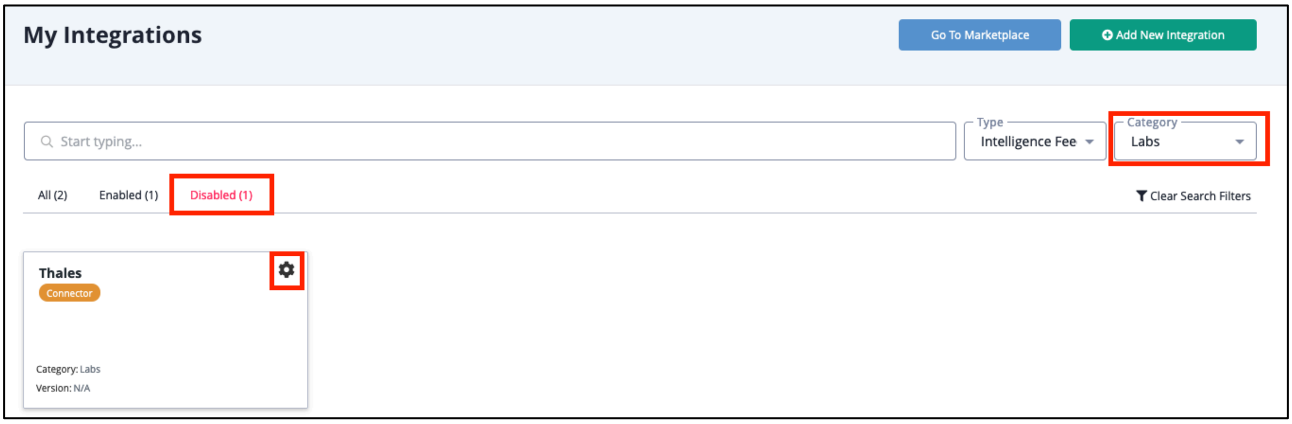
```
ThreatQ Host: https://127.0.0.1
Client ID: ************************
E-Mail Address: technical_feed_account@domain.com
Password: ***********
Connector configured.  Set information in UI.
```

2.2 CONFIGURATION

1. Log on ThreatQ Web Interface.
2. Navigate to your "*Integrations*" management page in ThreatQ.



3. Select the "*Labs*" option from the Category dropdown and click on "*Disabled*" tab. Configure the connector by clicking on the setting button.

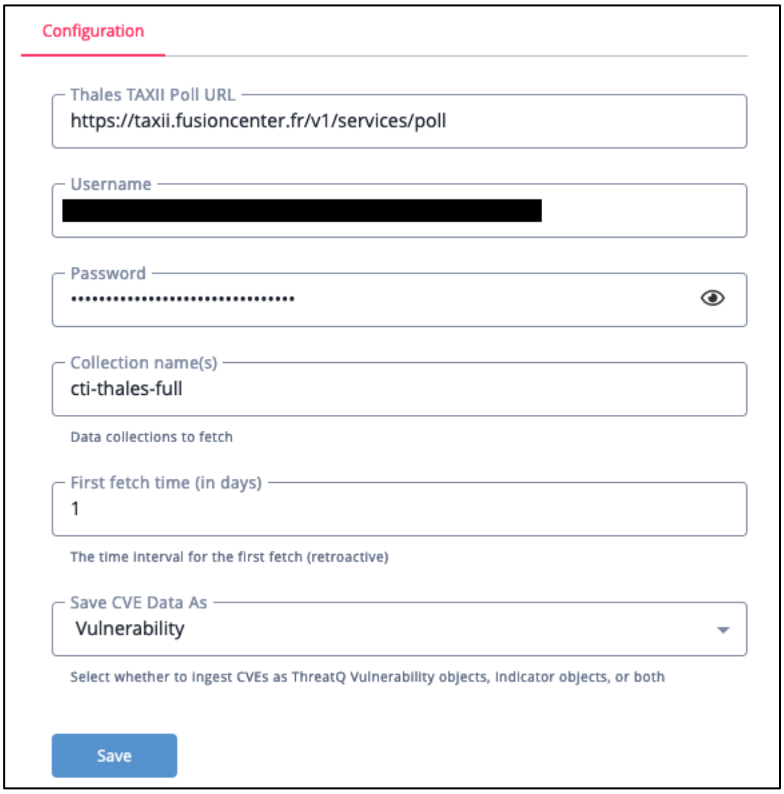


4. Enter the following configuration parameter:

Parameter	Description
Username	TAXII Username API
Password	TAXII Password API

Collection Name(s)	Comma-separated list of data collections to fetch. Default value: ‘cti-thales-full’
First Fetch Time	The time interval for the first fetch (retroactive) in days. Default value: 1
Save CVE Data As	Select whether to ingest CVEs as ThreatQ Vulnerability objects, Indicator objects, or both. Default value: Vulnerability

Configuration example:



Thales TAXII Poll URL: https://taxii.fusioncenter.fr/v1/services/poll

Username: [redacted]

Password: ••••••••••••••••••••••••••••••••••

Collection name(s): cti-thales-full

Data collections to fetch

First fetch time (in days): 1

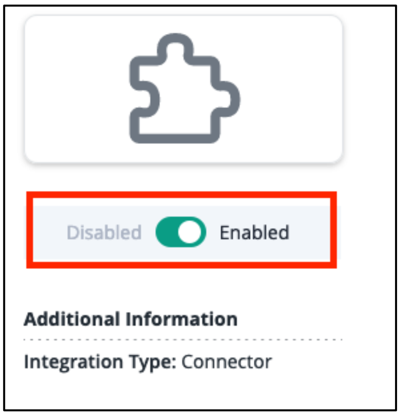
The time interval for the first fetch (retroactive)

Save CVE Data As: Vulnerability

Select whether to ingest CVEs as ThreatQ Vulnerability objects, Indicator objects, or both

5. Save changes

6. Enable Thales connector



Additional Information

Integration Type: Connector

2.3 SCHEDULE EXECUTION

1. Log with ssh on ThreatQ Appliance Operating System

2. Schedule automatic execution using CRON system. We recommend to execute Thales Connector each hour:

   ```
   # crontab -e
   ```

   Add the following line

   ```
   0 * * * * # /opt/threatq/python/bin/tq-conn-thales -ll
   /opt/connectors/tq_conn_thales/logs/ -n "Thales" -c
   /opt/connectors/tq_conn_thales/conf/ --verbosity 2
   ```

   The connector will run automatically at the next hour.

You can execute it manually by running the following command:

```
# /opt/threatq/python/bin/tq-conn-thales -ll /opt/connectors/tq_conn_thales/logs/ -n
"Thales" -c /opt/connectors/tq_conn_thales
```