ThreatQ Splunk Implementation Guide

Version 1.0.1

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: Tuesday, May 7, 2019

Contents

Versioning

App Version

- TA-threatquotient-add-on: 1.0.1
- ThreatQAppforSplunk: 1.0.0

Supported Splunk Version

These apps have been tested with Splunk versions 7.0.x, 7.1.x and 7.2.x.

Supported ThreatQuotient Version

These apps require ThreatQ version 4.16.0 or higher. They will NOT work with a ThreatQ version prior to 4.16.0.

Features

The ThreatQuotient App for Splunk provides the following capabilities:

Distributed Deployment

The solution is packaged as two separate Splunk packages:

- **ThreatQuotient Add-on for Splunk:** Deployed on Splunk heavy forwarder and search head.
- **ThreatQuotient App for Splunk:** Deployed on Splunk search head.

Support for Splunk's Common Information Model (CIM) and Enterprise Security (ES)

ES Support: Indicator Data exported from ThreatQuotient is mapped using Splunk CIM in the Splunk ES.

Export Indicators from ThreatQ using Score and Status Filters

- **Score Filter:** You can choose to export indicators with scores greater than or equal to the value configured in the score filter.
- **Status Filter:** You can choose to export indicators with statuses matching the ones configured in the status filter.

Detect Sightings and Return to ThreatQ

- **Detect Sightings:** Indicators from ThreatQ are matched against raw events in Splunk looking for evidence of sightings.
- **Report Sightings:** Sightings are reported back to ThreatQ as events that contain the most up to date information.

Contextualize ThreatQ Data

All data exported from ThreatQ is highly contextualized for Splunk. Context provided for exported indicators includes:

- Indicator sources
- Indicator adversaries
- Indicator attributes
- Indicator status, score and type

Workflow Actions in Splunk to Interact with ThreatQ Data

This App provides the following workflow actions to an analyst to interact with ThreatQ:

- Add Indicator to ThreatQ
  - The user provides indicator type, status and source
- Whitelist an indicator in ThreatQ
- Look up an indicator in ThreatQ
  - Additional context is fetched if this indicator exists in ThreatQ
- Mark an indicator **False Positive** in ThreatQ
- Mark an indicator **True Positive** in ThreatQ

Dashboard for Visualization

The dashboard provides a rich set of real time updated widgets and tables to summarize information, including (but not limited to):

- Total exported indicators and sightings filtered by time range, type and score
- Top 10 indicators with sightings
- Top 10 sources and adversaries (due to the context available from ThreatQ) with sightings
- Static tables summarizing indicators and sightings filtered by time range, type and score

Installation

From the main Splunk interface:

1. Click on the **Down** arrow on the Apps menu located in the main navigation bar.
2. Select the **Find More Apps** option.

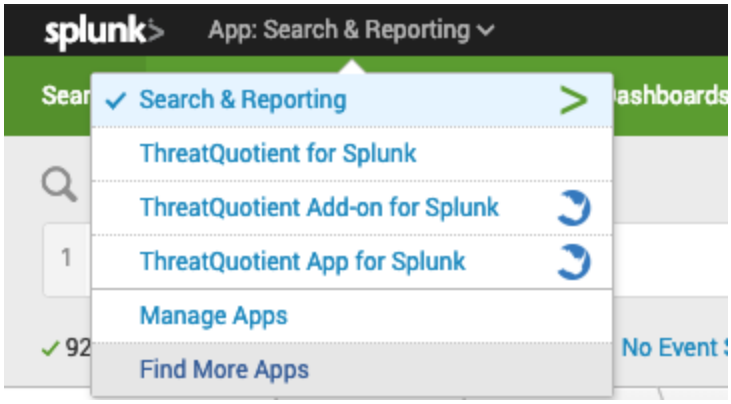


Figure 1: Installation from the Splunk App

3. Search for “ThreatQuotient” and follow the onscreen prompts to install the ThreatQuotient App and ThreatQuotient Add-on.

Deployment

Two Splunk packages need to be deployed for the App to work.

- **TA-threatquotient-add-on:** This package needs to be deployed both on the Splunk **heavy forwarder** and Splunk **search head**.
  - On the heavy forwarder, the add-on App extracts indicators from the ThreatQ appliance and forwards them to the configured splunk index.

- On the search head, the App provides support for ThreatQuotient workflow actions in Splunk.

- **ThreatQAppforSplunk:** This package needs to be deployed only on the Splunk **search head**.

There are two ways in which both Apps can be deployed in Splunk:

1. **Standalone Mode:** In this mode, both Apps are deployed and configured on the same machine.

2. **Distributed Mode:** In this mode, deployment is done as described in the picture below.

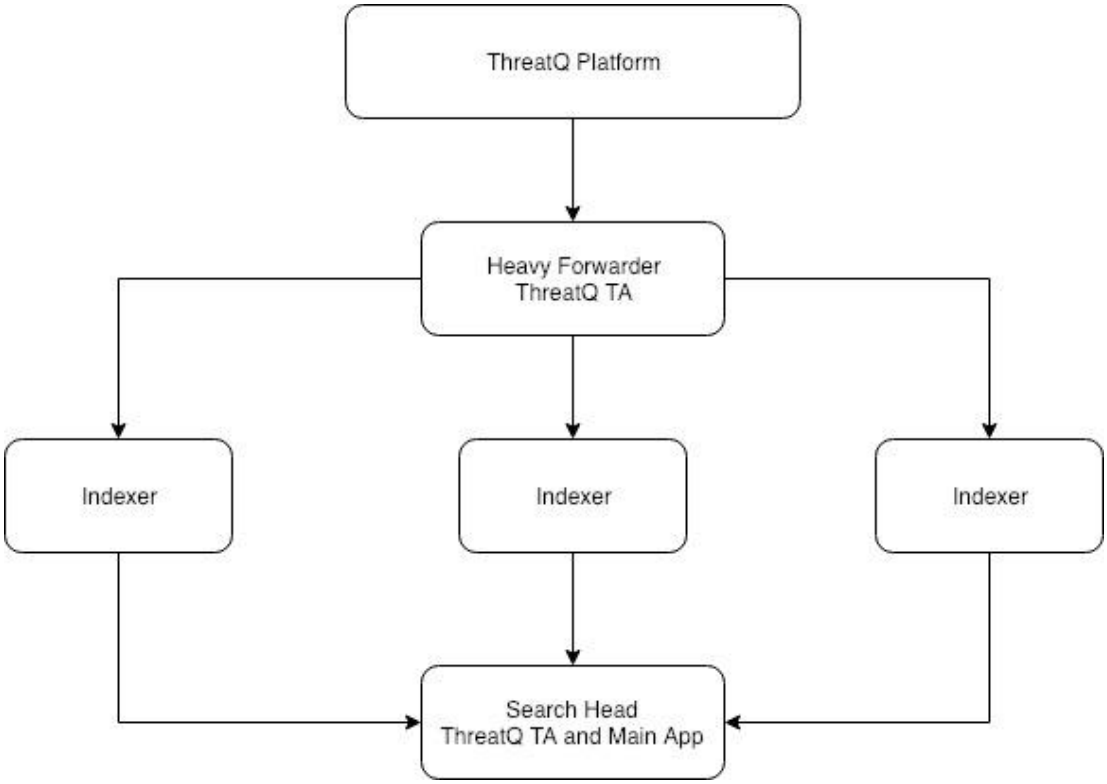


Figure 2: Deployment of Splunk App in Distributed Environment

For a distributed environment with a **cluster of search heads**, you will need to configure the ThreatQuotient Add-on App on the master node, and use the Splunk App deployer to propagate that configuration to all nodes. For the heavy forwarder, it is **not recommended**

that the users should deploy the Add-on app on a cluster, since the data extraction takes place with a custom script, and works the best with a single node.

The table below summarizes the deployment in the distributed Splunk environment:

	Heavy Forwarder	Indexer	Search Head
ThreatQuotient Add-on	Yes • Requires configuration with ThreatQuotient credentials • Requires creating the data collection job	No	Yes • Requires configuration with ThreatQuotient credentials • **Must not be** configured with data collection job
ThreatQuotient App	No	No	Yes • No configuration is required

Table 1: Deployment Matrix for Distributed Environment

Advanced Configuration
If you desire to configure multiple heavy forwarders for a single ThreatQuotient App - this is not typical since the indicators exported from ThreatQ do not exceed a few thousand at most - you would have to make multiple copies of the default ThreatQ Splunk Export, and use a different Export ID on each heavy forwarder. This way, the ThreatQ server can keep track of incremental indicator changes as seen by each distinct Export.

Configuration

ThreatQuotient Add-on

The **ThreatQuotient add-on** is responsible for the following:

Authentication with ThreatQ

On the **Configuration** tab, fields are presented to configure the ThreatQ account authentication as shown below.

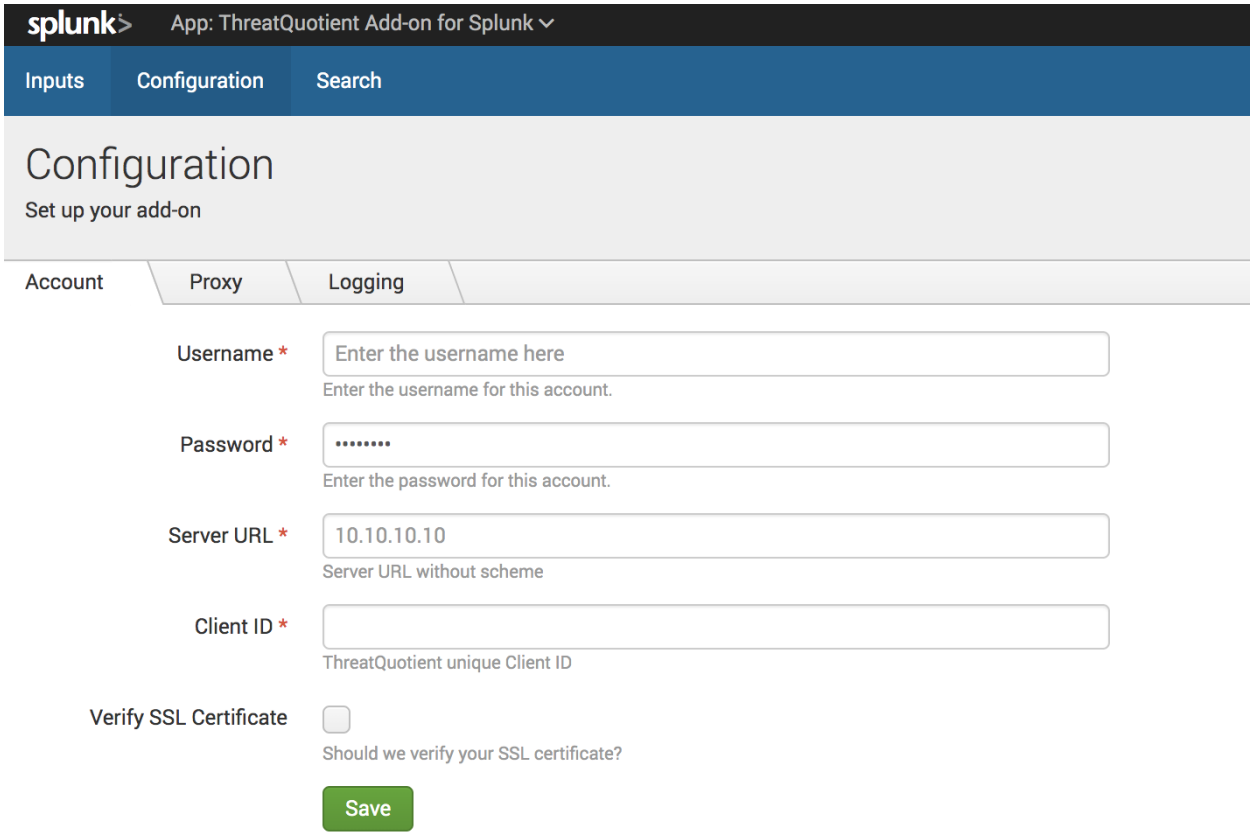


Figure 3: Configuration of Authentication Parameters

Upon clicking the **Save** button, you can see the status of the Authentication action. If the ThreatQuotient appliance is down, and/or the authentication parameters are invalid, an error message will be displayed. Unless the appliance is up and the authentication parameters are valid, this App will not work.

Authentication with the Use of Self Signed Certificates in ThreatQ

It is typical for many ThreatQuotient users to use self signed certificates. If you are doing this, you must perform the following additional configuration steps in the Splunk Add-On App.

In ${SPLUNK_HOME}/etc/apps/TA-threatquotient-add-on/default/ta_threatquotient_add_on_settings.conf, make the following configuration change:

Splunk Search for Listing TQ Indicators

```
[additional_parameters]
verify_cert = false
```

Data Extraction from ThreatQuotient

On the **Inputs** tab, you can click **Create New Input** to add a data collection job as shown below.

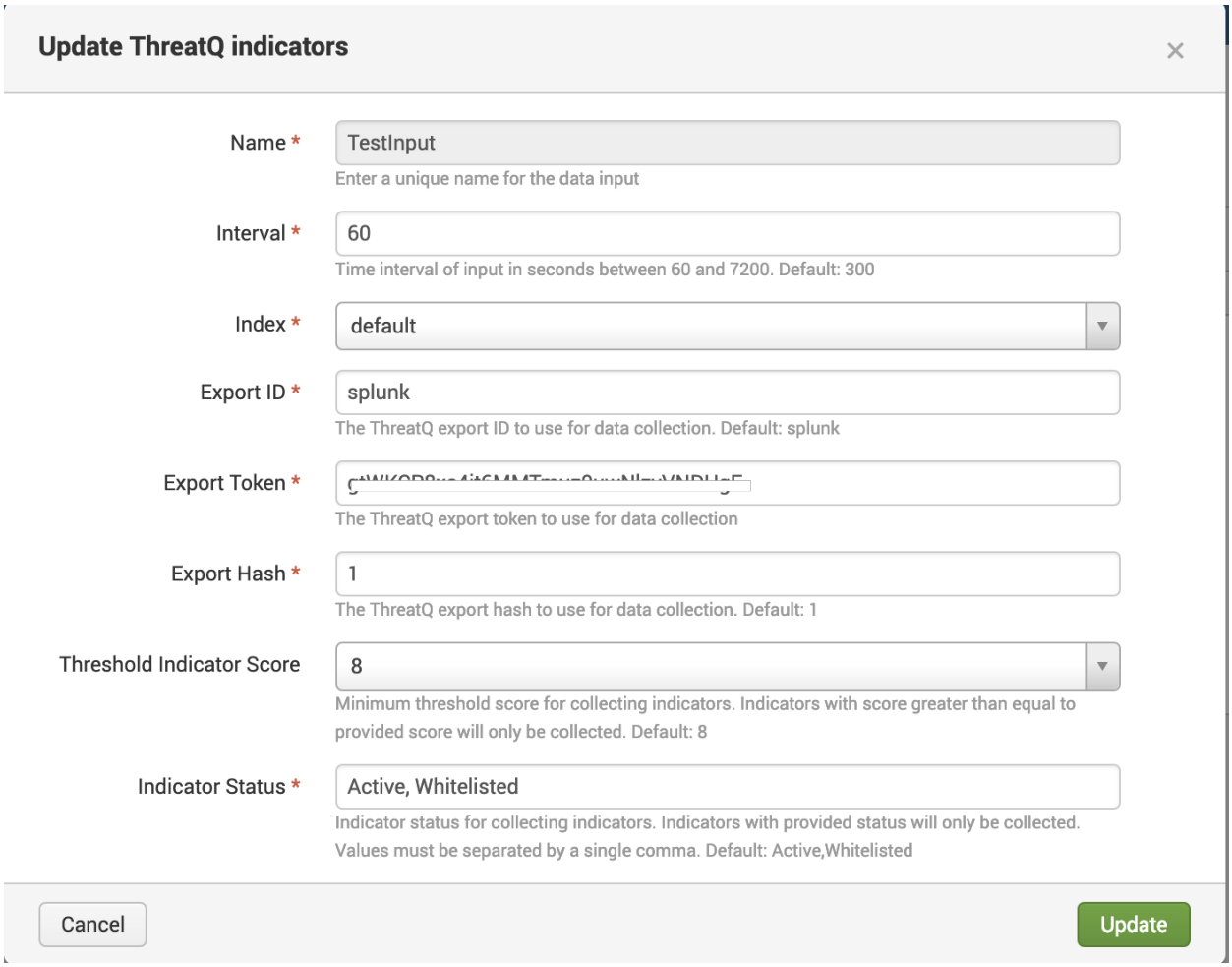


Figure 4: Configuration of Input Data Extraction

ThreatQ instances starting with versions **4.16.0** are shipped with an **Export** that this App uses. Upon the first execution of this job, it results in the export of all indicators. Every subsequent run of this job only results in getting new indicators as well as previously exported indicators that have since changed. Various configuration parameters are described below.

- **Interval:** The frequency of this job. For a faster detection and response, this value can be reduced. Minimum allowed is 60 seconds.
- **Threshold Indicator Score:** Any indicator below this score is not indexed in Splunk. This threshold is very useful to reduce the data being indexed in the **ThreatQuotient App. Default: 8**

- **Indicator Status:** Similar to the score threshold, any indicator not matching the status configured here is not indexed in Splunk. Again, this technique is useful for reducing indexed data. **Default: Active, Whitelisted**
- **Export ID:** Defaulted to **splunk** if you are using the default Splunk export in ThreatQ. If you make a copy of the export, you must configure the ID of the export in this field as seen on the ThreatQ instance.
- **Export Token:** On the ThreatQ appliance, find the export named as **Splunk Indicators Export** and click **Connection Settings**. The token is available in the following configuration screen. See the picture below for reference.

Figure 5: Splunk Export in ThreatQuotient

- **Export Hash:** Defaults to **1**. In the event you want to re-export all indicators from ThreatQ for any reason (such as installing a new Splunk instance), use this configuration. You can configure a different alphanumeric value of length up to 32 and cause exporting all indicators from ThreatQuotient again.

Limitations

- Reducing the set of indicators in Splunk comes at the expense of inability to detect change of scores and/or statuses in indicators. We recommend that users use the "Whitelisted" status in ThreatQ to mark indicators as false positives rather than reducing the indicator score or using custom statuses.
  - It is possible to configure custom indicator statuses (other than Active and Whitelisted) and use those statuses in the workflow for interaction with the **ThreatQuotient Add-on**.
- If you want to use advanced filters (such as adversaries, attributes or sources) to export only a subset of indicators from ThreatQuotient to Splunk, there are two ways to do it:
  - Duplicate the default export, and configure advanced filters. On the Splunk Add-On App, configure scoring filter in such a way that all indicators are accepted (i.e. value of 0).

- Configure a scoring policy to influence indicator scores on certain adversaries, sources or attributes only. On the Splunk Add-On App, configure the scoring filter to accept only certain scores (i.e. value >= 8 for example).

Data Loading in Splunk

As shown in Figure 4, the **Index** parameter allows you to map the data extracted from a job in a predetermined Splunk index. You can create multiple jobs and map them to different Splunk indexes as desired.

Exporting a Large Number of Indicators from ThreatQuotient

It is not recommended that you export an exceptionally large number of indicators from ThreatQuotient to Splunk. We recommend that at any one time (including the initial load up), users export no more than 500K indicators. If this limit is not observed, you may encounter problems including loading the data to Splunk, and assuming the data was loaded correctly anyway, with the performance of your Splunk deployment itself.

The default export shipped with the ThreatQuotient appliance does not apply any filters on the indicators to restrict the set of data being exported. However, you may make a copy of this export and specify any additional filters under Special Parameters. An example is shown in the picture below in which a user has configured a filter with score > 5.

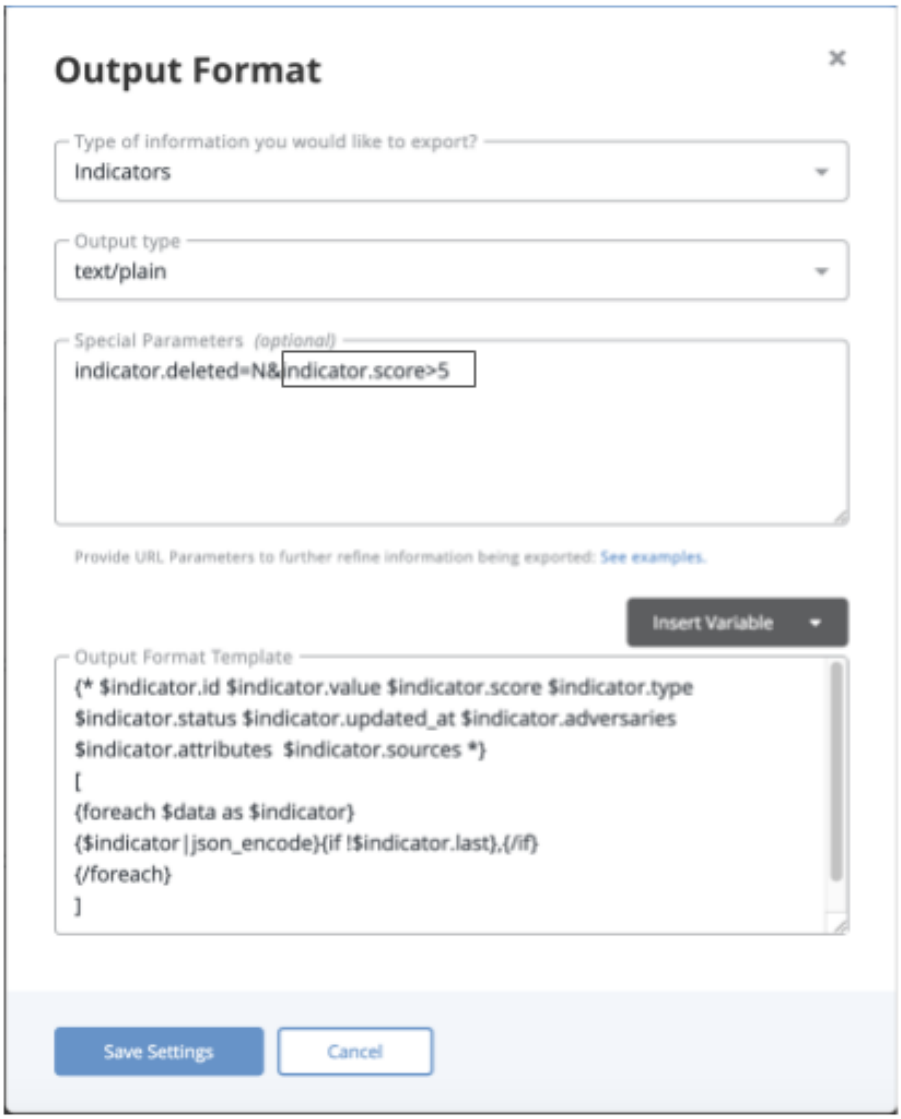


Figure 6: Example of Filters in Splunk Export

ThreatQuotient App

This App does not require any configuration.

Sightings and Feedback to ThreatQ

One of the primary features of this solution is to identify sightings and report them back to ThreatQ.

Sighting in this context is defined as evidence that a **ThreatQ Indicator** was discovered in one or more of the events in Splunk collected via other sources. Recording these sightings and reporting them back to ThreatQ provides analysts with important context around indicators included in their threat intelligence holdings. This section describes various user configurations (in form of macros and saved searches) available to the user to achieve this, and concludes with a summary diagram that describes the whole process.

Separation of Data

ThreatQ indicator data is separated from the rest of the data in this App using a specific **sourcetype**. You can use the following Splunk search query to discover all indicators exported from ThreatQuotient.

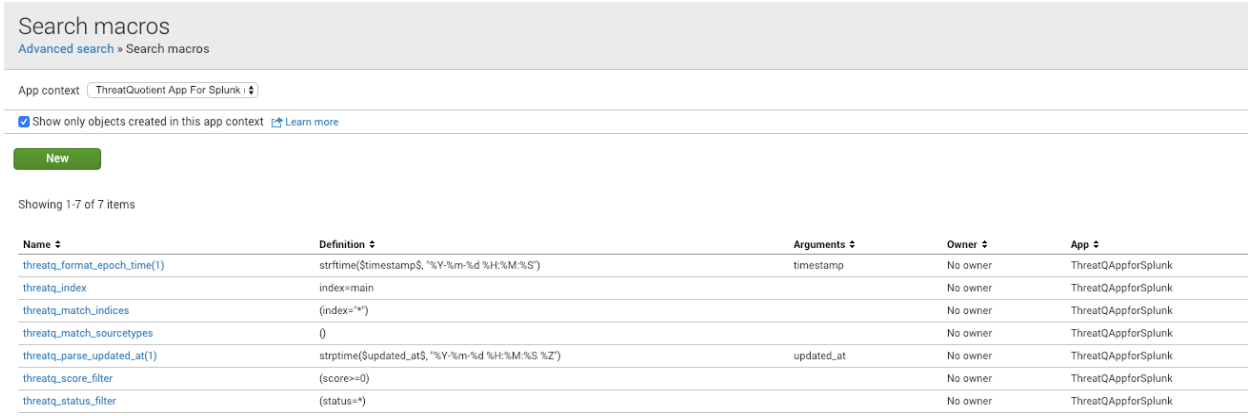
Splunk Search for Listing TQ Indicators

```
sourcetype="threatq:indicators"
```

Note that the same indicator can be exported multiple times if it experienced a change of status and/or score.

Macros

The following macros are used in most of the saved searches this App is configured with (available under **Settings > Advanced Search > Search Macros**).



Name	Definition	Arguments	Owner	App
threatq_format_epoch_time(1)	strftime($timestamp$, "%Y-%m-%d %H:%M:%S")	timestamp	No owner	ThreatQAppforSplunk
threatq_index	index=main		No owner	ThreatQAppforSplunk
threatq_match_indices	(index="*")		No owner	ThreatQAppforSplunk
threatq_match_sourcetypes	()		No owner	ThreatQAppforSplunk
threatq_parse_updated_at(1)	strptime($updated_at$, "%Y-%m-%d %H:%M:%S %Z")	updated_at	No owner	ThreatQAppforSplunk
threatq_score_filter	(score>=0)		No owner	ThreatQAppforSplunk
threatq_status_filter	(status=*)		No owner	ThreatQAppforSplunk

Figure 7: Configurable Macros in ThreatQuotient App

The description of some of these search macros is below.

Saved Search Macro	Description
`threatq_index`	Configures the name of the Splunk index that all ThreatQ indicators are mapped to.
`threatq_match_indices`	Configures which Splunk indices are considered for matching. The users can apply more specific filters here.
`threatq_match_sourcetypes`	Configures which sourcetypes should be **excluded** from matching (the sourcetype **threatq:indicators** is automatically excluded).
`threatq_score_filter`	Configures a score filter for all indexed indicators from ThreatQ that should be considered for detecting sightings.
`threatq_status_filter`	In conjunction with the score filter, configures a

Saved Search Macro	Description
	status filter for all indexed indicators from ThreatQ to be considered for detecting sightings.

[Table 2: Configurable Macros]

Saved Searches

The Splunk App uses saved searches for discovering sightings and reporting them back to ThreatQ. The App is preconfigured with saved searches, which are periodic processes (registered to the crontab) designed to map indicators to specific Splunk indices and match these indicators to events. Saved search processes also move older indicators out of the main lookup tables and for ES customers, move indicators to specific ES lookup tables according to the mapping described in this document.

The table below describes some of the saved searches this App is preconfigured with.

Saved Search	Description	Default Period
`threatq_update_master_lookup`	Fetch ThreatQ indicators from the **default** index and link them against the **master lookup table**. This prepares the indicators for detecting evidence of sightings.	15 mins
`threatq_match_indicators`	Finds evidence of sightings for all indicators in the **master lookup table**. If sightings are detected, indicators are moved to the **match lookup table**.	30 mins

Saved Search	Description	Default Period
threatq_cleanup_indicators_on_indicators_change	If indicator status changes from Active to Whitelisted (or any other status not considered for finding evidence of sightings), or if the indicator score drops below a threshold (making the indicator ineligible for finding evidence of sightings), removes those indicators from both **master lookup table** and **match lookup table**.	15 mins
threatq_update_matched_indicators	Finds evidence of sightings for all indicators in the **match lookup table**.	30 mins
threatq_consume_indicators	Creates events in ThreatQ for all newly detected sightings.	15 mins
threatq_update_retired_indicators	Clean up indicators that haven't been matched on in the last 90 days from both **master lookup table** and **match lookup table**.	1440 mins

[**Table 3: Saved Searches for Discovering and Reporting of Sightings**]

Editability rules
Because of the way sightings are found in Splunk using two saved searches (**threatq_match_indicators** and **threatq_update_matched_indicators**), their frequency must be the same if edited. The default frequency for both saved searches is 30 mins.

Saved Searches Documentation

The following table documents the macros for saved searches as configured by default on the ThreatQuotient App.

Saved Search	Default Macro
threatq_update_master_lookup	`threatq_index` sourcetype="threatq:indicators" `threatq_score_filter` `threatq_status_filter` | dedup value | eval ioc_id=id, ioc_value=value, sources='sources{}.value', adversaries='adversaries{}.value' | table value, updated_at, status, type, score, ioc_id, ioc_value, sources, adversaries | outputlookup key_field=value master_lookup | join ioc_value [| inputlookup threatq_matched_indicators | table ioc_value, match_time, first_seen, last_seen, match_count, sid] | eval value=ioc_value | table ioc_id, ioc_value, value, match_time, first_seen, last_seen, match_count, score, status, type, updated_at, sources,

Saved Search	Default Macro
	adversaries, sid | outputlookup key_field=value threatq_matched_indicators
threatq_cleanup_indicators_on_indicators_change	| inputlookup master_lookup | search NOT [search `threatq_index` sourcetype="threatq:indicators" | dedup value | search [| inputlookup master_lookup | table ioc_value | rename ioc_value as value | format] NOT (`threatq_score_filter` `threatq_status_filter`) | table value | rename value as ioc_value | format] | outputlookup master_lookup | join ioc_value [| inputlookup threatq_matched_indicators | table ioc_value, match_time, first_seen, last_seen, match_count, sid] | outputlookup threatq_matched_indicators
threatq_match_indicators	`threatq_match_indices` `threatq_match_sourcetypes` sourcetype!="threatq:indicators" | threatqmatchiocs

Saved Search	Default Macro
threatq_update_matched_indicators	`threatq_match_indices` `threatq_match_sourcetypes` sourcetype!="threatq:indicators" | threatqmatchiocs is_update=true
threatq_consume_indicators	| inputlookup threatq_matched_indicators | eval start_time=relative_time(now(), "-16m") | where last_seen > start_time | threatqconsumeindicators

Saved Search	Default Macro
threatq_update_retired_indicators		inputlookup master_lookup	search NOT [| inputlookup master_lookup	search NOT [| inputlookup threatq_matched_indicators | search NOT [| inputlookup threatq_matched_indicators	eval threshold_time=now()-7776000, value=ioc_value | where last_seen < threshold_time	outputlookup key_field=value threatq_retired_matched_indicators	table ioc_value | format]	outputlookup threatq_matched_indicators	table ioc_value	format] | eval threshold_time=now()-7776000, updated_at_epoch=`threatq_parse_updated_at(updated_at)`, value=ioc_value | where updated_at_epoch < threshold_time	outputlookup key_field=value threatq_retired_indicators	table ioc_value | format]	outputlookup master_lookup

[Table 4: Saved Search Macros]

Reporting Sightings in ThreatQ

A sighting in Splunk is evidence that an indicator from ThreatQ was seen in one or more events in Splunk. To an analyst, this is important information that can be reported back in form of an Event.

ThreatQuotient captures all sightings for an indicator in a single event. When more sightings are detected for the same indicator, certain attributes for that event are updated - this allows the analyst to gather context on sightings for that indicator. The following 4 attributes are recorded for the event.

- **First Seen:** Timestamp when the first sighting for this indicator was recorded in Splunk. This attribute does not change.
- **Last Seen:** Timestamp when the latest sighting for this indicator is recoded in Splunk. This attribute updates as newer sightings are detected.
- **Count:** The total count of all sightings recorded for this indicator starting from the time **First Seen** until **Last Seen**.
- **Splunk URL:** The URL that allows the analyst to view all sightings for this indicator in Splunk starting from **First Seen** until **Last Seen**.

The screen capture below shows an example event recorded in ThreatQuotient by the Splunk App.

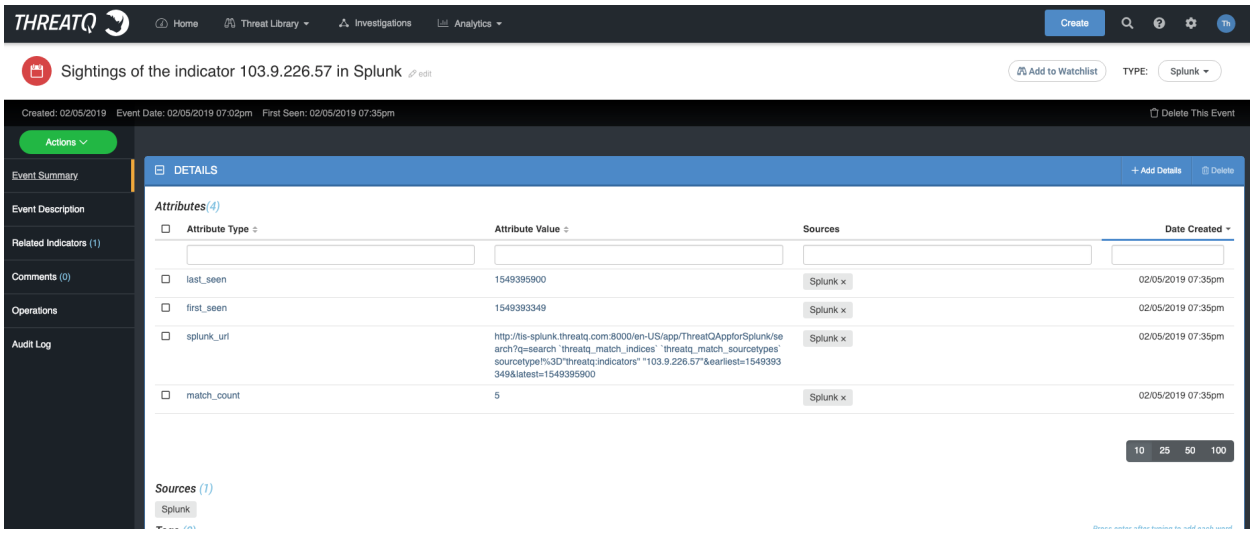


Figure 8: Example Event in ThreatQ

Putting Everything Together

The following steps summarize how indicators are stored in Splunk and how sightings are reported back to ThreatQ.

1. The Input job configured on **ThreatQuotient Add-on** (on the heavy forwarder) pulls indicators from ThreatQ.
2. The heavy forwarder sends the indicators to the indexer which indexes the indicators to the **default** index (user can override).
3. The periodic saved search job **threatq_update_master_lookup** maps the newly indexed ThreatQ indicators to the **master lookup table**.
4. The periodic saved search job **threatq_match_indicators** finds evidence of sightings of all indicators in the **master lookup table** against all events in Splunk (as filtered via various configurable macros described above in this section).
    a. If evidence of sightings is found for a specific indicator, it is moved to the **match lookup table**.

5. Simultaneously, another periodic saved search job **threatq_update_matched_indicators** finds more sightings for all indicators from the **match lookup table** against all events in Splunk (as filtered by the same configurable macros).
6. A periodic saved search **threatq_consume_indicators** will create events in ThreatQ to represent evidence of sightings in Splunk.
7. The periodic saved search job **threatq_update_retired_indicators** takes all indicators that are not updated in the past 90 days out of both the **master lookup table** and **matched lookup table**.

The following diagram summarizes this process.

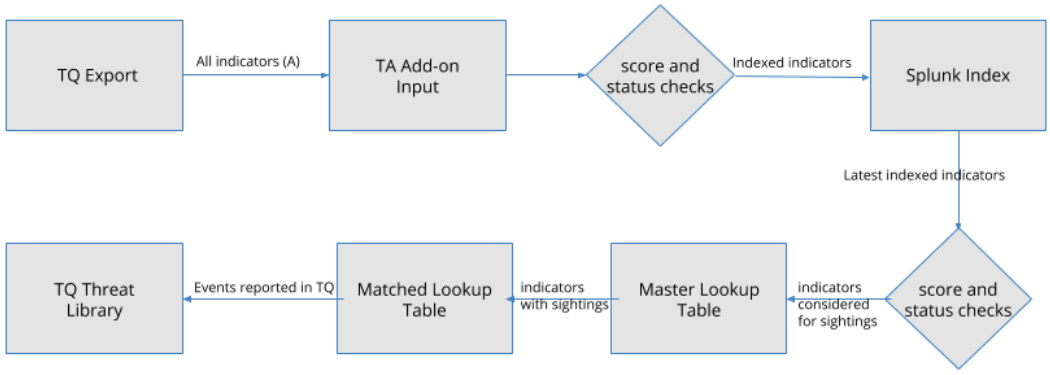


Figure 9: Detecting and Reporting Indicator Matches to ThreatQ

Workflow Actions

The **ThreatQuotient Add-on** provides five user workflow actions to the analysts for providing interactivity with the ThreatQuotient platform from Splunk. As shown on the diagram below, the actions can be invoked on any Splunk event by expanding the event view and clicking on the downarrow in the column below **Action**.

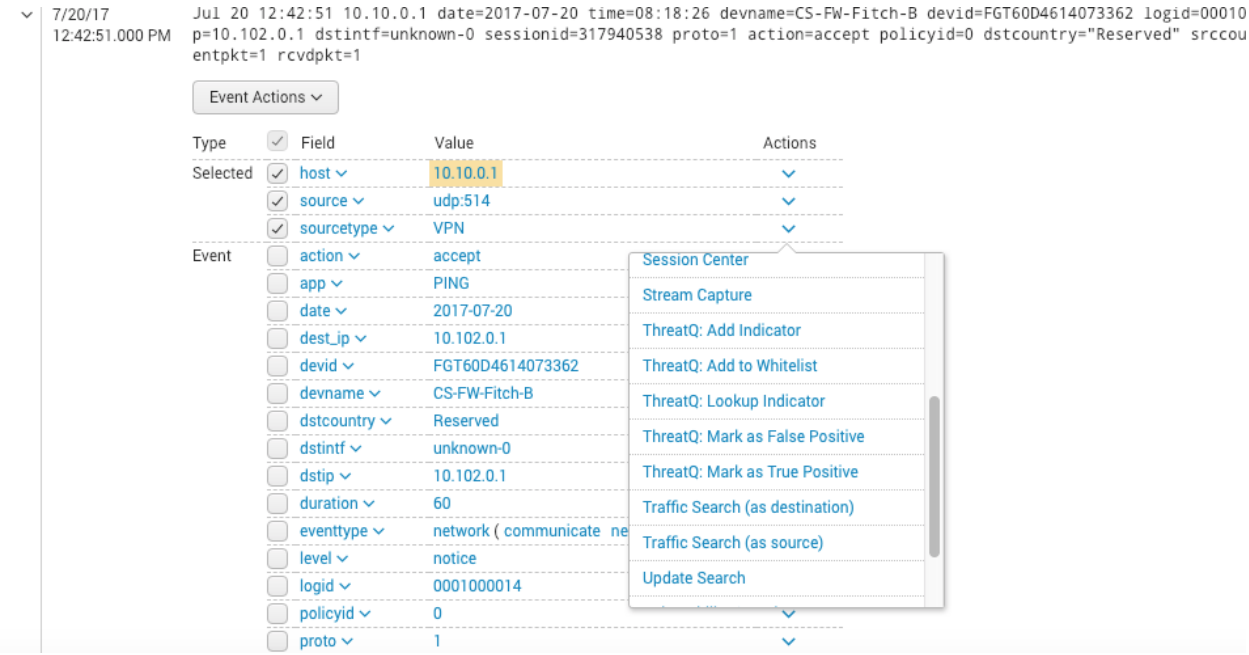


Figure 10: ThreatQuotient Workflow Actions

The actions are described below.

ThreatQ: Add Indicator

This workflow action adds the indicator to ThreatQ. You are presented with UI inputs that allow you to select indicator type, status and source. If the data and type do not match, an error is reported. Successful completion of this workflow action results in the indicator being successfully added to the ThreatQ Threat Library.

ThreatQ: Add to Whitelist

This workflow action sets the status of the indicator to Whitelisted in ThreatQ. If the indicator does not exist in ThreatQ, an error is reported.

ThreatQ: Lookup Indicator

This workflow action searches for an indicator in ThreatQ and pulls additional context for that indicator. If the indicator does not exist in ThreatQ, an error is reported.

ThreatQ: Mark as False Positive

This workflow action adds the attribute key-value "**False Positive: True**" to the indicator in ThreatQ. If the indicator does not exist in ThreatQ, an error is reported.

ThreatQ: Mark as True Positive

This workflow action adds the attribute key-value "**True Positive: True**" to the indicator in ThreatQ. If the indicator does not exist in ThreatQ, an error is reported.

Enterprise Security Support

ThreatQ Indicators to Splunk Lookup Mapping Table (using CIM)

The App provides support to the Splunk Enterprise Security (ES) customers by making ThreatQ data more accessible using Splunk's native ES lookup tables. The following table provides how ThreatQ data is mapped to the Splunk ES lookup tables. This data is then available in various ES dashboards.

ThreatQ type	Threat intelligence type
CIDR Block	local_ip_intel
Email Address	local_email_intel
Email Subject	local_email_intel
File Name	local_file_intel
FQDN	local_domain_intel
Fuzzy Hash	local_file_intel
GOST Hash	local_file_intel
IP Address	local_ip_intel
MD5	local_file_intel
Registry Key	local_registry_intel
Service Name	local_service_intel
SHA-1	local_file_intel
SHA-256	local_file_intel

ThreatQ type	Threat intelligence type
SHA-384	local_file_intel
SHA-512	local_file_intel
x509 Serial	local_certificate_intel
x509 Subject	local_certificate_intel
URL	local_http_intel
URL Path	local_http_intel
Username	local_user_intel

[Table 5: ThreatQ indicator mapping using Splunk Common Information Model (CIM)]

To view the events and indicators, navigate to **Enterprise Security > Security Intelligence > Threat Intelligence**.

- **Threat Activity:** Shows the list of event which are compatible with CIM apps.
- **Threat Artifacts:** Shows the list of indicators fetched from the ThreatQ

Using Threat Intelligence Data in Splunk Enterprise Security

Splunk's Enterprise Security App provides the means of using your threat intelligence data to match against events mapped to standard Splunk models. Refer to the Splunk's documentation on **Enterprise Security Workflow for Threat Intelligence** as described here: http://dev.splunk.com/view/enterprise-security/SP-CAAAFBC.

ThreatQuotient provides mapping of the threat intelligence data to the standard lookup tables in Splunk Enterprise Security via the saved searches described above. Using the default Threat Generation Searches in the Enterprise Security, the ES app will find matches and report those matches in the `threat_activity` index as described in the link above. However, note that when using the Enterprise Security App, you will not have additional context (sources and adversaries), workflow actions ,and reporting sightings back to ThreatQuotient available to you.

Saved Searches for CIM Mapping

In addition to the core saved searches, the following saved searches apply for Enterprise Security (ES) customers. The saved searches listed run once a day and map ThreatQ indicators by type to Splunk ES lookup tables as described in the **Mapping Table** section of the document.

> By default, the **scheduling** of all saved searches for porting Threat Intelligence data from ThreatQ to lookup tables in the ES are **disabled**. This is because not all users have Enterprise Security App installed. If you have this App installed and want to port the Threat Intelligence data over, you will need to enable the scheduling of these saved searches.

ES Saved Search	Description
`threatq_update_threat_intelligence_lookup_email_address`	Map ThreatQ **type 4** indicators to **local_email_intel**
`threatq_update_threat_intelligence_lookup_email_subject`	Map ThreatQ **type 6** indicators to **local_email_intel**
`threatq_update_threat_intelligence_lookup_file_name`	Map ThreatQ **type 9** indicators to **local_file_intel**

ES Saved Search	Description
threatq_update_threat_intel-ligence_lookup_fqdn	Map ThreatQ **type 10** indicators to **local_domain_intel**
threatq_update_threat_intel-ligence_lookup_hash	Map ThreatQ **type [11,12,15,20,21,22,23]** indicators to **local_file_intel**
threatq_update_threat_intel-ligence_lookup_ip	Map ThreatQ **type 14** indicators to **local_ip_intel**
threatq_update_threat_intel-ligence_lookup_registry	Map ThreatQ **type 18** indicators to **local_registry_intel**
threatq_update_threat_intel-ligence_lookup_service	Map ThreatQ **type 19** indicators to **local_service_intel**
threatq_update_threat_intel-ligence_lookup_certificate_serial	Map ThreatQ **type 25** indicators to **local_certificate_intel**
threatq_update_threat_intel-ligence_lookup_certificate_sub-ject	Map ThreatQ **type 26** indicators to **local_certificate_intel**
threatq_update_threat_intel-ligence_lookup_url	Map ThreatQ **type 27** indicators to **local_http_intel**

ES Saved Search	Description
`threatq_update_threat_intel-ligence_lookup_user`	Map ThreatQ **type 30** indicators to **local_user_intel**

[Table 6: Saved Searches for Mapping ThreatQ Indicator data to Splunk's CIM]

Dashboards

Preconfigured dashboards are packaged with **ThreatQuotient App** to allow the analyst versatile visual representation of all indicator data from ThreatQ and the corresponding sightings.

Cumulative Counts

The top section of the dashboard shows total count for all ThreatQ indicators in the **master lookup table** (on the left) and the **match lookup table** (in the right) (all time and the last 24 hours). It is important to note that the data displayed as Sightings are not the total sightings; rather it is the total number of indicators for which evidence of sightings has been found. Example screen capture below.

Figure 11: Dashboard: Cumulative Counts

Score Breakdown

The next section shows the distribution of indicator scores for indicators in master and match lookup tables as bar charts. Example screenshot below. These charts do not have a time filter. The counts for individual score breakdown represent the cumulative indicator count. As an example, notice that there are two indicators with sightings each with score 9 (which matches up with the cumulative sightings count of 2 in the chart above).

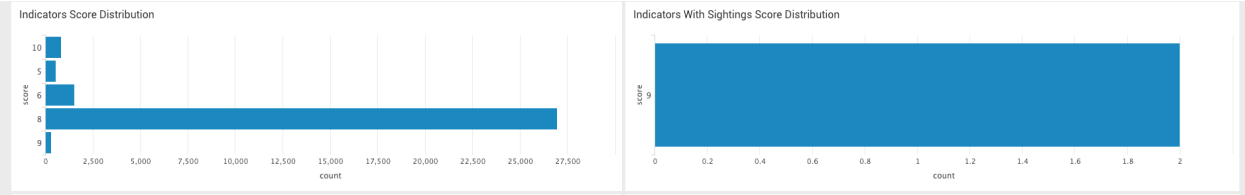


Figure 12: Dashboard: Score Breakdown

Type Breakdown

This section shows the distribution of indicator types for indicators in master and match lookup tables as pie charts. As the score distributions above, these are cumulative distributions. Example screenshot below. Hovering over each portion of the pie chart will display the indicator count for that specific portion.

Figure 13: Dashboard: Type Breakdown

Source Breakdown

This section shows the breakdown of indicators and sighted indicators by sources. Example screenshot below. One thing to note here is that all indicators must have at least one source, but some indicators may have more than one. For this reason, the cumulative counts in the

charts below may exceed the total number of indicators and sighted indicators in the lookup tables.

Figure 14: Dashboard: Source Breakdown

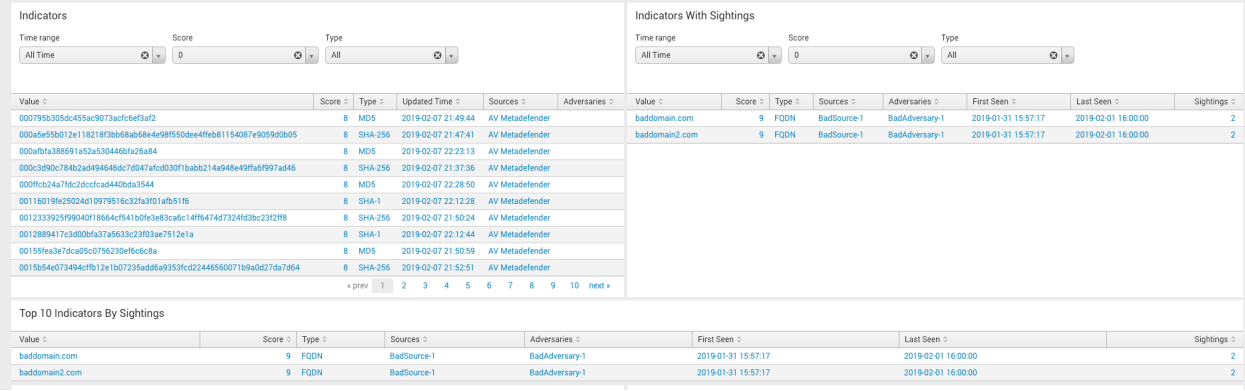
Adversaries Breakdown

This section shows the breakdown of indicators and sighted indicators by adversaries. Example screenshot below. One thing to note here is that not all indicators have adversaries; although some indicators may have more than one. Depending upon how many indicators have adversaries, the total cumulative counts in the charts below may be less or more than the total indicators and sighted indicators in the lookup tables. For the example dataset below, there is only one adversary assigned to a few indicators, and those same indicators are sighted.

Figure 15: Dashboard: Adversary Breakdown

Static Table View

This section shows all indicators and sightings in static tables - time filters are provided and defaulted to the last 24 hours. Score and type filters are also available for both. This information gives a threat analyst a single place to view all sightings in Splunk. In the screenshot below, notice there are two indicators sighted, each with 2 sightings.



Indicators

Time range: All Time | Score: 0 | Type: All

Value	Score	Type	Updated Time	Sources	Adversaries
000795b305dc455ac9073acfc6ef3af2	8	MD5	2019-02-07 21:49:44	AV Metadefender	
000a5e55b012e118218f3bb68ab68e4e98f550dee4ffeb81154087e9059d0b05	8	SHA-256	2019-02-07 21:47:41	AV Metadefender	
000afbfa388691a52a530446bfa26a84	8	MD5	2019-02-07 22:23:13	AV Metadefender	
000c3d90c784b2ad494646dc7d047afcd030f1babb214a948e49ffa6f997ad46	8	SHA-256	2019-02-07 21:37:36	AV Metadefender	
000ffcb24a7fdc2dccfcad440bda3544	8	MD5	2019-02-07 22:28:50	AV Metadefender	
00116019fe25024d10979516c32fa3f01afb51f6	8	SHA-1	2019-02-07 22:12:28	AV Metadefender	
0012333925f99040f18664cf541b0fe3e83ca6c14ff6474d7324fd3bc23f2ff8	8	SHA-256	2019-02-07 21:50:24	AV Metadefender	
0012889417c3d00bfa37a5633c23f03ae7512e1a	8	SHA-1	2019-02-07 22:12:44	AV Metadefender	
00155fea3e7dca05c0756230ef6c6c8a	8	MD5	2019-02-07 21:50:59	AV Metadefender	
0015b54e073494cffb12e1b07235add6a9353fcd22446560071b9a0d27da7d64	8	SHA-256	2019-02-07 21:52:51	AV Metadefender	

« prev 1 2 3 4 5 6 7 8 9 10 next »

Indicators With Sightings

Time range: All Time | Score: 0 | Type: All

Value	Score	Type	Sources	Adversaries	First Seen	Last Seen	Sightings
baddomain.com	9	FQDN	BadSource-1	BadAdversary-1	2019-01-31 15:57:17	2019-02-01 16:00:00	2
baddomain2.com	9	FQDN	BadSource-1	BadAdversary-1	2019-01-31 15:57:17	2019-02-01 16:00:00	2

Top 10 Indicators By Sightings

Value	Score	Type	Sources	Adversaries	First Seen	Last Seen	Sightings
baddomain.com	9	FQDN	BadSource-1	BadAdversary-1	2019-01-31 15:57:17	2019-02-01 16:00:00	2
baddomain2.com	9	FQDN	BadSource-1	BadAdversary-1	2019-01-31 15:57:17	2019-02-01 16:00:00	2

Figure 16: Dashboard: Static Indicators and Sighted Indicators Tables

Top 10 By Sightings

The final section displays top 10 indicators by sightings, top 10 sources by sightings and top 10 adversaries by sightings in form of a static table, bar chart and bar chart respectively. This information gives an analyst a quick view of the indicators sources and adversaries with the most matches within Splunk.

Top 10 Indicators By Sightings

Value	Score	Type	Sources	Adversaries	First Seen	Last Seen	Sightings
baddomain.com	9	FQDN	BadSource-1	BadAdversary-1	2019-01-31 15:57:17	2019-02-01 16:00:00	2
baddomain2.com	9	FQDN	BadSource-1	BadAdversary-1	2019-01-31 15:57:17	2019-02-01 16:00:00	2

Figure 17: Dashboard: Top 10 Indicators by Sightings

Sources

Example screenshot below. Notice the source **BadSource-1** appears as the top source with sightings corresponding to the sighted indicators as displayed in the static table above. Also notice that the sightings count is 4, which corresponds to 2 sightings each for the sighted indicators.

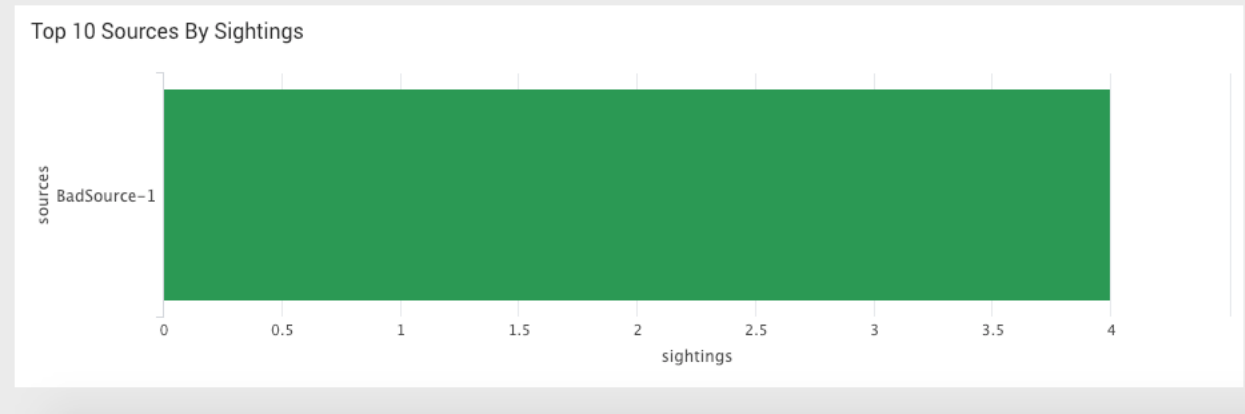


Figure 17: Dashboard: Top 10 Sources by Sightings

Adversaries

Example screenshot below. Notice the source **BadAdversary-1** appears as the top adversary with sightings corresponding to the sighted indicators as displayed in the static table above. Also notice that the sightings count is 4, which corresponds to 2 sightings each for the sighted indicators.

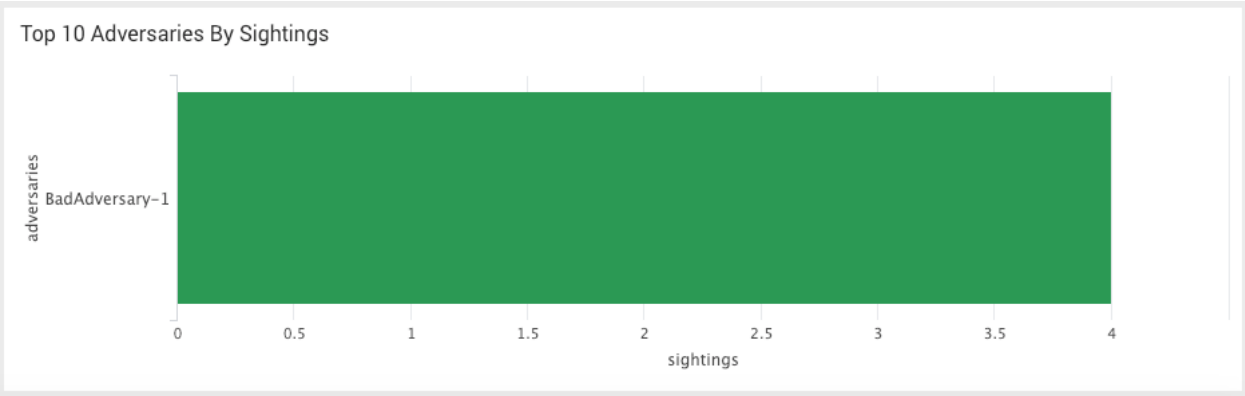


Figure 18: Dashboard: Top 10 Adversaries by Sightings

Troubleshooting

- To troubleshoot **ThreatQuotient Add-on** please check the log file below:

```
$SPLUNK_HOME/var/log/Splunk/ta_threatquotient_
add_on_threatq_indicators.log
```

- To find all unique indicators indexed in Splunk by the Add-On (Splunk App allows you to select a specific time range):

```
sourcetype="threatq:indicators" | dedup value
```

- To check the data collected by data collection use query like:

```
"index=your_index_name sourcetype=threatq_indic-
ators"
```

- Make sure all the saved searches are enabled.
- Make sure the macro is updated as per the settings.
- To troubleshoot any behavior with the master table lookup tables (which is used in the dashboards), the following query is useful:

```
index=_internal sourcetype="scheduler" saved-
search_name=threatq_update_master_lookup status-
s=success
```

The log file can be found at the following location:

```
/opt/splunk/var/log/splunk/scheduler.log
```

- If the user changes macros for global score and status thresholds, the audit logs can be accessed using the following two saved searches:

Splunk Search for Listing TQ Indicators

```
index=_internal threatq_score_filter source-
type="splunkd_ui_access"
index=_internal threatq_score_filter source-
type="splunkd_access"
```

- Logs for the saved search to update the master lookup table can be accessed using the following query (the same query can be used to check the run statuses of any saved search; just replace with the appropriate saved search name):

```
index=_internal sourcetype=scheduler saved-
search_id="nobody;threatqappforsplunk;threatq_
update_master_lookup
```

Change Log

Version 1.0.1:

During authentication, users can now specify whether to verify or disable the SSL certificate.