ThreatQuotient for MISP Export

June 25, 2018

Version 1.0

11400 Commerce Park Dr
Suite 200,
Reston, VA
20191, USA
https://www.threatq.com/
Support: support@threatq.com
Sales: sales@threatq.com

Contents

List of Figures and Tables

About This ThreatQuotient for MISP Export

Author	ThreatQuotient Professional Services

Document Conventions

📝	Alerts readers to take note. Notes contain suggestions or references to material not covered in the document.
⚠	Alerts readers to be careful. In this situation, you may do something that could result in equipment damage or loss of data.
⏰	Alerts the reader that they could save time by performing the action described in the paragraph.
ℹ	Alerts the reader that the information could help them solve a problem. The information might not be troubleshooting or even an action.

1 Introduction

1.1 Application Function

The ThreatQuotient for MISP Export is a uni-directional connector which creates an MISP Event and stores indicators of compromise (IoCs) in the MISP Event for the current day. If there are more than 25,000 IoCs to be exported on a given day, the code creates a new MISP Event for each 25,000 IoC collection.

In order to store 25,000 indicators, the MISP server's php.ini file must be modified.
Refer to Table 2: PHP Configuration Information below for more information.

1.2 Preface

This guide is to provide the information necessary to implement the ThreatQuotient for MISP Export. This document is not specifically intended as a site reference guide.

It is assumed that the implementation engineer has experience installing and commissioning ThreatQuotient Apps and integrations covered within the document, as well as experience necessary to troubleshoot at a basic level.

1.3 Audience

This document is intended for use by the following parties:

1. ThreatQ and Security Engineers.
2. ThreatQuotient Professional Services Project Team and Engineers.

1.4 Scope

This document covers the implementation of the application only.

Table 1: ThreatQuotient Software & App Version Information

Software/App Name	File Name	Version
ThreatQ	Version 3.6.x or greater	
ThreatQuotient for MISP Export	1.0.5	

1.5 Assumptions

The following criteria is assumed to be in place and functional to allow the implementation of the ThreatQuotient for MISP Export into the managed estate:

- All ThreatQuotient equipment is online and in service.
- Infrastructure/transmission at all sites and between sites is in place to support the network traffic.
- All required firewall ports have been opened.
- All equipment is powered from permanent power supplies.
- A clock source of sufficient accuracy is connected to the network and the network and devices are using it as the primary clock source.

2 Implementation Overview

This document explains how to install the ThreatQuotient for MISP Export into ThreatQ.

2.1 Prerequisites

Throughout this implementation document, we will refer to several files and directories, some of which will be symbolic, and others may change depending on specifics of the environmental setup.

2.1.1 PHP Configuration

The following are the minimum settings to allow for MISP to store ThreatQ indicators:

Table 2: PHP Configuration Information

Variable	Value
max_execution_time	300
memory_limit	512M
upload_max_filesize	50M
post_max_size	50M

Ensure all ThreatQ devices are set to the correct time, time zone and date, and using a stable clock source.

For Example:

Figure 1: Time Zone Change Example

```
sudo ln -sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
```

2.2 Security and Privacy

For ThreatQuotient Professional Services engineers to configure the system, local network access is required to connect to the managed estate. Therefore, the implementation must occur at an office or data center location.

Passwords have not been provided in this document. Please contact your project team for this information, if required.

All engineers are reminded that all data belonging and pertaining to the business is strictly confidential and should not be disclosed to any unauthorized parties.

The data held within this document is classed as confidential due to its nature.

3 ThreatQuotient for MISP Export Installation

3.1 Setting up the Integration

Ensure the file `tqMispExport-1.0.5-py2-none-any.whl` has been added to the ThreatQ instance or has an internet connection using yum credentials.

1. Install the .whl file using the following command.

Figure 2: PIP Installing the Application (Inc Example Output)

```
[root@localhost ~]# pip install -i
https://<USERNAME>:<PASSWORD>@extensions.threatq.com/threatq/integrations
tqMispExport
You are using pip version 7.1.0, however version 9.0.3 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
Collecting tqMispExport
  Downloading https://extensions.threatq.com/threatq/integrations-
dev/+f/466/1f226f831d57f/tqMispExport-1.0.5-py2-none-any.whl
Requirement already satisfied (use --upgrade to upgrade): threatqcc>=1.1.3 in
/usr/lib/python2.7/site-packages (from tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): pymisp in
/usr/lib/python2.7/site-packages (from tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): threatqsdk>=1.5.1 in
/usr/lib/python2.7/site-packages (from tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): requests in
/usr/lib/python2.7/site-packages (from tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): jinja2==2.8 in
/usr/lib/python2.7/site-packages (from threatqcc>=1.1.3->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): six in
/usr/lib/python2.7/site-packages (from pymisp->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): python-dateutil in
/usr/lib/python2.7/site-packages (from pymisp->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): jsonschema in
/usr/lib/python2.7/site-packages (from pymisp->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): setuptools>=36.4 in
/usr/lib/python2.7/site-packages (from pymisp->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): urllib3<1.23,>=1.21.1 in
/usr/lib/python2.7/site-packages (from requests->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): certifi>=2017.4.17 in
/usr/lib/python2.7/site-packages (from requests->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): chardet<3.1.0,>=3.0.2 in
/usr/lib/python2.7/site-packages (from requests->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): idna<2.7,>=2.5 in
/usr/lib/python2.7/site-packages (from requests->tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): MarkupSafe in
/usr/lib64/python2.7/site-packages (from jinja2==2.8->threatqcc>=1.1.3-
>tqMispExport)
Requirement already satisfied (use --upgrade to upgrade): functools32 in
/usr/lib/python2.7/site-packages (from jsonschema->pymisp->tqMispExport)
Installing collected packages: tqMispExport
Successfully installed tqMispExport-1.0.5
[root@localhost ~]#
```

Once the application has been installed, a directory structure must be created for all configuration, logs, and files, using the `mkdir` command. See example below:

Figure 3: Creating Integration Directories Example

```
mkdir -p /opt/tq-integrations/misp/
mkdir -p /opt/tq-integrations/misp/config
mkdir -p /opt/tq-integrations/misp/logs
mkdir -p /opt/tq-integrations/misp/data
cd /opt/tq-integrations/misp/
```

This package comes with a driver called **`tq-misp-export`**. After installing with pip, a script stub will appear in `/usr//bin/tq-misp-export`.

2. Issue the following commands to initialize the integration.

Figure 4: Initial Running of the Integration (Inc Example Output)

```
root@localhost ~]#tq-misp-export -v 3 -ll stdout
WARNING [abstract.py:19 - <module>() ] You're using python 2, it is strongly
recommended to use python >=3.5
WARNING [mispevent.py:26 - <module>() ] You're using python 2, it is strongly
recommended to use python >=3.5
WARNING [api.py:31 - <module>() ] You're using python 2, it is strongly recommended
to use python >=3.5
XXXX-XX-XX XX:XX:XX - tqMispExport.tq_driver DEBUG: Set name of connector to MISP
Export.
XXXX-XX-XX XX:XX:XX - tqMispExport.tq_driver DEBUG: Successfully parsed command
line arguments.
XXXX-XX-XX XX:XX:XX - threatqcc.custom_connector DEBUG: Using Current working
directory for config path
ThreatQ Host: 192.168.1.182
Connector Name: MISP Export
Client ID: c630f44fcefe84aeb22e2c62e2ab7573
E-Mail Address: misp@domain.com
Password:
Connector configured.  Set information in UI.
XXXX-XX-XX XX:XX:XX - tqMispExport.tq_driver INFO: Further configuration is
required in the UI.
[root@localhost ~]#
```

The driver will run once, where it connects to the ThreatQ instance and installs the user interface component of the connector.

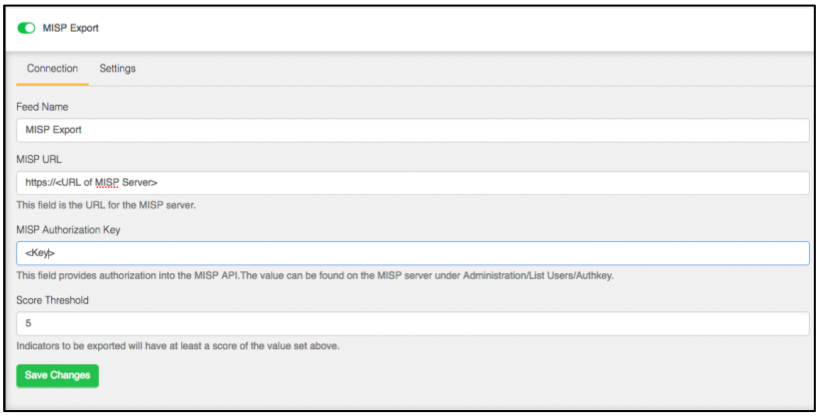
3.2 Configuring the connector

To edit the configuration, go to the **Incoming Feeds** page within ThreatQ, click the **ThreatQ Labs** tab, then expand the Feed Settings for the **MISP Export** section.

1. The following information must be entered as described below.
   - **Feed name**: Name of the **MISP Export** feed.
   - **MISP URL**: The URL for the applicable MISP SIEM.
   - **MISP Authorization Key**: This field provides authorization into the MISP API. The value can be found on the MISP server under "Administration/List Users/Authkey."
   - **Score Threshold**: Active indicators to be exported will have at least a score of the value set in this field. The default is 0.

An invalid value will cause an error to occur upon execution. The log file for this should periodically be checked to ensure that the application has successfully run.

Figure 5: ThreatQ MISP Export UI Configuration



Feed Name

MISP Export

MISP URL

https://<URL of MISP Server>

This field is the URL for the MISP server.

MISP Authorization Key

<Key>

This field provides authorization into the MISP API.The value can be found on the MISP server under Administration/List Users/Authkey.

Score Threshold

5

Indicators to be exported will have at least a score of the value set above.

The running of the application with the `-ds` switch disables the requirement for SSL.

Figure 6: ThreatQ MISP Export Manual Running (Inc Example Output)

```
[root@localhost ~]# tq-misp-export -v 3 -ll stdout -ds
WARNING [abstract.py:19 - <module>() ] You're using python 2, it is strongly
recommended to use python >=3.5
WARNING [mispevent.py:26 - <module>() ] You're using python 2, it is strongly
recommended to use python >=3.5
WARNING [api.py:31 - <module>() ] You're using python 2, it is strongly recommended
to use python >=3.5
2018-04-09 13:57:40 - tqMispExport.tq_driver DEBUG: Set name of connector to MISP
Export.
2018-04-09 13:57:40 - tqMispExport.tq_driver DEBUG: MISP SSL Verify is set to
False.
2018-04-09 13:57:40 - tqMispExport.tq_driver DEBUG: Successfully parsed command
line arguments.
2018-04-09 13:57:40 - threatqcc.custom_connector DEBUG: Using Current working
directory for config path
2018-04-09 13:57:40 - tqMispExport.tq_driver DEBUG: Private Connection Established.
Beginning execution of the MISP Export Connector.
2018-04-09 13:57:45 - tqMispExport.tq_driver DEBUG: Establishing MISP
communications.
2018-04-09 13:59:02 - tqMispExport.tq_driver DEBUG: Processing [9682] indicators.
2018-04-09 13:59:02 - tqMispExport.tq_driver DEBUG: Creating Event in MISP.
2018-04-09 14:02:03 - tqMispExport.tq_driver DEBUG: Event [5230] created in MISP.
2018-04-09 14:02:03 - tqMispExport.tq_driver DEBUG: Purged processed indicators -
[0] indicators remaining.
```

```
Completed execution of the MISP Export Connector in 262 seconds
[root@localhost ~]#
```

3.3 CRON

To run this script on a reoccurring basis, use CRON or some other system schedule. The argument in the cron script ***must*** specify the config and log locations.

This can be run multiple times a day and should not be run more often than once per hour.

3.3.1 Setting Up the CRONJOB

1. Login via a CLI terminal session to the ThreatQ instance.
2. Input the commands below.

   Figure 7: Command Line Crontab Command

   ```
   $> crontab -e
   ```

   This will enable the editing of the crontab, using vi.

   Depending on how often the cronjob is required to run, it will need to be adjusted to a time to suit the environment.
3. Input the commands below – this example shows every **4 Hours.**

   Figure 8: Command Line Crontab tq-misp-export Command

   ```
   0 */4 * * * tq-misp-export -v 3 -ll stdout -ds
   ```

   To run this script on a reoccurring basis, use CRON or some other on system schedule. CRON is shown below.

   The argument in the CRON script ***must*** specify the config and log locations.

   This can be run multiple times a day and should **not** be run more often than once per hr.

For further reference, see the ThreatQ Help Center.

Appendix A: Supplementary Information

Uninstalling the Connector

```
sudo pip uninstall tq-misp-export
```

Driver command line options

The tq-misp-export driver has several command line arguments that will help execute the application. These options can be displayed by executing `/usr/bin/tq-misp-export --help.`

```
usage: tq-misp-export Connector [-h] [-ll LOGLOCATION][-c CONFIG] [-v VERBOSITY]
```

optional arguments:

```
  -h, --help
```

Shows the help message and exit.

```
  -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).

```
  -c CONFIG, --config CONFIG
```

This is the location of the configuration file for the connector. This location must have read and write permissions for the current user. If no config file is given, the current directory will be used. This file is also where some information from each run of the connector may be put (e.g. last run time, private OAuth, etc).

```
  -v {1,2,3}, --verbosity {1,2,3}
```

This is the logging verbosity level. The Default is 1 (Warning).

Trademarks and Disclaimers

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR THREATQUOTIENT REPRESENTATIVE FOR A COPY.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THIRD PARTY SUPPLIERS ARE PROVIDED “AS IS” WITH ALL FAULTS. THREATQUOTIENT AND THIRD-PARTY SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL THREATQUOTIENT OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF THREATQUOTIENT OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

ThreatQuotient and the ThreatQuotient Logo are trademarks of ThreatQuotient, Inc. and/or its affiliates in the U.S. and other countries.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.