

# ThreatQuotient



## Jira Connector User Guide

**Version 1.5.0**

October 18, 2023

**ThreatQuotient**

20130 Lakeview Center Plaza Suite 400  
Ashburn, VA 20147

 **ThreatQ Supported**

### **Support**

Email: [support@threatq.com](mailto:support@threatq.com)

Web: [support.threatq.com](https://support.threatq.com)

Phone: 703.574.9893

# Contents

<b>Warning and Disclaimer .....</b>	<b>3</b>
<b>Support .....</b>	<b>4</b>
<b>Integration Details.....</b>	<b>5</b>
<b>Introduction .....</b>	<b>6</b>
<b>Prerequisites .....</b>	<b>7</b>
Time Zone .....	7
Integration Dependencies .....	8
<b>Installation.....</b>	<b>9</b>
Creating a Python 3.6 Virtual Environment .....	9
Installing the Connector.....	10
<b>Configuration .....</b>	<b>12</b>
<b>Usage.....</b>	<b>14</b>
Command Line Arguments.....	14
CRON .....	16
<b>ThreatQ Project Configuration .....</b>	<b>17</b>
Importing the ThreatQ Project to JIRA .....	17
Install Project Configurator Add-on to JIRA.....	17
Creating the ThreatQ Project.....	19
Adding ThreatQ Event and Indicator Queues .....	21
Using the ThreatQ Project .....	23
Creating Events .....	23
Creating Indicators .....	25
Editing the ThreatQ Project .....	27
<b>Known Issues / Limitations .....</b>	<b>28</b>
<b>Change Log .....</b>	<b>29</b>

# 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 © 2023 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](mailto: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	1.5.0
-----------------------------	-------

Compatible with ThreatQ Versions	>= 4.34.0
-------------------------------------	-----------

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

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

# Introduction

ThreatQ's JIRA Integration executes a sync between a JIRA instance and ThreatQ instance.

---

# 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
```

## 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	>=1.8.1	N/A
threatqcc	>=1.1.1	N/A
pytz	>=2017.2	N/A
python-dateutil	>=2.6.0	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 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/tqenv/<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_jira-<version>-py3-none-any.whl
```



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

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/tqenv/<environment_name>/bin/tq-conn-jira -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-jira -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
JIRA Host	The FQDN of your JIRA instance. Make sure it includes either http:// or https://.
Username	The JIRA username you want to use for the integration
Password	The JIRA password associated with the username above.
JIRA Project	the key of the project you want the integration to sync with. Make sure you type the key and not the name. The key will be all caps.
ThreatQ Event Types	<p>The event types you want to sync with JIRA.</p> <p>You can choose <b>All</b>, which will sync all ThreatQ event types. You can also choose <b>None</b> which will make it so ThreatQ events will not be added to JIRA. You can also select individual or list of event types to sync..</p>
JIRA Event Issue Type	<p>The name of the issue type you want the ThreatQ events to be synced with.</p> <p>This field does not apply if you install the ThreatQ Project configuration. In that instance, leave this field as is.</p>
JIRA Indicator Issue Type	The name of the issue type you want the ThreatQ indicators to be synced with.

PARAMETER	DESCRIPTION
	This field does not apply if you install the ThreatQ Project configuration. In that instance, leave this field as is.
<b>Historical Timeframe (days)</b>	<p>The number of days to pull historical data from (for the first run).</p> <p>This option only applies to the first time you run the integration. For ThreatQ events, it compares against the <b>Happened At</b> date. For JIRA issues, it will compare against the <b>Updated</b> date.</p>
<b>ThreatQ Host</b>	The hostname for your ThreatQ instance. No URL path or scheme is required.

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-jira -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>	Shows this help message and exits.
<code>-n NAME, --name NAME</code>	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
<code>-d, --no-differential</code>	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.
<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. The default is 1 (Warning).

---

ARGUMENT	DESCRIPTION
<code>--external-proxy, -ep</code>	This enables a proxy to be used to contact the internet for the data required by this connector. This specifies an internet facing proxy, NOT a proxy to the TQ instance.

## 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-jira -c /etc/tq_labs/ -ll /var/log/tq_labs/ -v3
```

4. Save and exit CRON.



# ThreatQ Project Configuration

This section provides setup steps and instruction for users that opt to install the ThreatQ project configuration.

This section provides setup steps and instruction for users that opt to install the ThreatQ project configuration.

The configuration creates two new issue types (Event and Indicator), and they will be synced with ThreatQ's Event and Indicator types.

If you do not use the ThreatQ Project configuration, there is no more configuration required to use the integration. Verify that the timezone of the JIRA instance aligns with the timezone of the machine the integration is running on.

## Importing the ThreatQ Project to JIRA

The first thing you will want to do is import the ThreatQ Project into your JIRA instance. This will bring in all of the ThreatQ fields, screens, and issue types.

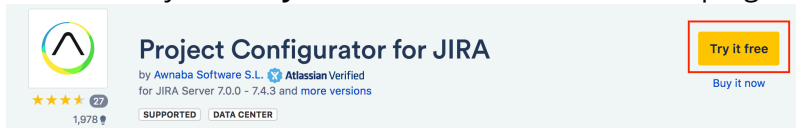
### Install Project Configurator Add-on to JIRA

The first step to import the project is to install the Project Configurator add-on from the JIRA marketplace.

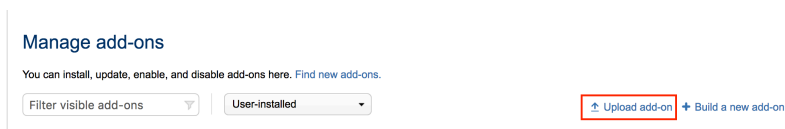
1. Navigate to the Project Configurator add-on for JIRA:

<https://marketplace.atlassian.com/apps/1211147/project-configurator-for-jira?hosting=server&tab=overview>

2. Click on the yellow **Try it free** button located to the top right.



3. Accept the license agreement.
4. Click the **Download** button to download the Project Configurator jar file.
5. Open up your JIRA instance.
6. Click the **Gear Icon** in the top right, and select the **Add-ons** menu option.
7. In the left menu under the **Atlassian Marketplace** tab, click **Manage add-ons**
8. Click the **Upload add-on** link



A popup should show where you can select a file from your computer

9. Click **Choose File** and select the jar file you downloaded in *Step 4*. Then click **Upload** to install the add-on



This may take a few minutes. Refresh the page if the add-on is still installing after 5 minutes.

10. Expand the **Project Configurator** entry in the **User-installed add-ons** list.
11. Click the **Free trial** button and accept the license agreement



You should now see a **Project Configurator** tab in the left menu of the **Add-ons** page. Refresh the page if you do not see it.

## Administration Search

[Applications](#)
[Projects](#)
[Issues](#)

### ATLASSIAN MARKETPLACE

**Find new add-ons**

[Manage add-ons](#)

### PROJECT CONFIGURATOR

[Import project configuration](#)

[Import complete project](#)

[Export all projects](#)

[Export selected projects](#)

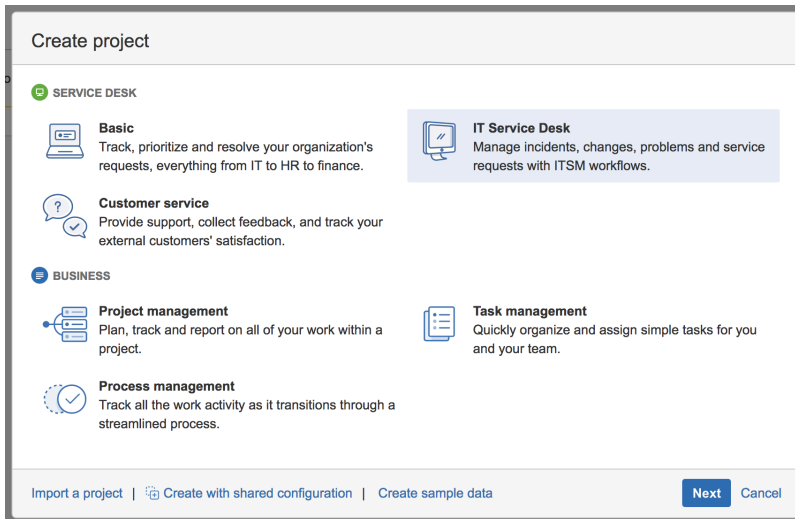
[Import conflict detection](#)

["Used by" report](#)

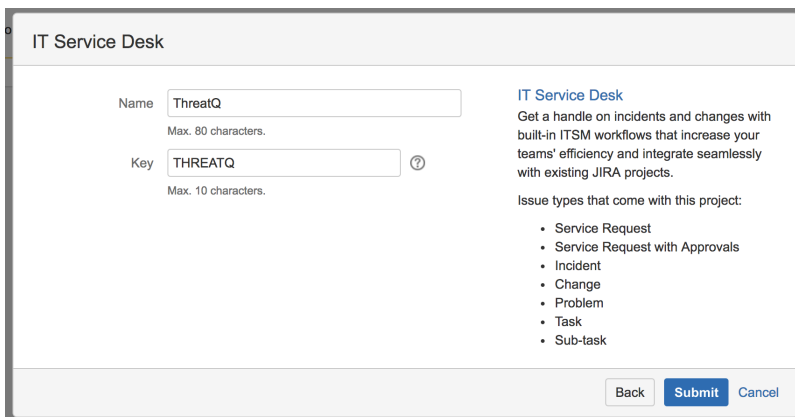
## Creating the ThreatQ Project

This section will detail the installation process for adding the ThreatQ Project to JIRA

1. Click the **Projects** dropdown in the JIRA navigation bar, and select the **Create Project** item
2. Select **IT Service Desk** and click **Next**.



3. For name, enter "**ThreatQ**", and for the key, enter "**THREATQ**". Then click the **Submit** button  
This will create the ThreatQ base project



4. Next, the ThreatQ Project must be imported using the Project Configurator add-on you previously installed. Go back to the Add-ons page in your JIRA settings.
5. Click the **Import project configuration** link in the **Project Configurator** tab in the left menu
6. Click the **Choose File** button and select the **ThreatQ Project Configuration for JIRA.xml** file
7. Check the **Apply Changes?** and **Smart custom fields contexts** fields and click **Import project configuration**



This will start the import process.

If there are any errors, please forward them to your ThreatQ representative If there

are any warnings, they should be ignorable.

If a warning has to do with 'THREATQ: Event' or 'THREATQ: Indicator', forward it to your ThreatQ representative.

#### ATLASSIAN MARKETPLACE

[Find new add-ons](#)

[Manage add-ons](#)

#### PROJECT CONFIGURATOR

**Import project configuration**

[Import complete project](#)

[Export all projects](#)

[Export selected projects](#)

[Import conflict detection](#)

["Used by" report](#)

## Import project configuration



### Important for first time users!

Please read these [warnings](#) before importing

Project  ThreatQ Project for JIRA.xml  
Configuration File

☒ **Apply changes?**

When ticked, configuration changes will be applied to JIRA

☐ **Create other projects?**

When ticked, load will create other projects needed by custom field configuration contexts.

☒ **Smart custom field contexts**

When ticked, load will change only those custom field configuration contexts related to projects being imported

☐ **Try to publish drafts**

Try to publish automatically workflow drafts and workflow scheme drafts created during the import

☐ **Continue on errors found in dashboards and filters**

When ticked, load will not stop when an error is found importing a dashboard or filter

Do not load:

Projects (changes)

Project specific

Versions

Components

Role members

Global

Users

Groups

Selected object types will not be created or modified in the load

8. The custom configuration should now be in your ThreatQ Project

## Adding ThreatQ Event and Indicator Queues

The project configurator add-on does not support exporting/importing queues. You can manually create queues from the ThreatQ Project. Adding the additional queues are for the most part, optional, but adding them will separate them from other issues that may be created.

1. In the JIRA navigation bar, click the **Projects** dropdown and select the **ThreatQ Project**



If you are not already on the **Queues** page, go to it. It is the 1st icon in the left sidebar

2. In the **Queues** tab, there should be a + **New queue** button at the bottom of the list, click that to add a queue.

QUEUES	
⋮ All open	1
⋮ Unassigned issues	1
⋮ Assigned to me	0
⋮ ↪ Waiting on me	0
⋮ Incidents	0
⋮ ↪ Reported in the l...	0
⋮ ↪ Critical	0
⋮ Service requests	1
⋮ ↪ Due in 24h	0
⋮ Change	0
⋮ ↪ Ready for imple...	0
⋮ ↪ Emergency change	0
⋮ Problem	0
⋮ ↪ Completed last 3...	0
⋮ Recently resolved	0
+ New queue	

3. The first queue we want to add is the **ThreatQ Events** queue, so for the name field, enter **ThreatQ Events**
4. For the **Issues to show** field, set the **Type** to **Event**, and leave the rest set to **All**
5. For the **Columns** field, you can choose any you'd like, however, we recommend the following columns (in order):
  - Key
  - Summary
  - Event Type
  - Happened At
  - Linked Issues
  - ThreatQ Link
  - Priority
  - Created

New queue

Name  
**ThreatQ Events**

Issues to show  
More ▾ Event ▾ Status: All ▾ Label: All ▾ Order by ▾ Advanced

Columns  
More ▾ Key ⓘ Summary ⓘ Event Type ⓘ Happened At ⓘ Linked Issues ⓘ ThreatQ Link ⓘ Priority ⓘ Created ⓘ

Create Cancel

6. Click **Create** to create the Queue
7. The second queue we want to add is the **ThreatQ Indicators** queue, so click the **+ New queue** button again and enter **ThreatQ Indicators** for the name field
8. For the **Issues to show** field, set the **Type** to **Indicator**, and leave the rest set to **All**
9. For the **Columns** field, you can choose any you'd like, however, we recommend the following columns (in order):
  - Key
  - Summary
  - Indicator Type
  - Indicator Status
  - Score
  - Linked Issues
  - ThreatQ Link
  - Created

New queue

Name  
**ThreatQ Indicators**

Issues to show  
More ▾ Indicator ▾ Status: All ▾ Label: All ▾ Order by ▾ Advanced

Columns  
More ▾ Key ⓘ Summary ⓘ Indicator Type ⓘ Indicator Status ⓘ Priority ⓘ Score ⓘ ThreatQ Link ⓘ Created ⓘ Linked Issues ⓘ

Create Cancel

10. Click **Create** to create the Queue
- The Queues are now fully setup

## Using the ThreatQ Project

This section will go over how to create events and indicators as well as editing the fields.

### Creating Events

Creating an event is simple, and events can even be linked to indicators.

1. Click the **Create** button in the JIRA navigation bar



This should trigger a popup to create an issue

2. For the **Project** dropdown, select the **ThreatQ** project
3. For the **Issue Type** dropdown, select **Event**. Click on **Next** if required.
4. Fill out the fields:

FIELD	DESCRIPTION
Event Type	The type of event, corresponding with the ThreatQ event types.
Priority	This field will be synced as an attribute in ThreatQ.
Summary	This will be the title of the event.
Description	The description of the event.
Happened At	The date/time the event occurred.
Labels	These will be synced as attributes in ThreatQ.
Linked Issues	If you have a created indicator or event already, it can be linked using this field. Select <b>Relates to</b> from the dropdown.
ThreatQ Link	Do not edit this field. It will be overridden by ThreatQ when synced.

Create Issue

Configure Fields

Project ThreatQ (THREATQ)

Issue Type Event

Event Type Incident

Priority Lowest

Summary Event title here

Description

Style B I U A A

Event description here

Visual Text

Happened At 28/Aug/17 02:50 PM

Labels False-positive

Linked Issues blocks

Issue

ThreatQ Link Not Synced

Create another Create Cancel

- Click the **Create** button
- Once created, you can view the issue from the queue you created in the previous section.
- If you would like to add a linked issue after creating the issue, you will need to open the issue, then click the **Edit** button



## Creating Indicators

Creating and indicator is simple, and indicators can even be linked to events.

1. Click the **Create** button in the JIRA navigation bar



This should trigger a popup to create an issue

2. For the **Project** dropdown, select the **ThreatQ** project
3. For the **Issue Type** dropdown, select **Indicator**. Click **Next** if required.
4. Fill out the fields:

FIELD	DESCRIPTION
<b>Indicator Type</b>	The type of indicator, corresponding with the ThreatQ event types.
<b>Priority</b>	This field will be synced as an attribute in ThreatQ.
<b>Summary</b>	This will be the title of the indicator.
<b>Indicator Status</b>	The status of the indicator, corresponding with the ThreatQ Status.
<b>Score</b>	The score of the indicator, corresponding with ThreatQ's Scoring System.
<b>Labels</b>	These will be synced as attributes in ThreatQ.
<b>Linked Issues</b>	If you have a created indicator or event already, it can be linked using this field. Select <b>Relates to</b> from the dropdown.
<b>ThreatQ Link</b>	Do not edit this field. It will be overridden by ThreatQ when synced.

Create Issue

Configure Fields

Project

ThreatQ (THREATQ)

Issue Type

Indicator

Indicator Type

IP Address

The type of indicator

Priority

Lowest

Summary

1.1.1.1

Indicator Status

Whitelisted

Status corresponding to ThreatQ's Indicator Status

Score

0

Score corresponding to ThreatQ's Score attribute

Labels

Begin typing to find and create labels or press down to select a suggested label.

Linked Issues

relates to

Issue

THREATQ-2

Begin typing to search for issues to link. If you leave it blank, no link will be made.

ThreatQ Link

Not Synced

The link to the associated ThreatQ object (ThreatQ will fill this out)

Create another

Create

Cancel

5. Click the **Create** button
6. Once created, you can view the issue from the queue you created in the previous section.
7. If you would like to add a linked issue after creating the issue, you will need to open the issue, then click the **Edit** button

## Editing the ThreatQ Project

If you would like, you can edit the ThreatQ Project, however, it is advised that you do not remove any of the default configurations that were imported. This may cause issues with the integration. You may add fields to the screens or field configurations, however, they will not be synced with the Integration. They will be only there for your company's viewing purposes.

---

## Known Issues / Limitations

- Importing a project configuration **only works** for **JIRA Servers** (on-premise)
- If you are using **JIRA Cloud**, you **will not** be able to import the ThreatQ Project into JIRA. JIRA Cloud does not allow the installation of any Project Configuration Managers such as Project Configurator or Configuration Manager. Instead, this will be able to sync with a generic JIRA project.
- Make sure that your JIRA instance's 'Default user time zone' setting is set to the same timezone as machine's timezone that the integration will run on. If you do not align the time zones, issues may be synced multiple times. In addition, depending on the timezone difference, issues may be skipped.
- In order to use this integration, you **must** have the JIRA **IT Service Desk** application installed

---

# Change Log

- **Version 1.5.0**
  - Added python 3 support.
  - Removed the ability for JIRA to set the score for an IOC in ThreatQ.
  - Fixed an issue where users were unable to select None as the score.
  - IOCs can now be synced in ThreatQ only if they originated from JIRA.
- **Version 1.3.0**
  - Added the ability to disable syncing from ThreatQ to JIRA. This requires the removal of the old configuration file to work. Users will still be able to sync from JIRA to ThreatQ.
  - Improved support for Event syncing in terms of preventing duplicates.
  - Improved support for unicode/ascii.
  - Fixed a subtype issue with JIRA.
  - Fixed timezone issues.
  - JIRA is now listed as the source when indicators are synced.
- **Version 1.2.2**
  - Added the ability to change status via UI.
- **Version 1.2.1**
  - flake8 validation
- **Version 1.2.0**
  - Added support for syncing indicators if a specified issue type.
- **Version 1.1.0**
  - Added the ability to sync with generic JIRA projects
  - Added UI option to sync with specific JIRA issue types.
  - Fixed duplicate linked issue bug.
- **Version 1.0.0**
  - Initial Release