ThreatQuotient

ThreatQuotient App for Splunk Guide

Version 2.3.0

August 26, 2021

ThreatQuotient
11400 Commerce Park Dr., Suite 200
Reston, VA 20191

Support
Email: support@threatq.com
Web: support.threatq.com
Phone: 703.574.9893

Contents

Warning and Disclaimer

ThreatQuotient, Inc. provides this document “as is”, without representation or warranty of any kind, express or implied, including without limitation any warranty concerning the accuracy, adequacy, or completeness of such information contained herein. ThreatQuotient, Inc. does not assume responsibility for the use or inability to use the software product as a result of providing this information.

Versioning

- TA-threatquotient-add-on: `2.3.0`
- ThreatQ App for Splunk: `2.3.0`
- Supported on ThreatQ versions >= `4.16.0`

Supported Splunk Versions

- Splunk 7.2.x
- Splunk 7.3.x
- Splunk 8.x (Python 2)
- Splunk 8.x (Python 3)

Features

The ThreatQuotient App for Splunk provides the following capabilities:

- Distributed Deployment
- Support for Splunk's Common Information Model (CIM) and Enterprise Security (ES)
- Export Indicators from ThreatQ using Score and Status Filters
- Detect Sightings and Return to ThreatQ
- Contextualize ThreatQ Data
- Workflow Actions in Splunk to Interact with ThreatQ Data
- Dashboard Visualization

Distributed Deployment

The ThreatQ Splunk App consists of two separate Splunk packages:

PACKAGE	DESCRIPTION
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)

SUPPORT	DESCRIPTION
CIM Support	For users who map third party data (firewall events, logs for example) to Splunk's data models in CIM. The App provides optimized performance by leveraging those data models.
ES Support	Indicator data exported from ThreatQ is mapped to lookup tables native to Splunk ES. Threat Intelligence support for Enterprise Security is provided using its REST APIs.

Export Indicators from ThreatQ using Score and Status Filters

PACKAGE	DESCRIPTION
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

PACKAGE	DESCRIPTION
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
- Indicator Score
- Indicator Type

Workflow Actions in Splunk to Interact with ThreatQ Data

> Workflow actions are only available for fields that are configured to be extracted. Additional fields can be configured for extraction by clicking **Event Actions -> Extract Fields**.

The ThreatQ Splunk App provides the following workflow actions to allow an analyst to interact with ThreatQ:

- Add an 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.

See the Workflow Actions chapter for more details.

Dashboard for Visualization

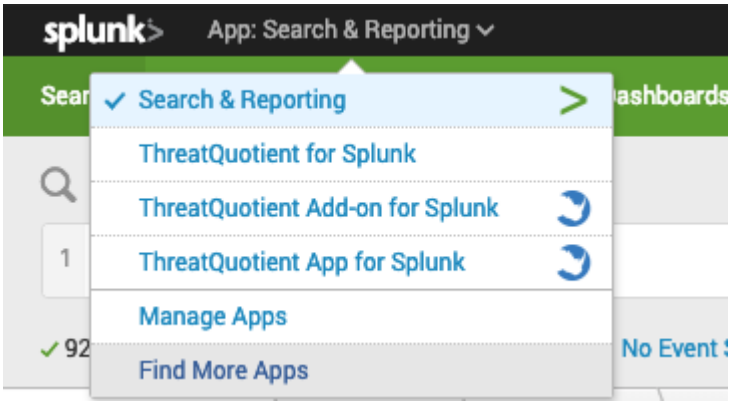
The dashboard provides a rich set of real time updated widgets and tables to summarize information, including (but not limited to):

- 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.

See the Dashboards section of the Scaling the App chapter for more details.

Installation

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



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

Upgrading

The ThreatQ Splunk App Version 2.1.0 introduced several key improvements and updates that require user action upon upgrade.

KV Store

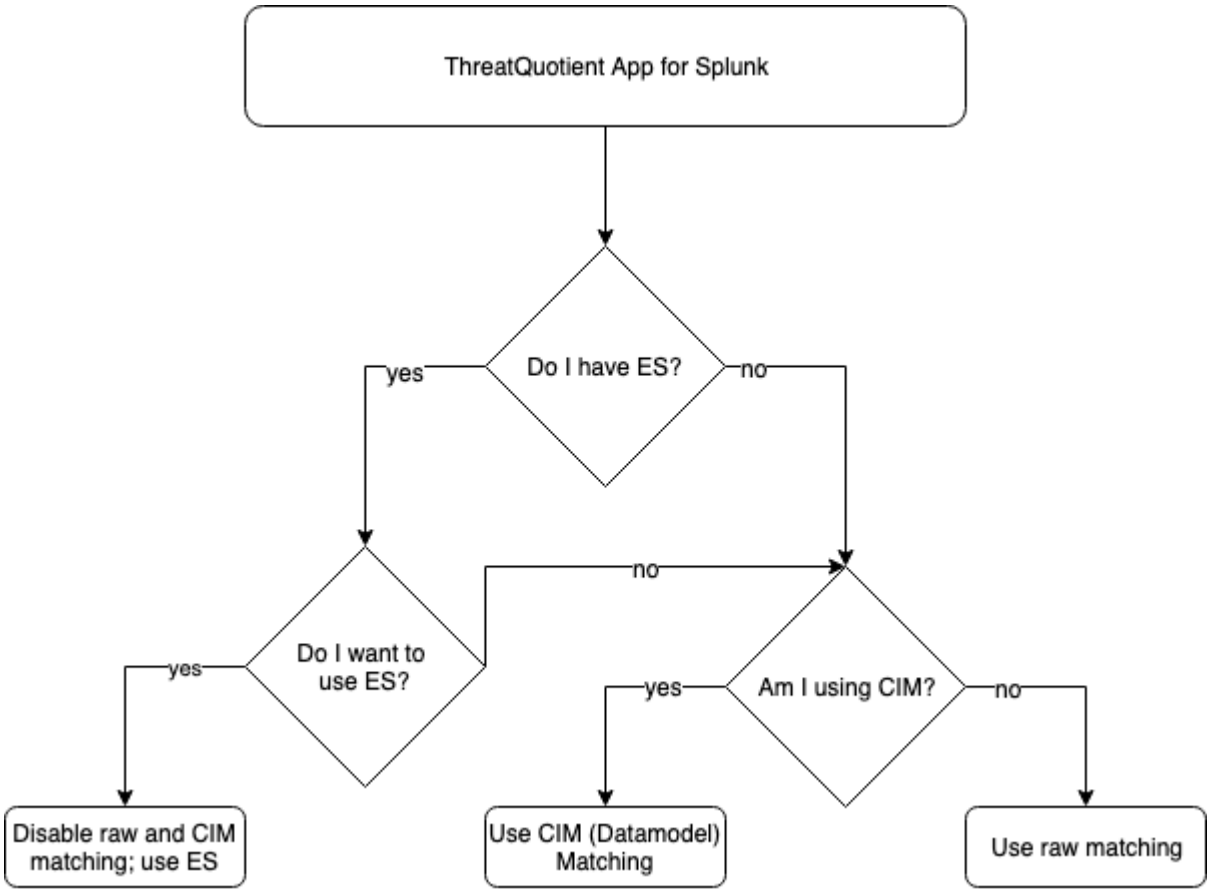
Upon upgrade to version 2.1.0, ThreatQ data will be directly stored to the **KVStore** opposed to a designated index. After the upgrade process completes, users are required to complete the **Splunk KVStore Rest Configuration** form, found under the Configuration section. Users can continue to store data in the index or update the input configuration to disable it.

Scoring and Status Threshold Configurations

With ThreatQ Splunk App version 2.1.0 release, `threatq_score_filter` and `threatq_status_filter` macro configurations are no longer available. The thresholds for score and status can be configured on the input and the values will determine what data gets added to the ThreatQ app for Splunk and the index (if enabled).

App Usage

The ThreatQuotient App for Splunk can be used in one of three possible modes. Follow the flow diagram below to determine which mode to use.



MODES	DESCRIPTION
Raw Matching Mode	Raw Matching Mode is applicable if you do not have Splunk Enterprise Security (ES) and do not map your traffic to Splunk's CIM. In this mode, the App treats all events as raw binary data and looks for evidence of sightings inside said data using optimized regexes. See the tables in the Performance section for the expected performance data.
CIM Matching Mode	CIM Matching Mode, sometimes referred to as **Datamodel Match Mode**, should be used if you want to amp your traffic using Spunk's CIM but do not wish to use Enterprise Security (ES). In this mode, the app uses the

MODES	DESCRIPTION
	mapping table described in CIM Support section to find evidence of sightings and report back matches. This form of matching is more optimized since the algorithm can now reference well known fields in standard data models instead of looking for matches in the whole binary data.
Enterprise Security	Enterprise Security Mode is applicable if you use Enterprise Security for your end-to-end workflow, and want to get the threat data in ES. In this mode, you do not use any capability of the ThreatQuotient App, and instead rely on Enterprise Security to find and report on evidence of sightings.

Deployment

The ThreatQ Splunk app requires two packages to be deployed:

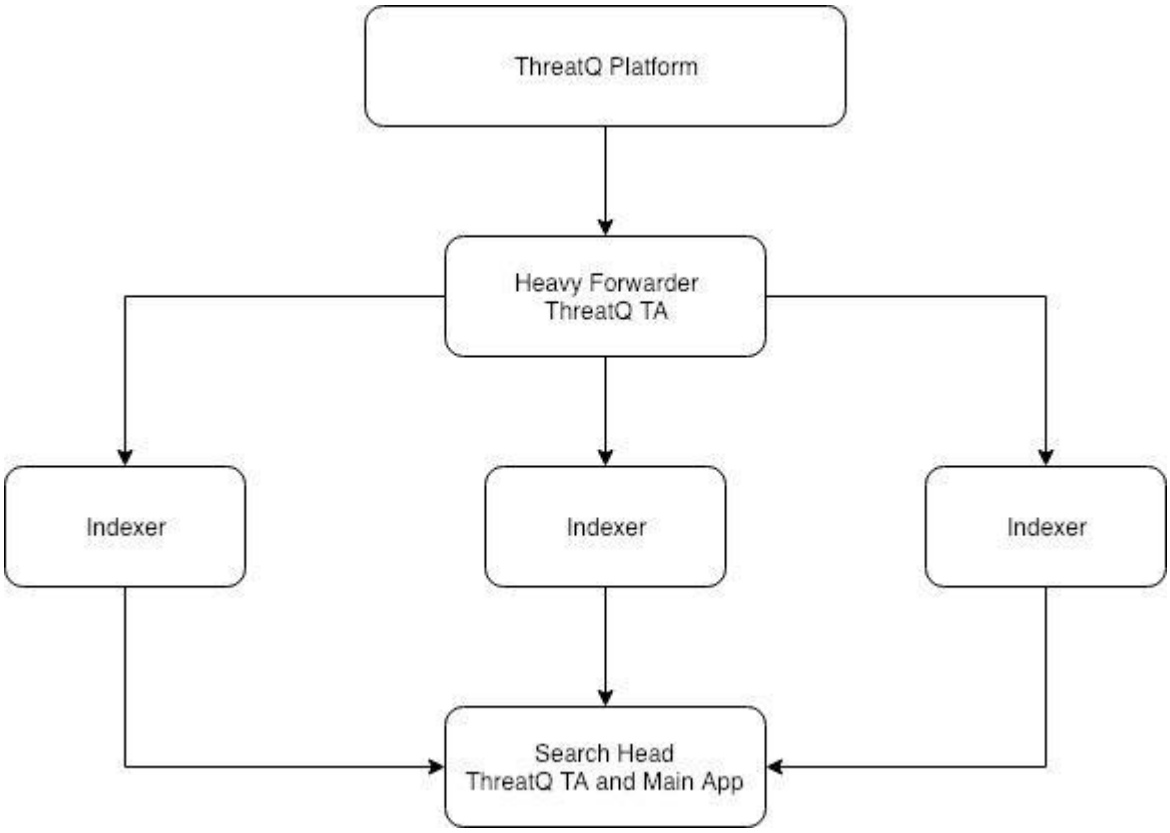
PACKAGE	DESCRIPTION
TA-threatquotient-add-on	This package needs to be deployed both on the Splunk heavy forwarder and Splunk search head. • On the heavy forwarder, the add-on App extracts indicators from the ThreatQ appliance and forwards them to the configured Splunk index. • On the search head, the App provides support for ThreatQuotient workflow actions in Splunk.
ThreatQAppforSplunk	This package needs to be deployed only on the Splunk search head.

Deployment Methods

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

METHOD	DESCRIPTION
Standalone Mode	In this mode, both Apps are deployed and configured on the same machine.
Distributed Mode	In this mode, deployment is done as described in the image below.

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 you 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. See the Advanced Configuration section for more details regarding heavy forwarders.

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

Deployment Matrix for Distributed Environment

APP	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.

Advanced Configuration

Configuring multiple heavy forwarders for a single ThreatQ Splunk App is not typical as the indicators exported from ThreatQ do not exceed a few thousand at most. If you intend to have multiple heavy forwarders, you will have to make multiple copies of the default ThreatQ Splunk Export using a different Export ID on each heavy forwarder. This will allow the ThreatQ server to keep track of incremental indicator changes as seen by each distinct export.

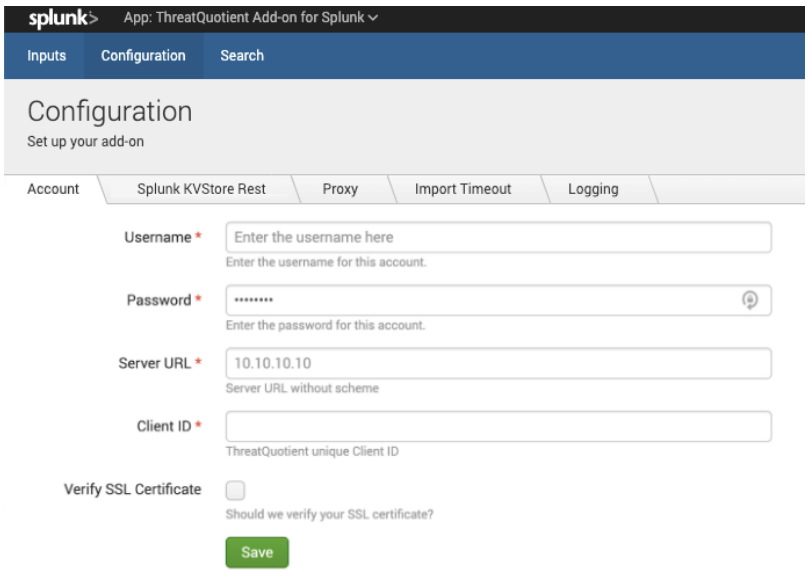
Configuration

ThreatQuotient Add-on

The **ThreatQuotient add-on** is responsible for authenticating with the ThreatQ platform.

Authentication with ThreatQ

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



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 common for many ThreatQuotient users to leverage self signed certificates. If this is the case, 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
```

Splunk KVStore Rest

The App Key Value Store, commonly referred to as the **Splunk KVStore**, is a Splunk Enterprise feature that allows you to save/retrieve data within Splunk apps.

You can read more about the KVStore Splunk feature in Splunk's Developer documentation:

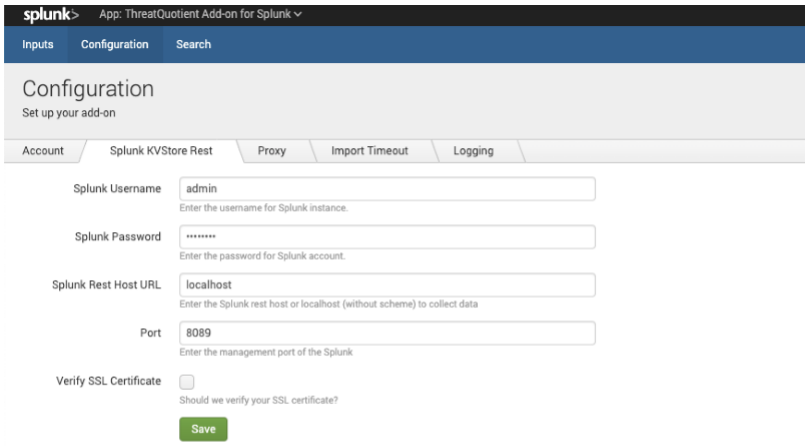
https://dev.splunk.com/enterprise/docs/developapps/manageknowledge/kvstore/

The Splunk KVStore Rest configuration should be updated for distributed setups to ensure data is saved into the KVStore.

Click on the **Splunk KV Store Rest** tab and complete the following fields:

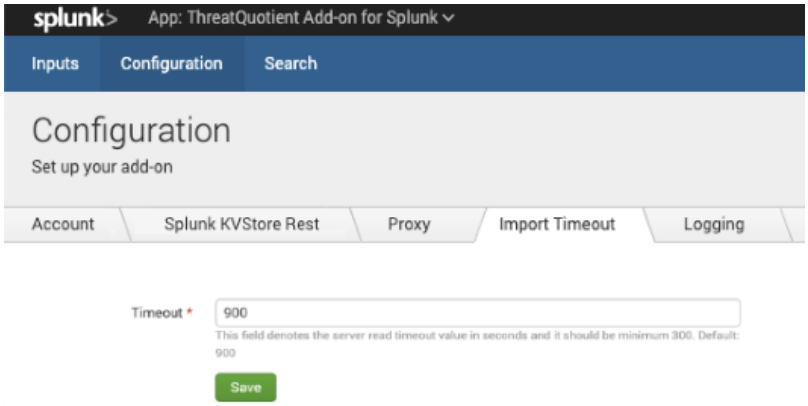
FIELD	DESCRIPTION
Splunk Username	Your username for your Splunk instance.
Splunk Password	The password for your Splunk user account.

FIELD	DESCRIPTION
Splunk Rest Host URL	The Splunk rest host or localhost (without scheme) to collect data. This is the Splunk Management Host, commonly the Search Head or Cluster member.
Port	The Management port for Splunk.
Verify SSL Certificate	Checkbox – verify SSL Certificate.



Import Timeout

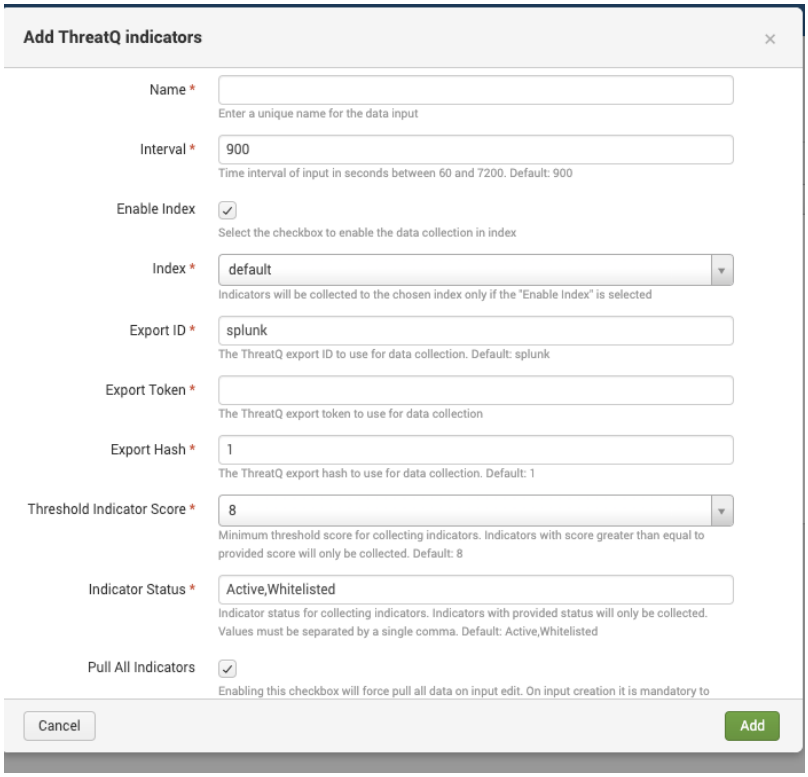
Click on the Import Timeout tab to set server read timeout value in seconds.



The default value is 900 seconds. The minimum value allowed is 300 seconds.

Data Extraction from ThreatQ

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



ThreatQ instances, starting with version 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.

> ⚠ With ThreatQ Splunk App version 2.1.0 release, `threatq_score_filter` and `threatq_status_filter` macro configurations are no longer available. The thresholds for score and status can be configured on the input and those values will determine what data gets added to the ThreatQ app for Splunk and the index (if enabled).

PARAMETER	DESCRIPTION
Interval	The frequency of this job. For a faster detection and response, this value can be reduced. Minimum allowed is 60 seconds.
Enable Index	Enabling this option will result in data being saved to the designated index. Unchecking this option will result in data being saved directly to the KVStore. This option is enabled by default as previous app versions required ThreatQ data to be saved to the index. You must first complete the Splunk KVStore Rest configuration tab before disabling index storage. See the Authentication with the Use of Self Signed Certificates in ThreatQ section for more details.
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. The default value is 8. See the important note, located at the top of this table, regarding changes to score and status thresholds with ThreatQ Splunk App version 2.1.0.
Pull All Indicators	Enabling this checkbox will force pull all data on input edit.

PARAMETER	DESCRIPTION
	The checkbox must be selected, upon input creation, before saving. This option should be utilized when changing the status or score of any input.
Indicator Status	Similar to the score threshold, any indicator not matching the status configured here is not indexed in Splunk. This technique is useful for reducing indexed data. The default values are **Active** and **Whitelisted**. See the important note, located at the top of this table, regarding changes to score and status thresholds with ThreatQ Splunk App version 2.1.0
Export ID	Defaults to splunk. Use this value when using the default splunk export in ThreatQ (Splunk Indicators Export). 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 instance, 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. 
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 con-figuration. You can configure a different alphanumeric value of length up to 32 and cause exporting all indicators from ThreatQuotient again.

Pagination Support

Initial import of ThreatQ data will now be performed using the pagination feature which imports a maximum of 10,000 records at once. After the initial import is complete, the import will revert to the differential method of pulling data. This will be the default behavior whenever a new input is created.The following commands can be used to view, add or update the pagination setting on the inputs:

> The following actions can be performed through the app UI via the Pull All Indicators option –see the Pull All Indicators option in the Data Extraction from ThreatQ section. The commands below are CLI alternatives to the UI option.

ACTION	COMMAND
View the pagination setting for each input	`curl -k -u username:password https://localhost:8089/servicesNS/nobody/TA-threatquotient-add- on/storage/collections/data/TA_threatquotient_add_on_checkpointer`
Update the pagination setting for an input	`curl -k -u username:password https://localhost:8089/servicesNS/nobody/TA-threatquotient-add- on/storage/collections/data/TA_threatquotient_add_on_checkpointer/ {input_name} -H 'Content-Type: application/json' -d '{"state" : " {\"pull_all_iocs\": true|false}"}'`
Add a pagination setting for an input	`curl -k -u user:password https://localhost:8089/servicesNS/nobody/TA-threatquotient-add- on/storage/collections/data/TA_threatquotient_add_on_checkpointer -H 'Content-Type: application/json' -d '{"_key": "<input_name>", "state" : "{\"pull_all_iocs\": <true|false>}"}'`
Delete the pagination setting for an input	`curl -k -u username:password -X DELETE https://localhost:8089/servicesNS/n obody/TA-threatquotient-add- on/storage/collections/data/TA_ threatquotient_add_on_checkpointer/ {input_name}`

ACTION	COMMAND

> Whenever a new input is created, the pagination setting (`pull_all_iocs`) will default to true and will be automatically set to false after the initial import is completed.

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.

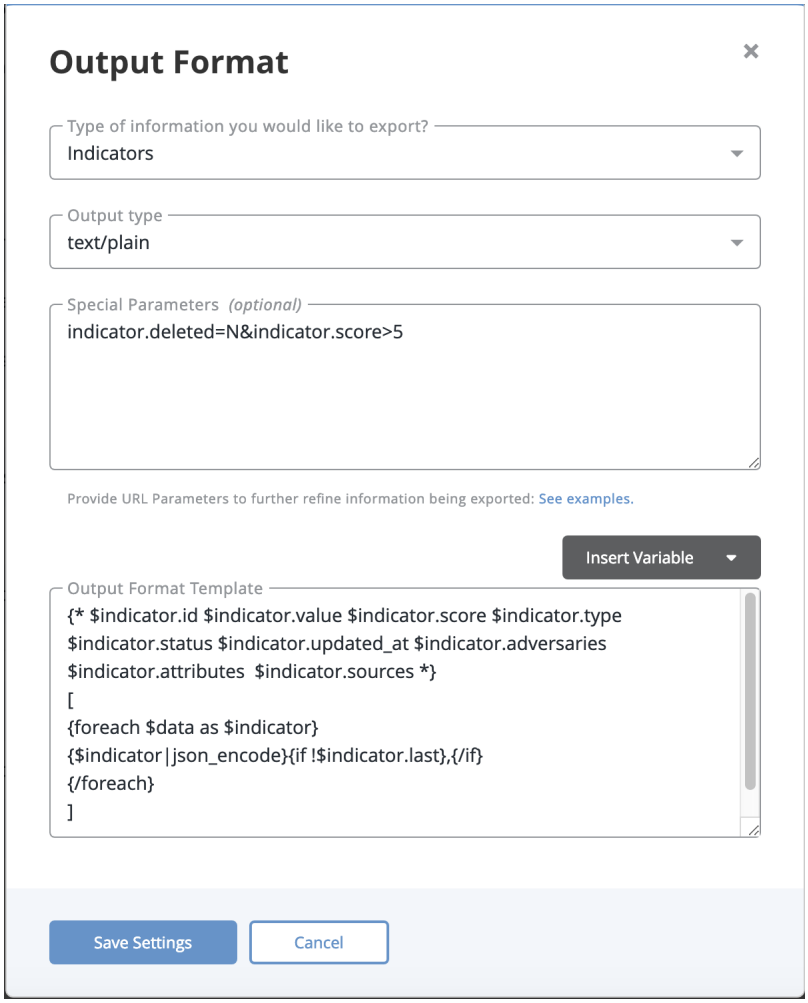
- If you want to use advanced filters (such as adversaries, attributes or sources) to export only a subset of indicators from ThreatQuotient to Splunk, there are two ways to do it:
  - Duplicate the default export and configure advanced filters. On the Splunk Add-On App, configure scoring filter in such 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).

Exporting a Large Number of Indicators from ThreatQ

It is not recommended that you export an exceptionally large number of indicators from ThreatQ to Splunk. We recommend that at any one time, 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.

> If there is a need to re-import the data from ThreatQ, revert the pagination setting for the input to True. This will ensure that the data is imported in batches of 10,000 records at a time.

The default export shipped with the ThreatQ 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.



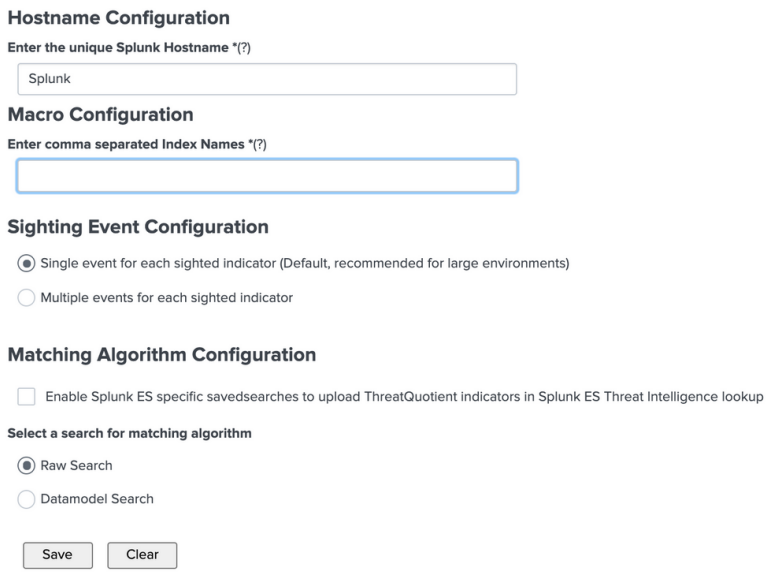
Data Loading in Splunk

As shown in image above, 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.

ThreatQuotient App

The ThreatQuotient App allows you to select one the three modes of operation described in App Usage. The configuration is available on the Splunk's setup page. Navigate to all Apps, locate the ThreatQuotient App for Splunk and click on setup.

The image below displays an example configuration.



The following rules apply for selection:

PARAMETER	DESCRIPTION
Hostname Configuration	The Hostname will be used as a Source Name when Splunk updates attributes on the ThreatQ platform.
Macro Configuration	ThreatQ indicators will be matched against the events from the selected indexes.
Sighting Event Configuration	Configuration option for event creation in ThreatQ for sighted indicators.
	MATCHING ALGORITHM CONFIGURATION
Splunk ES Specific Saved Searches	You can select Enable Splunk ES in conjunction with either **Raw Search** or **Datamodel Search**.
Search for Matching Algorithm	You can select either **Raw Search** or **Datamodel Search**, but not both. At the initial setup, you do not have to select either **Raw Search** or **Datamodel Search** modes. This disables the matching algorithm completely and gives you the opportunity to determine the right scale of data your installation can handle. See the Performance chapter for more details. ⚠ Attempting to search too much data may results in some saved searches being skipped based on your Splunk deployment type and hardware specification.

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.

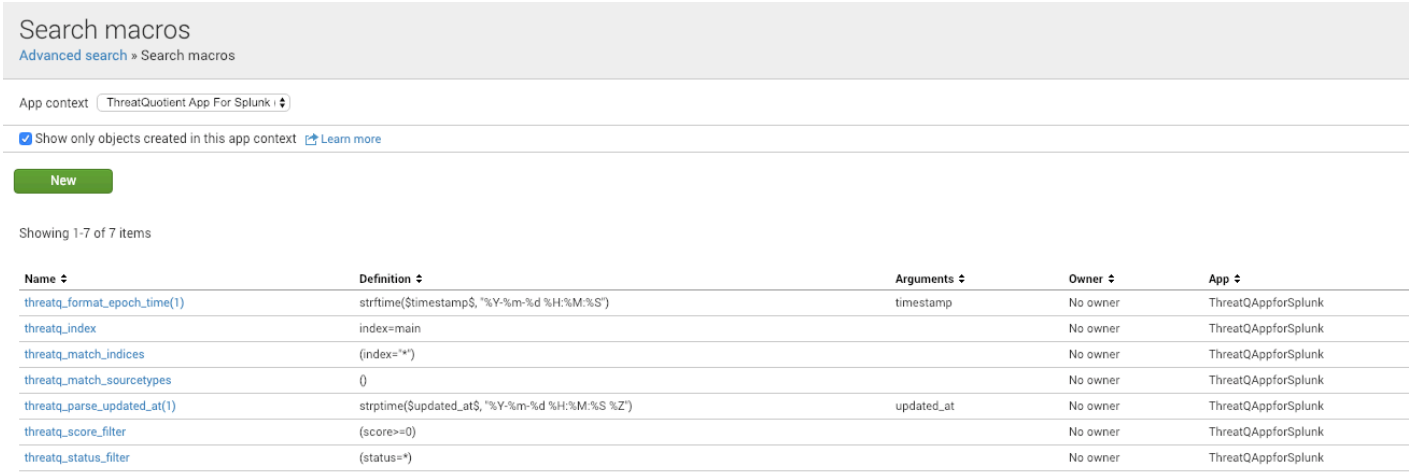
Splunk Search for Listing TQ Indicators

```
sourcetype=”threatq:indicators”
```

> 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

The description of some of these search macros is below.

> With ThreatQ Splunk App version 2.1.0 release, `threatq_score_filter` and `threatq_status_filter` macro configurations are no longer available. The thresholds for score and status can be configured on the input and those values will determine what data gets added to ThreatQ app for Splunk and the index (if enabled).

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_match_process_count	Determines the number of cpu cores utilized for processing the saved searches that are responsible for finding evidence of sightings.
enable_url_partial_match_datamodel	Configures partial URL indicator matching for the **Datamodel**. The default setting is **False**. This macro should be set to **True** if URL indicators are sent to Splunk with a scheme. EX http://, https://

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 with which this App is preconfigured. This table displays two searches applicable only for Raw Matching Mode. Equivalent searches are available for each data model in the Datamodel Matching Mode.

> ThreatQuotient does not recommend setting the frequency to less than 30 minutes, the application default for `threatq_match_ indicator` saved searches, if using the configuration option for creating multiple events for each sighted indicator.

SAVED SEARCH	DESCRIPTION	DEFAULT PERIOD
`threatq_consume_ indicators_new`	Post matched indicators to the consume endpoint of ThreatQ and create atomic events. This search will only be enabled if using the "Create multiple events for each sighted indicator" configuration.	30 minutes
`threatq_match_ indicators (Raw Matching Mode only)`	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 minutes
`threatq_match indicators`	Finds evidence of sightings for all indicators in the match lookup table.	30 minutes
`threatq_cleanup _indicators_on_`	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	15 minutes

SAVED SEARCH	DESCRIPTION	DEFAULT PERIOD
`indicators_change`	evidence of sightings), removes those indicators from both master lookup table and match lookup table.	
`threatq_update_matched_indicators`	Finds evidence of sightings for all indicators in the match lookup table.	30 minutes
`threatq_consume_ indicators`	Creates events in ThreatQ for all newly detected sightings.	15 minutes
`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.	1,440 minutes

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 minutes.

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_consume_indicators_new	| inputlookup threatq_matched_indicators | eval start_time=relative_time(now(), "-35m") | where match_time > start_time | sort 10000 -num(score), -num(match_count) | threatqconsumeindicatorsnew
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 (only Raw Matching Mode)	`threatq_match_indices` `threatq_match_sourcetypes` sourcetype!="threatq:indicators" | threatqmatchiocs
threatq_update_matched_indicators (only Raw Matching Mode)	`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
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

As described above, two of the saved searches are applicable only for the Raw Matching Mode. If you select **Datamodel Matching Mode** from the configuration as described in the Configuration section, the above two saved searches for **Raw Matching Mode** will disable automatically, and the equivalent saved searches for the **Datamodel Matching Mode** will be enabled.

Reporting Sightings in ThreatQ

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

Single Event for Each Sighted Indicator

ThreatQ 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.

Multiple Events for Each Sighted Indicator

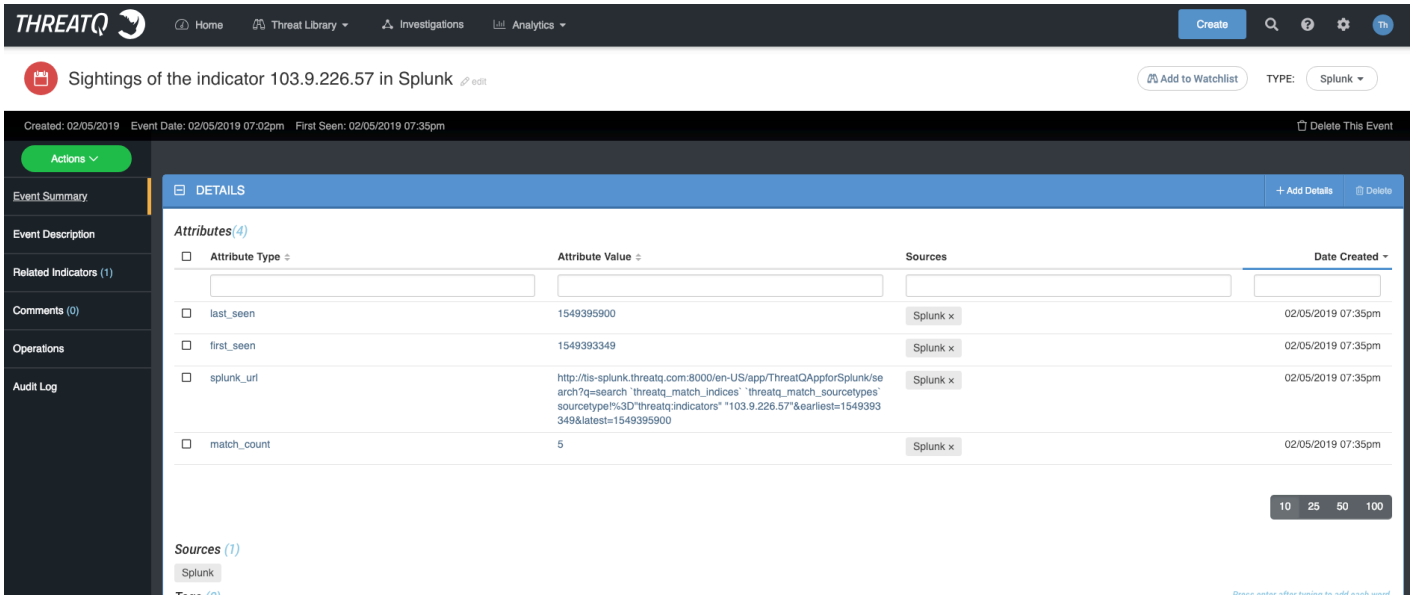
If multiple sightings for the event are seen during the same time period, all sightings will be captured in a single event. However, if more sightings are seen in the future for the same indicator, a new event will be created in ThreatQ.

See the **Sighting Event Configuration** instructions under the ThreatQuotient App section for more details.

The following 4 attributes are recorded for the event.

ATTRIBUTE	DESCRIPTION
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.



The following contextual data are added to the indicator:

ATTRIBUTE	DESCRIPTION
Splunk Sighting Timestamp	When the latest sighting for this indicator was recorded in Splunk.
Match Count	The total count of all sightings recorded for this indicator.
Source	**Splunk** will be added as the **Source** for this indicator.

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) or KVStore.

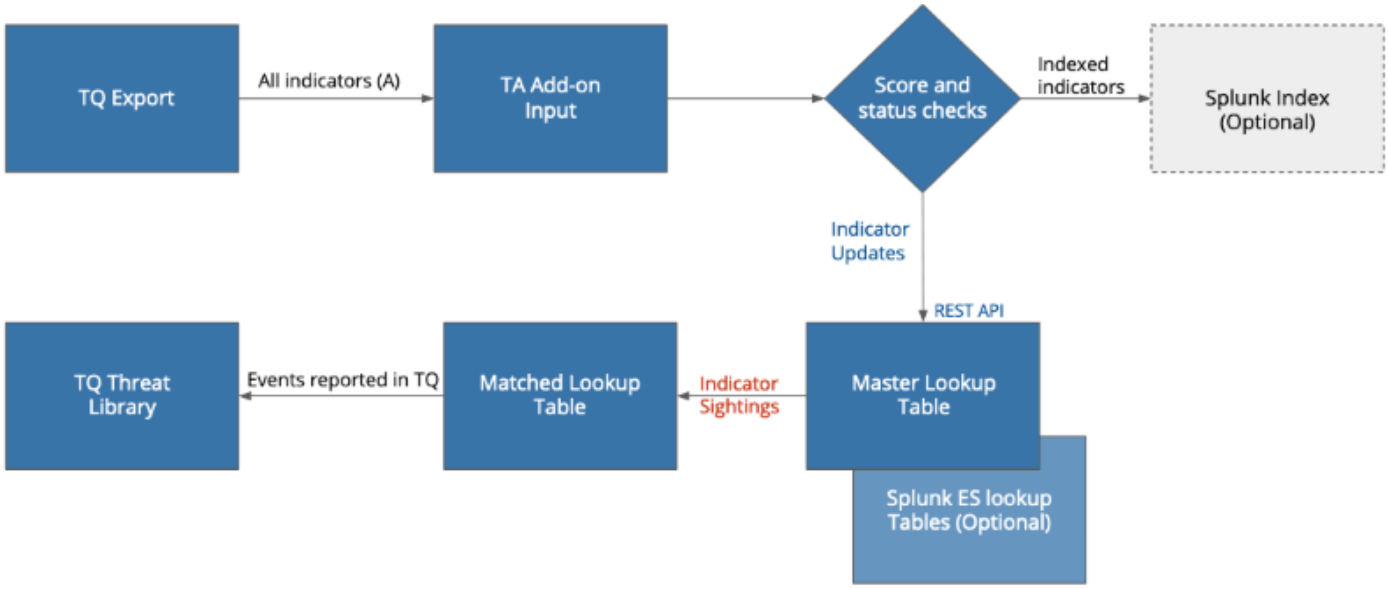
> You can configure how data is saved, to the designed index or KVStore, via the Enable Index checkbox on the Add ThreatQ Indicators form. See the Data Extraction from ThreatQ section of this guide for more details.

3. 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).

   > If evidence of sightings is found for a specific indicator, it is moved to the **match lookup table**.

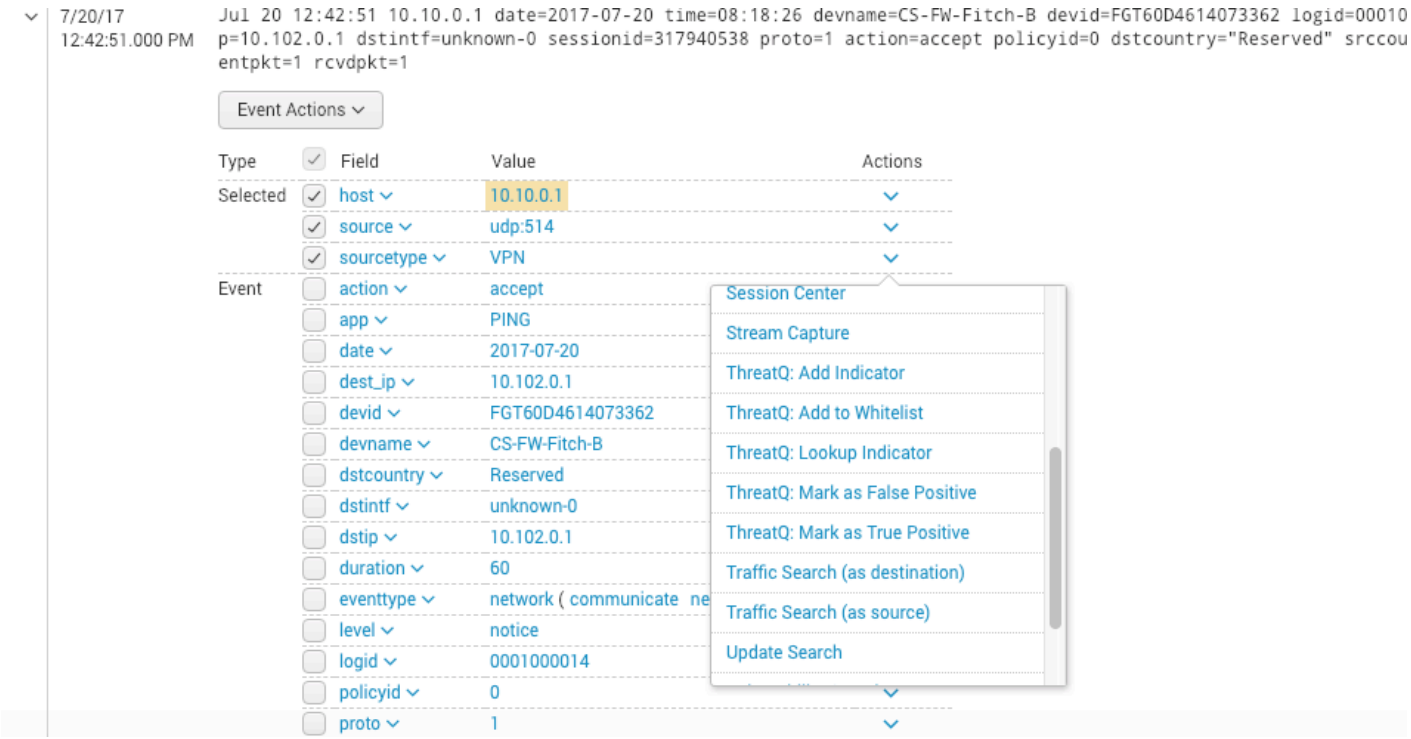
4. Simultaneously, another periodic saved search job `threatq_update_matched_ indicators` finds more sightings for all indicators from the match lookup table against all events in Splunk (as filtered by the same configurable macros).
5. A periodic saved search `threatq_consume_indicators` will create events in ThreatQ to represent evidence of sightings in Splunk.
6. 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.



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 down arrow in the column below **Action**.



The actions are described below.

ACTION	DESCRIPTION
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.

ACTION	DESCRIPTION
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.

CIM Support

The ThreatQuotient App for Splunk runs in the Datamodel Search mode when you are taking advantage of Splunk's CIM and mapping your logs and events to various data models provided by Splunk.

The following table summarizes how the matching algorithm will match specific data model fields to specific indicator types in ThreatQuotient.

ThreatQ indicator type to CIM field map for the matching algorithm

CIM DATA MODELS	DATA MODEL FIELDS	THREATQ INDICATOR TYPES MATCHED
Authentication	Authentication.src_user	Username
	Authentication.user	Username
Certificates	Certificates.All_Certificates.SSL.ssl_hash	SHA-1, SHA-256, SHA-384, SHA- 512
	Certificates.All_Certificates.SSL.ssl_issuer_email	Email Address
	Certificates.All_Certificates.SSL.ssl_subject_ email	Email Address
	Certificates.All_Certificates.SSL.ssl_subject_ common_name	String
	Certificates.All_Certificates.SSL.ssl_issuer_common_name	String
	Certificates.All_Certificates.SSL.ssl_subject_ organization	String
	Certificates.All_Certificates.SSL.ssl_issuer_ organization	String
	Certificates.All_Certificates.SSL.ssl_serial	String
	Certificates.All_Certificates.SSL.ssl_subject_ unit	String
	Certificates.All_Certificates.SSL.ssl_issuer_unit	String
Endpoint	Endpoint.Services.service	Service Name
	Endpoint.Processes.process_name	Service Name
	Endpoint.Filesystem.file_name	Filename
	Endpoint.Filesystem.file_hash	SHA-1, SHA-256, SHA-384, SHA-512
Email	Email.All_Email.file_name	Filename
	Email.All_Email.file_hash	SHA-1, SHA-256, SHA-384, SHA-512
	Email.All_Email.subject	Email Subject
	Email.All_Email.src_user	Email Address
Intrusion_Detection	Intrusion_Detection.IDS_Attacks.src	IP Address, IPv6 Address
	Intrusion_Detection.IDS_Attacks.signature	String
	Intrusion_Detection.IDS_Attacks.user	Username
Inventory	All_Inventory.User.user	Username
Malware	Malware.Malware_Attacks.file_name	Filename

CIM DATA MODELS	DATA MODEL FIELDS	THREATQ INDICATOR TYPES MATCHED
	Malware.Malware_Attacks.file_hash	SHA-1, SHA-256, SHA-384, SHA- 512
	Malware.Malware_Attacks.signature	String
	Malware.Malware_Attacks.sender	Email Address
	Malware.Malware_Attacks.src	IP Address, IPv6 Address
	Malware.Malware_Attacks.user	Username
Network_Traffic	Network_Traffic.All_Traffic.src	IP Address, IPv6 Address
Network Resolution (DNS)	Network_Resolution.DNS.query	FQDN, String
	Network_Resolution.DNS.answer	FQDN, String
Updates	Updates.Updates.file_name	Filename
	Updates.Updates.file_hash	SHA-1, SHA-256, SHA-384, SHA- 512
Web	Web.Web.user	Username
	Web.Web.http_referrer	URL
	Web.Web.url	URL
	Web.Web.http_user_agent	User-agent
	Web.Web.src	IP Address, IPv6 Address
	Web.Web.dest	IP Address, IPv6 Address
Incident_Management	Incident_Management.Notable_Events.src	IP Address, IPv6 Address
	Incident_Management.Suppressed_Notable_ Events.src	IP Address, IPv6 Address
	Incident_Management.Notable_Event_Suppressions. Suppression_Audit.signature	String
	Incident_Management.Notable_Event_Suppressions. Suppression_Audit_Expired.signature	String
	Incident_Management.Notable_Event_Suppressions. Suppression_Audit.user	Username

Enterprise Security Support

ThreatQ Indicators to Splunk Enterprise Security Lookup Tables

The ThreatQuotient App for Splunk 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 Indicator Type Mapping to Enterprise Security Lookup Tables

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

THREATQ TYPE	THREAT INTELLIGENCE TYPE
Registry Key	local_registry_intel
Service Name	local_service_intel
SHA-1	local_file_intel
SHA-256	local_file_intel
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

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

- **Threat Activity:** Shows the list of events 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.

Threat Intelligence data will be added to Enterprise Security using their REST APIs with a threat_key of `threatq_indicator`. The score for ThreatQ Indicators will be mapped to the **Weight** attribute in ES. Any updates to the score will be automatically reflected in ES using the periodic saved searches.

The indicator will be updated in ES and put into a disabled state (will no longer be used in further correlation) if the score or status of a ThreatQ indicator changes to a value that is no longer within the parameters configured in the macro settings for ThreatQ Splunk App.

If you are using ES and are upgrading from an older version of ThreatQ Splunk App, please run the "threatq_cleanup_es_lookups" saved search once to remove the old data. All the threat intelligence data is automatically added upon upgrade using the Enterprise Security's REST APIs

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 Enterprise Security

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.

Saved Searches for Mapping ThreatQ Indicator data to Splunk's CIM

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**
threatq_update_threat_intelligence_lookup_fqdn	Map ThreatQ **type 10** indicators to **local_domain_intel**
threatq_update_threat_intelligence_lookup_hash	Map ThreatQ **type [11,12,15,20,21,22,23]** indicators to **local_file_intel**

ES SAVED SEARCH	DESCRIPTION
threatq_update_threat_intelligence_lookup_ip	Map ThreatQ **type 14** indicators to **local_ip_intel**
threatq_update_threat_intelligence_lookup_registry	Map ThreatQ **type 18** indicators to **local_registry_intel**
threatq_update_threat_intelligence_lookup_service	Map ThreatQ **type 19** indicators to **local_service_intel**
threatq_update_threat_intelligence_lookup_certificate_serial	Map ThreatQ **type 25** indicators to **local_certificate_intel**
threatq_update_threat_intelligence_lookup_certificate_subject	Map ThreatQ **type 26** indicators to **local_certificate_intel**
threatq_update_threat_intelligence_lookup_url	Map ThreatQ **type 27** indicators to **local_http_intel**
threatq_update_threat_intelligence_lookup_user	Map ThreatQ **type 30** indicators to **local_user_intel**

Performance

The primary objective of this App is to find evidence of sightings and report those sightings back to ThreatQuotient. The sightings are discovered using the **matching algorithm** that works either in the **Raw Matching** or **Datamodel Matching** mode.

To summarize, the matching algorithm will take a set of indicators from ThreatQuotient, a set of events from Splunk, and find which indicators (and how many times) appear in the events.

> ⚠ The matching algorithm, by default, runs every 30 minutes in a saved search, so it is important that it completes in under 30 minutes on average just to keep up with incoming load.

The tables below demonstrate the performance of matching algorithm for both modes. These tables are meant to be used as guidelines so you can configure your App to run for an optimal performance.

It is advised that you experiment on your system to ensure that your system does not have data loaded at higher rates than is implied by the tables below. If the machine specs are different, it is advisable to first run the match queries in the Splunk's search bar and get a sense of how long it takes a typical query to finish. Once you find the right amount of data your installation can handle, you are advised to instrument the App in a way that it will only perform matching on the said amount of data.

Experiments

Experiements were conducted on a machine with 16 cores and 32 GB RAM. The parameters are total number of indicators from TQ, total events from Splunk, and number of indicators matched. Events were generated from various standard templates covering a wide range of firewall and web proxy logs. The bolded rows show the upper limit of performance in that the time to complete is slightly over 30 mins. We discovered that the **upper limit** was reached at around 1 million Splunk events and was largely invariant of number of indicators from ThreatQ (due to how this algorithm is implemented). As more matches are found, it takes more time to write them in the lookup tables, thus slightly increasing the runtime.

Raw Matching Performance Table

TOTAL INDICATORS FROM TQ	TOTAL RAW EVENTS IN SPLUNK	TOTAL INDICATORS MATCHED	TIME TO COMPLETE (S) MACHINE SPECS:(16 CORE, 32GB RAM)
100,000	500,000	0	885.36
100,000	500,000	10,000	899.75
100,000	**1,000,000**	**20,000**	**1,932.04**
500,000	**1,000,000**	**0**	**1,926.62**
500,000	**1,000,000**	**10000**	**2,020.56**
1,000,000	1,000,000	0	2,174.18
1,000,000	1,000,000	25,000	2,294.39
1,000,000	5,000,000	0	11,354.64
10,000	50,000,000	0	35,233.185 (9 hr 47 min)

Datamodel Matching Performance Table

Similar experiments were done for a **Datamodel Matching** case. From the table below, we determined that at around 15 million mark for Splunk events, the algorithm runtime started exceeding 30 minutes. **Thus, for a single saved search, this represents the upper limit of how much data this algorithm can handle every 30 minutes.**

TOTAL INDICATORS FROM TQ	TOTAL RAW EVENTS IN SPLUNK	TOTAL INDICATORS MATCHED	TIME TO COMPLETE (S) MACHINE SPECS:(16 CORE, 32GB RAM)
100,000	500,000	0	29.282
100,000	500,000	10,000	36.92
100,000	1,000,000	20,000	77.649
500,000	1,000,000	0	99.473
500,000	1,000,000	10000	130.991
1,000,000	1,000,000	0	166.517
1,000,000	1,000,000	25,000	261.362
1,000,000	5,000,000	0	420.111
100,000	5,000,000	10,000	619.047
1,000,000	10,000,000	10,000	1,316.541
1,000,000	**15,000,000**	**10,000**	**1,866.059**

TOTAL INDICATORS FROM TQ	TOTAL RAW EVENTS IN SPLUNK	TOTAL INDICATORS MATCHED	TIME TO COMPLETE (S) MACHINE SPECS:(16 CORE, 32GB RAM)
1,000,000	50,000,000	25,000	6,554.610

Scaling the App

The tables displayed in the Performance section offer a guideline of how many Splunk events the App can handle with default configuration. As found from the internal testing, the Raw Matching Performance table demonstrates the upper limit for **raw search** is about **1 million events/30 minutes**, and the same is **15 million events/30 minutes** for the **datamodel search** (on a dedicated box with 16 cores and 32 GB RAM). However, this is valid only for one saved search running on one node.

The best way to scale the App is to run multiple saved searches for matching. This can be achieved using the **datamodel search** mode. If your data is mapped to multiple Splunk data models from the CIM Support table, each data model is handled by a separate saved search. In such an instance, you would need to deploy your search head in a cluster and ensure that these saved searches are distributed in that cluster. You can run up to five of them, thus potentially scaling your App to handle five times the traffic.

For the raw matching mode, the App by default will only be able to run one saved search. In order to extend it to multiple searches, you will have to break apart this one saved search into multiple, and then, distribute these saved searches in the Splunk cluster of search heads. You can do this by running a separate saved search for:

- Splunk index for events
- ThreatQuotient indicator types.

For using a fixed Splunk index for the saved search, you can modify the default saved searches for matching as shown below.

Splunk Search for Listing TQ Indicators

```
index=<my_index> `threatq_match_sourcetypes` source- type!="threatq:indicators" |
threatqmatchiocs indicator_types- ='IP Address, FQDN'(threatq_match_indicators saved
search) index=<my_index> `threatq_match_sourcetypes` source- type!="threatq:indicators" |
threatqmatchiocs is_update=true (threatq_update_matched_indicators saved search)
```

Compare the above saved searches with the defaults as shown in the Save Search Documentation table. The macro `threatq_match_indices` is replaced by passing an actual index to the saved search. Now, you can make multiple copies of the default saved search, run them on the same schedule, and have each saved search get events from a different Splunk index.

To use the similar technique for ThreatQuotient indicator types, you can pass an additional argument to the `threatqmatchiocs` module as shown below. This allows you to make the saved search use only a specific indicator type. Again, as before, you can then make multiple copies of the saved searches and have each one handle only specific ThreatQuotient indicator types. You are free to pass a single indicator type, or a comma separated list as shown below.

Splunk Search for Listing TQ Indicators

```
index=<my_index> `threatq_match_sourcetypes` source- type!="threatq:indicators" |
threatqmatchiocs indicator_types- ='IP Address, FQDN'(threatq_match_indicators saved
search) index=<my_index> `threatq_match_sourcetypes` source- type!="threatq:indicators" |
threatqmatchiocs is_update=true indicator_types='IP Address,
FQDN'(threatq_update_matched_ indicators saved search)
```

Finally, both these techniques for scaling the App are equally applicable for the datamodel mode as well.

Dashboards

The Threat Dashboard and Indicator Dashboard are preconfigured dashboards packaged with ThreatQuotient App to allow the analyst versatile visual representation of all indicator data from ThreatQ and the corresponding sightings. These dashboards are only a suggestion and can be modified via Splunk's standard dashboard editing capability to meet your needs.


You can also access several shortcut options and search tools via the Info tab.

Threat Dashboards

The Threat Dashboard displays indicator sighting-related information such as:

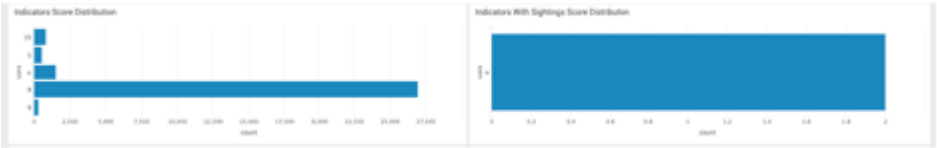
- Cumulative Counts
- Score Breakdown
- Type Breakdown
- Source Breakdown
- Adversaries Breakdown
- Static Table View
- Top 10 By Sightings

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.

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).



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.

Source Breakdown

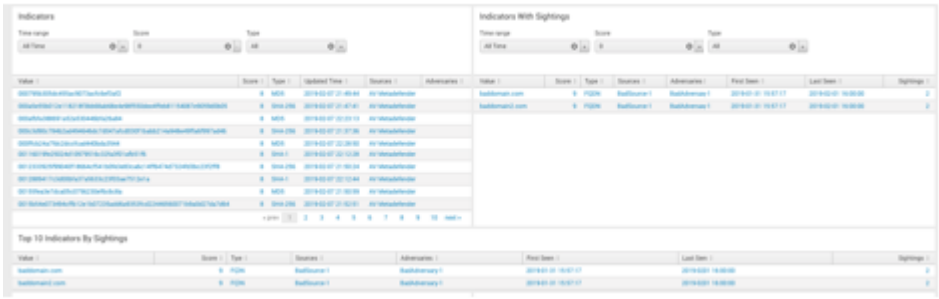
This section shows the breakdown of indicators and sighted indicators by sources. Example screenshot below. One thing to note here is that all indicators must have at least one source, but some indicators may have more than one. For this reason, the cumulative counts in the charts below may exceed the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

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.

Static View Table

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.

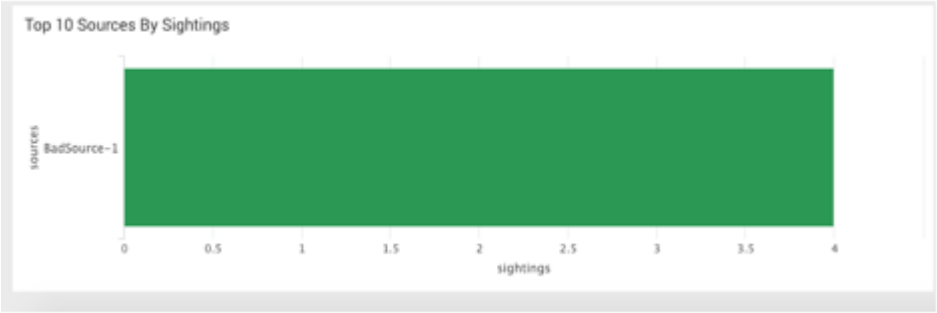


Top 10 By Sightings

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 indicator's sources and adversaries with the most matches within Splunk.

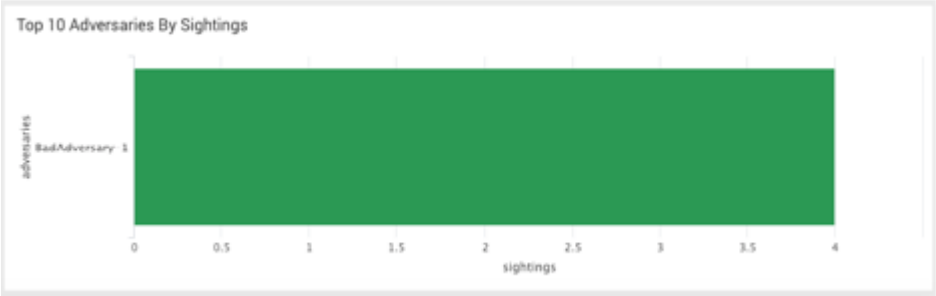
Sources

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.



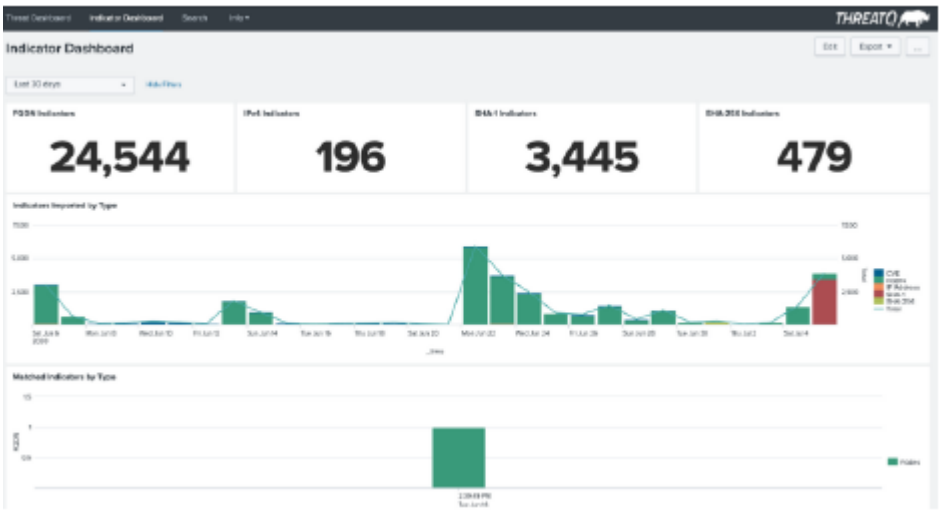
Adversaries

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.



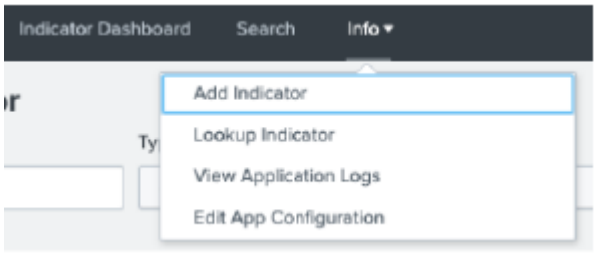
Indicator Dashboard

The Indicator Dashboard displays indicator-related widgets, such as type counts and bar charts, for a user-specified time frame.



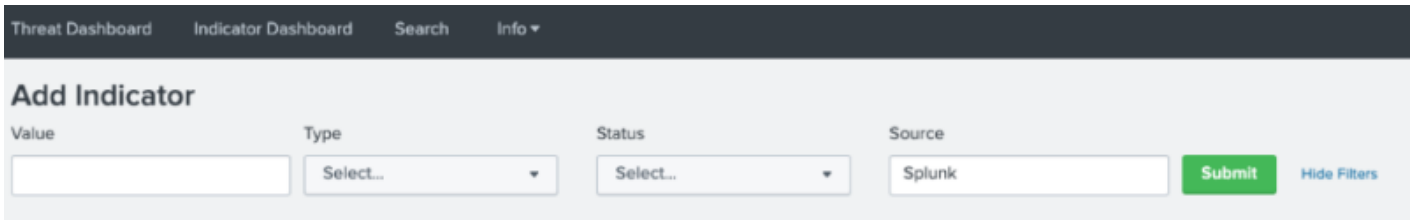
Info Tab

The Dashboard Info tab, located next to the Search Option, provides you with the ability to perform indicator and application log searches along with shortcuts to the Add Indicator and Edit App Configuration functions.



Add Indicator

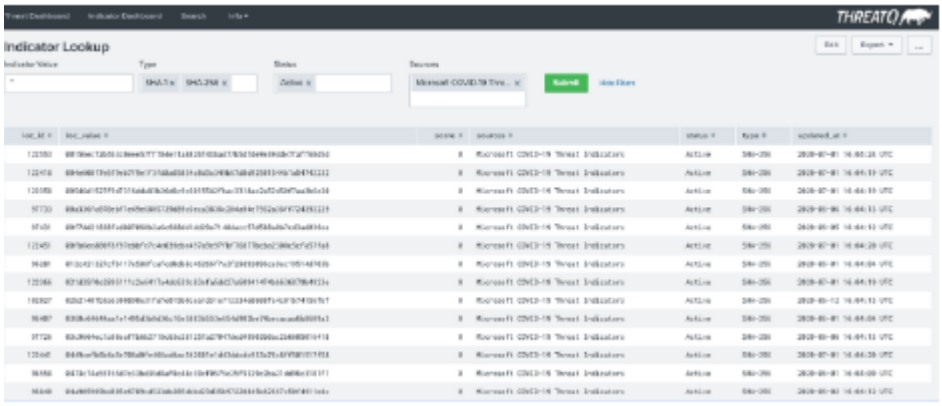
The Add Indicator option will open the Add Indicator input from within the dashboard. You can use this form to manually add indicators to ThreatQ.



Indicator Lookup

The Indicator Lookup option allows you to perform a search based on:

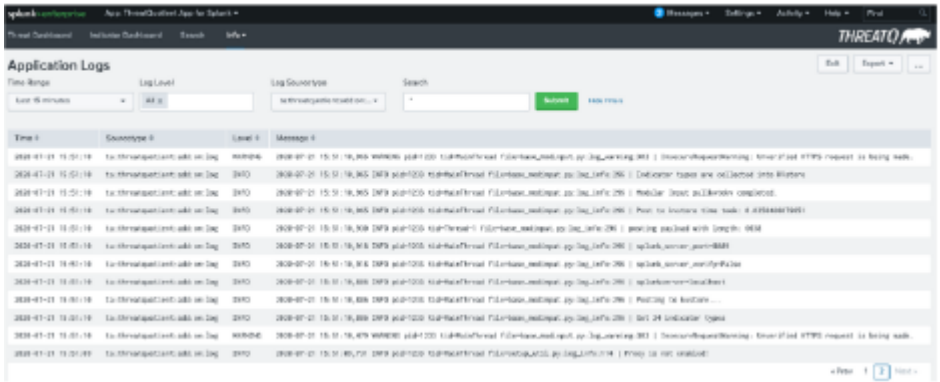
- IndicatorValue
- IndicatorType
- Status
- Source(s)



Application Log Search

The Application Log Search allows you to perform a search of logs based on:

- Time Range
- Log Level
- Log Source Type
- Search



Edit App Configuration

The Edit App Configuration open will open the app's Setup page. See the ThreatQuotient App section for more details.

Troubleshooting

- Use the log file below to troubleshoot the ThreatQuotient Add-on:

  ```
  $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 review the data collected by data collection, use a query such as:

  ```
  "index=your_index_name sourcetype=threatq_indicators"
  ```

  - Confirm all the saved searches are enabled.
  - Confirm the macro is updated as per the settings.
  - 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 sourcetype="splunkd_ui_access"
  ```

  ```
  index=_internal threatq_score_filter sourcetype="splunkd_access"
  ```

Change Log

- App Version 2.3.0
  - Fixed an issue which caused certain datamodel searches to not complete.
  - Fixed an issue where saved searches would fail if events had Chinese characters.
  - Upgraded the Splunklib.
- App Version 2.2.0
  - A Hostname configuration field has been added to the Setup page. This value will be used as a Source Attribute when calling consume endpoints.
  - Saved Searches have been staggered to prevent encountering concurrent search limitations.
  - Added a Malware family attribute field to the KVStore.
  - Added partial URL matching support for Datamodel searches.
  - Combined saved searches for Datamodel to have only a single search per Datamodel.
- TA-threatquotient-add-on: Version 2.3.0
  - Fixed an authentication issue with the KVStore configuration.
  - Malware family data, if available for ThreatQ indicators, will now be stored in the KVStore.
  - The localhost Username and Password dependency for the KVStore data collection has been removed.
- App Version 2.1.0
  - Added new Indicator Dashboard.
  - Added ability to use KVStore for saving data.
  - Added Info tab to dashboards page with the following options/shortcuts:
  - Add Indicator
  - Lookup Indicator
  - View Application Logs
  - Edit App Configurations
  - Fixed an issue where no sightings were generated for domain object types within Splunk.
  - Fixed an issue with data listed in multi-valued fields.

- TA-threatquotient-add-on: Version 2.2.0
  - Added new Splunk KVStore Rest configuration tab. This configuration tab is required if users save data to KVStore.
  - Additional options Enable Index and Pull all Indicators available under input configuration.
- TA-threatquotient-add-on: Version 2.1.0
  - Import timeout is now configurable from UI
  - Pagination support for initial import of ThreatQ data
  - Updated default frequency for ThreatQ Exports from 300 to 900
- App Version 2.0.0
  - Python 3 Support - ThreatQuotient App for Splunk is now compatible with Python 3. Supported versions include:
    - Splunk 7.2.x
    - Splunk 7.3.x
    - Splunk 8.X (Python 2)
    - Splunk 8.X (Python 3)
- TA-threatquotient-add-on: Version 2.0.0
  - Python 3 Support - ThreatQuotient Add-on for Splunk is now compatible with Python 3.

    Supported versions include:
    - Splunk 7.2.x
    - Splunk 7.3.x
    - Splunk 8.X (Python 2)
    - Splunk 8.X (Python 3)
  - Resolved an issue where creating an indicator in Splunk would occasionally result in the creation of an indicator with an incorrect type within the ThreatQ platform.
- App Version 1.3.0
  - Threat Intelligence support for Enterprise Security is now provided using its REST APIs
- App Version 1.2.0
  - Added the following contextual data to Indicators:
    - Splunk Sighting Timestamp - Last seen value
    - Match Count
    - The Source for sighted indicators is now reported as Splunk in ThreatQ.
  - Added Macro Configuration option to App Setup page. Users now have the ability to select indices, the location they want to search.

> Note: If you have the macro configuration for threatq_match_indices set to *, you will need to update the app configuration upon upgrade to 1.2.0 and add the required indexes where matching should take place with ThreatQ indicators. This step is mandatory for the app to continue to perform matching against the required indexes.

  - Added Sighting Event Configuration option to the App Setup page. Users now have the ability to configure how the app create events for a sighted indicator.
  - Added a new Saved Search - threatq_consume_indicators_new

- **TA-threatquotient-add-on: Version 1.1.2**
  - Certificate-based errors will no longer appear in the Splunk log. They will now be added as a warning in the ThreatQ application log.
- **TA-threatquotient-add-on: Version 1.1.1**
  - We have fixed an issue where Splunk credential parsing was generating a 500 error and leaving the configuration page in an unusable state.
- **App Version 1.1.0**
  - The ThreatQuotient Splunk integration now includes support for the Common Information Model (CIM). For users who map third party data (firewall events, logs, for example) to Splunk's data models in CIM, this App provides optimized performance by leveraging those data models. As such, we now support the CIM Data Model Search. We have enhanced Enterprise Security (ES) support to provide single-click enablement within the ThreatQ App for Splunk application settings.
  - We have fixed issues where:
    - Users could not re-enable and use searches without crashing Splunk ES search head.threatq_match_indicators searches failed to complete. All saved search queries for matching can now accept an optional argument called indicator_types that allows users to match only specific indicator types from ThreatQ.
- **App Version 1.0.1**
  - During authentication, users can now specify whether to verify or disable the SSL certificate.