ThreatQuotient

ThreatQuotient for Carbon Black Response Connector Guide

Version 1.0.0

Thursday, February 20, 2020

ThreatQuotient

11400 Commerce Park Dr., Suite 200

Reston, VA 20191

Support

Email: support@threatq.com

Web: support.threatq.com

Phone: 703.574.9893

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.

Last Updated: Thursday, February 20, 2020

Contents

ThreatQuotient for Carbon Black Response Connector Guide **1**

Warning and Disclaimer **2**

Contents **3**

Introduction **4**

Preface 4

Audience 4

Scope 4

Versioning 5

Installation **6**

Configuration **7**

ThreatQ Instance 7

Carbon Black Response Instance 12

Upgrading the Connector **17**

Introduction

The Carbon Black Response Connector for ThreatQ allows a user to ingest a Threat Library saved search into Carbon Black Response as a Threat Report.

Preface

This guide is to provide the information necessary to implement the ThreatQuotient for Carbon Black Response Connector. This document is not specifically intended to form 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.

Audience

This document is intended for use by the following parties:

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

Scope

This document covers the implementation of the Carbon Black Response Connector for ThreatQuotient only.

Versioning

Software/App	Version
ThreatQ	v4.25 or greater
ThreatQuotient for Carbon Black Response Connector	v1.0.0

Installation

The Carbon Black Response Connector is packaged into an RPM file which will be installed on your Carbon Black Response server.

1. Download the RPM file from the ThreatQuotient Download Repository:

   https://download.threatq.com/integrations/python-cb-threatq-connector-1.0.0-10.x86_64.rpm

   > Follow the next several steps to install the connector onto your Carbon Black Response Instance.

2. Transfer the RPM file to your Carbon Black Response instance.

```
scp python-cb-threatq-connector-1.0.0-10.x86_
64.rpm root@<cb-ip-address>:/tmp/
```

3. SSH into your Carbon Black Response instance.
4. Install the package using RPM.

```
cd /tmp
rpm -ivh python-cb-threatq-connector-1.0.0-
10.x86_64.rpm --ignoreos --nofiledigest
```

Configuration

You will need to configure the connector in 2 locations. The first location is on your ThreatQ Instance by creating exports for the connector. The second is on your Carbon Black Response instance by editing the connector config file.

ThreatQ Instance

The connector can handle multiple exports. Each export will be imported to Carbon Black Response as a report. Exports are customizable and it is recommended that you create exports that pertain to your instance and ecosystem.

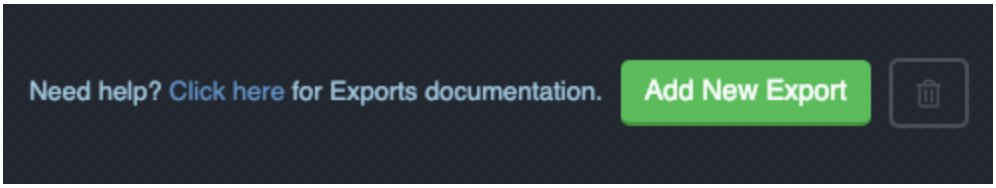
The steps below detail how to create the basic export configuration. You can change it according to your needs.

In your ThreatQ instance:

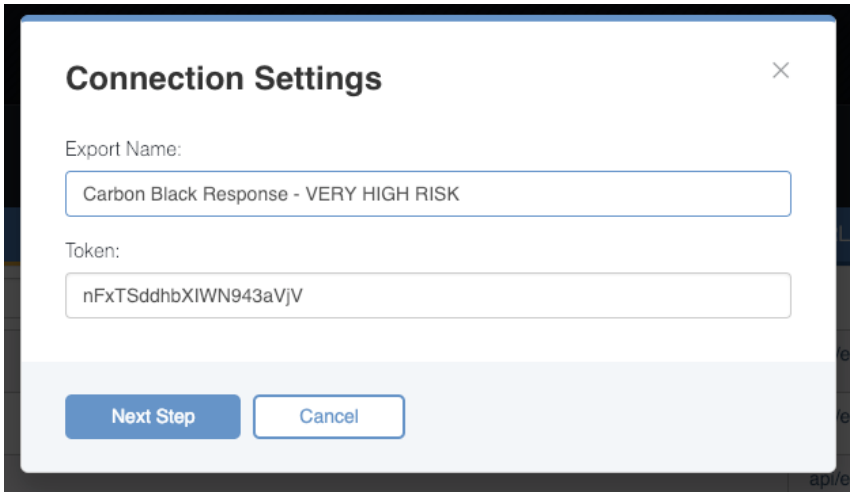
1. Navigate to **Settings > Exports**.

   The Exports page appears with a table listing all exports in alphabetical order.

2. Click on the **Add New Export** button.



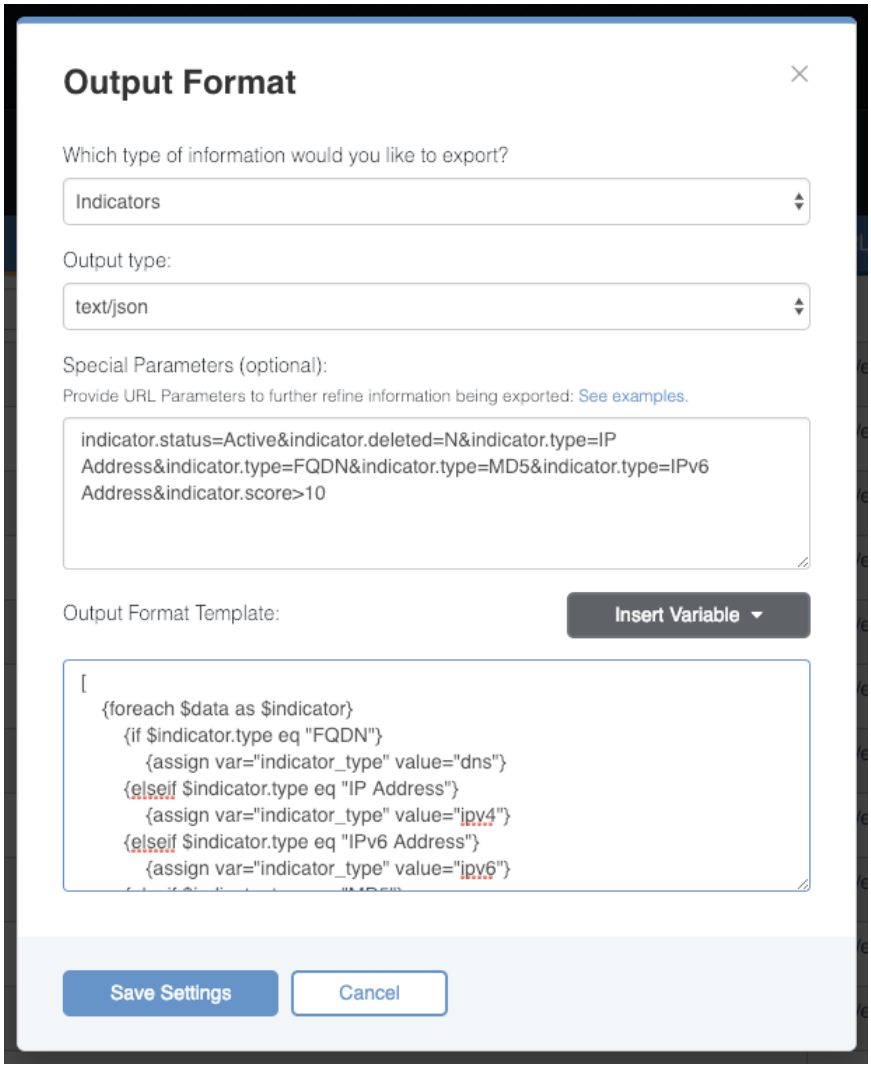
The Connection Settings dialog box opens.



3. Enter the Export name and click **Next**.

> Name your export according to the special parameters that you set. This example export screenshot above will be for indicators with a score greater than 10.

The Output Format dialog box will load.



4. Complete the Output Format dialog fields:

Field	Entry
Which type of information would you like to export?	Indicators
Output Type	text/json

5. Enter your Special Parameters:

```
indicator.status=Active&indicator.deleted=N&ind
icator.type=IP
Address&indicator.type=FQDN&indicator.type=MD5&
indicator.type=IPv6
Address&indicator.score=>=10
```

This example export will pull all Active indicators with a score of 10 or greater, that are either an IP Address, IPv6 Address, FQDN, or MD5

The Output Format Template will look like the following.

You do not need to update this field.

```
[
  {foreach $data as $indicator}
    {if $indicator.type eq "FQDN"}
        {assign var="indicator_type" value="dns"}
      {elseif $indicator.type eq "IP Address"}
        {assign var="indicator_type" value="ipv4"}
      {elseif $indicator.type eq "IPv6 Address"}
        {assign var="indicator_type" value="ipv6"}
      {elseif $indicator.type eq "MD5"}
        {assign var="indicator_type" value="md5"}
      {else}
        {assign var="indicator_type" value=""}
      {/if}
```

```
        {if $indicator.score gt 10}
          {assign var="indicator_score" value="100"}
        {else}
           {assign var="indicator_score" value-
e=$indicator.score*10}
        {/if}
        {if $indicator_type ne ""}
          {ldelim}
                "id":"{$indicator.id}{$smarty.now}",
                "timestamp":{$smarty.now},
                "link":"https://{$http_host}/indicators/{$in-
dicator.id}/details",
                "title":"Indicators from ThreatQ",
                "score":{$indicator_score},
                "iocs":{ldelim}
                        "{$indicator_type}":[
                                "{$indicator.value}"
                                ]
                        }
                }
                {if $indicator.last == false},{/if}
        {/if}
{/foreach}
]
```

6. Click on **Save Settings**.

Carbon Black Response Instance

After creating your exports in ThreatQ, the connector will now need to be configured on the Carbon Black Response instance. You will need to create a default "credentials.response" file for the connector to use and then configure the connector via the configuration file found at **/etc/cb/integrations/threatq/connector.conf**.

1. SSH into your Carbon Black Response instance.
2. Create your **credentials.response** file.

   You can skip this step if you have already created file.

   ```
   mkdir /etc/carbonblack
   touch /etc/carbonblack/credentials.response
   vi /etc/carbonblack/credentials.response
   ```

   Credenitals.Reponse

   ```
   [default]
   url=https://localhost
   token=xxxxxxxx
   ssl_verify=False
   ```

   Enter the token found in your Profile on Carbon Black Response's Web Portal for the **Token** field.

3. Copy the example ThreatQ connector configuration.

   ```
   cp
   ```

```
/etc/cb/integrations/threatq/connector.conf.exa
mple
/etc/cb/integrations/threatq/connector.conf
```

4. Edit the new configuration file.

```
vi /etc/cb/integrations/threatq/connector.conf
```

Connector.conf

```
[auth]
#----------------------------------#
ThreatQ API configuration
#----------------------------------
# This section allows global configuration options to be
passed to the ThreatQ feed.

threatq_host=https://<your threatq host>

# You can specify multiple exports to pull from. Simply
list them as a comme-delimited list
# Make sure that the tokens are aligned with the IDs
# Example:
# threatq_export_tokens=export_token_1,export_token_2,ex-
port_token_3
# threatq_export_ids=export_id_1,export_id_2,export_id_3
# threatq_export_titles=VERY HIGH RISK Indicators,HIGH
RISK Indicators,MEDIUM RISK Indicators
threatq_export_tokens=
```

```
threatq_export_ids=
threatq_export_titles=

threatq_verify_ssl=false
threatq_http_proxy=
threatq_https_proxy=

[bridge]
#-----------------------------------------------------
       #    Core Configuration
#-----------------------------------------------------

listener_port=6300
listener_address=127.0.0.1
feed_retrieval_minutes=60
#debug=1

# API key for an administrative user of the Carbon Black
server
carbonblack_server_token=
carbonblack_server_sslverify=false

# Only uncomment out the carbonblack_server_url if you are
running the connector on a machine
#  *other* than the Cb server itself.
# carbonblack_server_url=

# If you need to use an HTTPS proxy to access the iSIGHT
```

```
API server, uncomment and configure the https_proxy
# variable below.
# https_proxy-
y=http://proxyuser:proxypass@proxyhostname:proxyport
```

Field	Details
threatq_host	**Example:** https://<threatq-host>
threatq_ export_ids	• IDs can be found in the URL path for the export. The ID is everything after **api/export/**. • This is a comma-delimited list of export IDs for the exports you want to import as Threat Reports. • The order of the tokens must match up with the ordering of the tokens and titles
threatq_ export_ tokens	• Tokens can be found in the **Connection Settings** for the export. • This is a comma-delimited list of export tokens for the exports you want to import as Threat Reports. • The order of the tokens must match up with the ordering of the IDs and titles.
threatq_ export_titles	• This title is fully customizable and it is whatever you want to show up in Carbon Black Response as the Threat Report Title. • This is a comma-delimited list of export titles for the exports you want to import as Threat Reports. • The order of the tokens must match up with the ordering of the

Field	Details
	IDs and tokens.
threatq_ verify_ssl	• This is whether you want to verify the SSL connection between CB Response and ThreatQ. Setting this to true may cause connection issues. This option is set to false by default
hreatq_http_ proxy (optional)	• If you want to communicate with ThreatQ via an HTTP proxy, set it here • This must include the username, password, host/ip, and port
threatq_ https_proxy (optional)	• If you want to communicate with ThreatQ via an HTTPS proxy, set it here • This must include the username, password, host/ip, and port

5. Save the configuration file by pressing the **Esc** key, typing **:wq**, then pressing the **Enter** key.

Upgrading the Connector

To upgrade an existing ThreatQ connector:

1. Transfer the new RPM file to your Carbon Black Response instance.

```
scp python-cb-threatq-connector-1.0.0-10.x86_
64.rpm root@<cb-ip-address>:/tmp/
```

2. SSH into your Carbon Black Response instance.
3. Stop the old connector.

```
service cb-threatq-connector stop
```

4. Use RPM to upgrade your connector.

```
cd /tmp
rpm -Uvh --force python-cb-threatq-connector-
1.0.0-10.x86_64.rpm --ignoreos --nofiledigest
```

5. Start the connector up again.

```
service cb-threatq-connector start
```