ThreatQuotient

ServiceNow Connector User Guide

Version 2.5.2 rev-a

July 17, 2023

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

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	2.5.2
Compatible with ThreatQ Versions	>= 4.56.0
Compatible with ServiceNow Versions	Utah, Tokyo, San Diego
Python Version	3.6
Support Tier	ThreatQ Supported

Introduction

The ServiceNow Connector allows you to sync ServiceNow's default Service Desk Incidents, as well as Security Incidents, Security Cases, and Observables (indicators) to ThreatQ. You can also add ThreatQ related objects to ServiceNow.

> ⚠ The ServiceNow connector requires that your ThreatQ instance use the **same timezone** as your ServiceNow instance. See the Prerequisites section for more details.

Use Cases

This section will detail how the connector interacts with both platforms.

- If a new incident (security, case, service desk, etc.) is detected in ServiceNow, it will be added to ThreatQ with it's ServiceNow number appended to the title
  - If there are any parent or related incidents, they will be added and related in ThreatQ.
  - Problems and Change requests will *not* be added to ThreatQ.
- If an observable (indicator) is added to ServiceNow, it will be added to ThreatQ.
- If there has been any new relationships created between incidents and other incidents or incidents and observables, those relationships will be added to ThreatQ.
- If you add related indicators (observables) or events to a ThreatQ event (from ServiceNow), those related indicators/events will be added to ServiceNow and related to the incident.
  - Related events will only sync if they fall under the supported tables
  - Related indicators will only sync if they fall under the supported indicator types
- ThreatQ attributes and metadata for Indicators will be added to ServiceNow as an Observable's security annotations
  - See the 'Sync Security Annotations Immediately' option in the ThreatQ UI
- Workflow comments/work_notes will *not* be added to ThreatQ
  - They have been excluded because they provide unneeded information and clutter the comments section in ThreatQ.
- Occasionally, ServiceNow will not update the 'sys_updated_on property of incidents immediately. It could take 30 seconds. Keep that in mind while testing
- Due to syncing security annotations (indicator attributes), syncing will take longer
  - ServiceNow only supports 1 HTTP request per added security annotation at a time

ThreatQ Event Types

The following are the ThreatQ event types that are created when installing the integration

- ServiceNow Service Desk Incident
- ServiceNow Security Incident
- ServiceNow Security Case

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

This section will list tables that the integration will need read/write access to. This may vary depending on which options you have enabled in the ThreatQ UI configuration.

- incident
- sn_si_incident
- sn_ti_case
- sn_ti_m2m_task_observable
- sn_ti_m2m_case_task
- sn_ti_observable
- sn_ti_m2m_task_observable
- sys_journal_field (only before v2.0.2)

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
requests [security]	>=2.25.1	N/A
urllib3	**1.26.3**	**Pinned**
threatqsdk	>=1.8.1	N/A
threatqcc	>=1.4.1	N/A
python-dateutil	N/A	N/A
pytz	N/A	N/A
configparser	**4.0.2**	**Pinned**
future	N/A	N/A
six	**1.16.0**	**Pinned**
beautifulsoup4	**4.9.3**	**Pinned**
zipp	**1.2.0**	**Pinned**

DEPENDENCY	VERSION	NOTES
importlib-metadata	**2.1.1**	**Pinned**
flake8	N/A	N/A
markdownify	**0.9.4**	**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 python-dateutil
pip install 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_servicenow-<version>-py3-none-any.whl
```

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

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-servicenow -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-servicenow -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

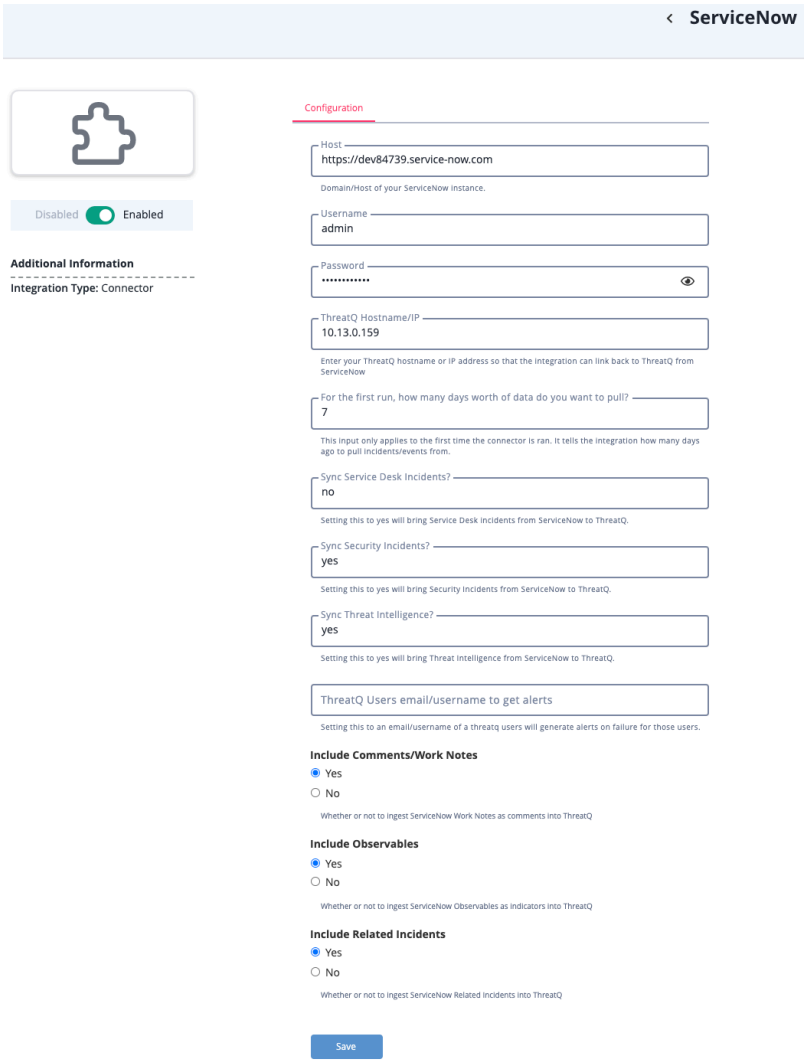
> ThreatQuotient does not issue API keys for third-party vendors. Contact the specific vendor to obtain API keys and other integration-related credentials.

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
Host	Your ServiceNow hostname. **https://** is optional.
Username	The ServiceNow username of the user that will be interacting with ServiceNow/ThreatQ.
Password	The password associated with the user listed above.
ThreatQ Hostname/IP	The ThreatQ IP address that Servicenow will connect with. This is the IP address in the address bar at the top of your ThreatQ window.
For the first run, how many days worth of data do you want to pull	This option will determine how many days in the past to pull information from. It must be a positive integer (no decimals).
Sync Service Desk Incidents	Select whether to sync Service Desk Incidents from ServiceNow to ThreatQ.
Sync Security Incidents	Select whether to sync Security Incidents from ServiceNow to ThreatQ. You must have the **Security Incident Response** plugin in order to select this option.

PARAMETER	DESCRIPTION
Sync Threat Intelligence	Select whether to sync Threat Intelligence from ServiceNow to ThreatQ. This also includes observables (indicators) and Security Cases (campaigns, adversaries, etc.) You must have the **Threat Intelligence** plugin in order to select this option.
ThreatQ Users email/ username to get alerts	The email/username of a ThreatQ user will generate alerts on failure for those users.
Include Comments/Work notes	Select whether to ingest ServiceNow Work Notes into ThreatQ as comments.
Include Observables	Select whether to ingest ServiceNow Observables into ThreatQ as indicators.
Include Related Incidents	Select whether to ingest ServiceNow Related Incidents into ThreatQ.
Strip HTML from comments	Select whether to strip HTML from comments when ingesting the data from ServiceNow to ThreatQ.



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-servicenow -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`	Shows this help message and exits.
`-n`, `--name NAME`	This sets the name for this connector. In some cases, it is useful to have multiple connectors of the same type executing against a single TQ instance. For example, the Syslog Exporter can be run against multiple target and multiple exports, each with their own name and configuration.
`-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 (this happens by default).
`-d`, `--no-differential`	If exports are used in this connector, this will turn 'off' the differential flag for the execution. This allows debugging and testing to be done on export endpoints without having to rebuild the exports after the test. THIS SHOULD NEVER BE USED IN PRODUCTION.
`-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.)

ARGUMENT	DESCRIPTION
`-v {1,2,3}, --verbosity {1,2,3}`	This is the logging verbosity level. The default setting is **1** (Warning).
`-ep, --external-proxy`	This allows you to use the proxy that is specified in the ThreatQ UI. This specifies an internet facing proxy, NOT a proxy to the TQ instance.
`-i, --import`	N/A
`-e, --export`	N/A
`-pt PID_TIMEOUT, --pid-timeout PID_TIMEOUT`	N/A

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

Supported Observable Types

The following are supported observable (indicator) types from ServiceNow. If you would like a type supported and it is not listed, please contact ThreatQ support and we can add support for those types.

- IP Address (V4) -> IP Address
- Email address -> Email Address
- Unknown -> String
- SHA512 hash -> SHA-512
- Filename -> Filename
- File path -> File Path
- CIDR rule -> CIDR Block
- SHA1 hash -> SHA-1
- Registry key -> Registry Key
- Domain name -> FQDN
- MUTEX name -> Mutex
- URI -> URL Path
- SHA256 hash -> SHA-256
- MD5 hash -> MD5
- CVE number -> CVE
- URL -> URL
- Top-level domain name: FQDN

Change Log

- **Version 2.5.2 rev-a**
  - Guide Update - Updated ServiceNow compatability versions.
- **Version 2.5.2**
  - Added the ability to parse ServiceNow comments for IoCs. These IoCs can be ingested by ThreatQ and related to their corresponding event.
- **Version 2.5.1**
  - Added new configuration option: `Strip HTML from Comments`.
- **Version 2.5.0**
  - Added user fields to allow the user to choose whether to ingest observables, comments/work notes, or related incidents
  - Fixed dependency issues for configparser.
- **Version 2.4.0**
  - Added functionality to allow the program to be run in both Python 2 and Python 3.
  - Alphabetized the imports in all .py files.
  - Removed requests from requirements.
- **Version 2.3.2**
  - Fixed a bug that caused indicators without a type from ServiceNow to not be synced properly.
- **Version 2.3.1**
  - Added more logging.
  - Fixed issues with syncing unicode from ServiceNow to ThreatQ.
  - Added the ability to get a notification in the ThreatQ UI when a failure in the ThreatQ app has occurred.
- **Version 2.3.0**
  - Fixed issues that may arise with timezone props.
  - Improved comment handling.
  - Added pagination to fetch tickets.
  - Added PID timeout.
- **Version 2.2.0**
  - Fixed issue with duplicates and updating new Tasks.
  - Added support for Security Incident Response Tasks.
  - Fixed issue with invalid observable link.
  - Fixed issue where child incidents were not related to the parent object in ThreatQ.
  - Fixed formatting when adding comments from workflow actions.
  - Added support for “updating” security tags that change.
  - Added the ability to run import/export separately.
- **Version 2.1.2**
  - Fixed a bug that stopped data from being synced due to missing data.
  - Improved timezone handling and date formatting.
  - Fixed some instances of unicode characters not being synced to ThreatQ.
- **Version 2.1.1**

  - Fixed an issue where an empty observable created in ServiceNow would break the sync to ThreatQ.
- **Version 2.1.0**
  - Fixed several issues with syncing security annotations to ThreatQ.
  - Fixed several issues with creating comments in ThreatQ.
- **Version 2.0.2**
  - Removed need to use sys_journal_field since only admins can access it.
  - Fixed comment differential.
- **Version 2.0.1**
  - Created a fallback happened_at date so there are no crashes when sys_created_on is not available.
- **Version 2.0.0**
  - Sync Service Desk Incidents.
  - Sync Threat Intelligence (Threat Intelligence Plugin).
  - Sync Security Incidents (Security Incident Response Plugin).
  - Sync relationships between incidents/cases and observables.
  - Add related indicators and events from ThreatQ to ServiceNow.
- **Version 1.0.0**
  - Initial Release