HCP Prioritizing Threat Intelligence 1

Prioritizing Threat Intelligence

Date of Publish: 2019-04-09

https://docs.hortonworks.com

Contents

Understanding Threat Triage Rule Configuration

Not all threat intelligence indicators are equal. Some require immediate response, while others can be addressed as time and availability permits. As a result, you must triage and rank threats by severity. The goal of threat triage is to prioritize the alerts that pose the greatest threat and need urgent attention. To create a threat triage rule configuration, you must first define your rules.

In HCP, you assign severity by associating possibly complex conditions with numeric scores. Then, for each message, you use a configurable aggregation function to evaluate the set of conditions and to aggregate the set of numbers for matching conditions This aggregated score is added to the message in the threat.triage.level field.

Each rule has a predicate to determine whether or not the rule applies. The threat score from each applied rule is aggregated into a single threat triage score that is used to prioritize high risk threats.

Following are some examples:

Rule 1

If a threat intelligence enrichment type zeusList is alerted, imagine that you want to receive an alert score of 5.

Rule 2

If the URL ends with neither .com nor .net, then imagine that you want to receive an alert score of 10.

Rule 3

For each message, the triage score is the maximum score across all conditions.

These example rules become the following example configuration:

```
“triageConfig” : {
   “riskLevelRules” : [
{
“name” : “zeusList is alerted"
“comment” : “Threat intelligence enrichment type zeusList is alerted."
“rule”:
 "exists(threatintels.hbaseThreatIntel.domain_without_subdomains.zeusList)”
“score” : 5
}
{
“name” : “Does not end with .com or .net"
“comment” : “The URL ends with neither .com nor .net."
“rule”: “not(ENDS_WITH(domain_without_subdomains, ‘.com’) or
 ENDS_WITH(domain_without_subdomains, ‘.net’))“ : 10
“score” : 10
}
]
      ,“aggregator” : “MAX”
       ,”aggregationConfig” : { }
}
```

You can use the 'reason' field to generate a message explaining why a rule fired. One or more rules may fire when triaging a threat. Having detailed, contextual information about the environment when a rule fired can greatly assist actioning the alert. For example:

Rule 1

For hostname, the value exceeds threshold of value-threshold, receive an alert score of 10.

This example rule becomes the following example configuration:

```
“triageConfig” : {
   “riskLevelRules” : [
      {
      “name” : “Abnormal Value"
      “comment” : “The value has exceeded the threshold",
      "reason": "FORMAT('For '%s' the value '%d' exceeds threshold of '%d',
 hostname, value, value_threshold)"
      “rule”: "value > value_threshold”,
      “score” : 10
      }
   ],
   “aggregator” : “MAX”,
   ”aggregationConfig” : { }
}
```

If the value threshold is exceeded, Threat Triage will generate a message similar to the following:

```

"threat.triage.score": 10.0,
"threat.triage.rules.0.name": "Abnormal Value",
"threat.triage.rules.0.comment": "The value has exceeded the threshold",
"threat.triage.rules.0.score": 10.0,
"threat.triage.rules.0.reason": "For '10.0.0.1' the value '101' exceeds
 threshold of '42'"
```

where

riskLevelRules

This is a list of rules (represented as Stellar expressions) associated with scores with optional names and comments.

- **name** The name of the threat triage rule.
- **comment** A comment describing the rule.
- **reason** An optional Stellar expression that when executed results in a custom message describing why the rule fired.
- **rule** The rule, represented as a Stellar statement.
- **score** Associated threat triage score for the rule.

aggregator

An aggregation function that takes all non-zero scores representing the matching queries from riskLevelRules and aggregates them into a single score.

You can choose between:

MAX	The maximum of all of the associated values for matching queries.
MIN	The minimum of all of the associated values for matching queries.
MEAN	the mean of all of the associated values for matching queries.
POSITIVE_MEAN	The mean of the positive associated values for the matching queries.

Configure Basic Threat Triage Rules

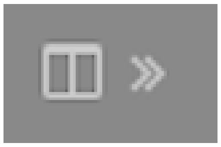
You can use the Threat Triage field in the Management UI to assign basic threat triage rules and scores. To specify more granular triage rules, you need to specify the information with the CLI or the Advanced JSON field in the Management UI.

Before you begin

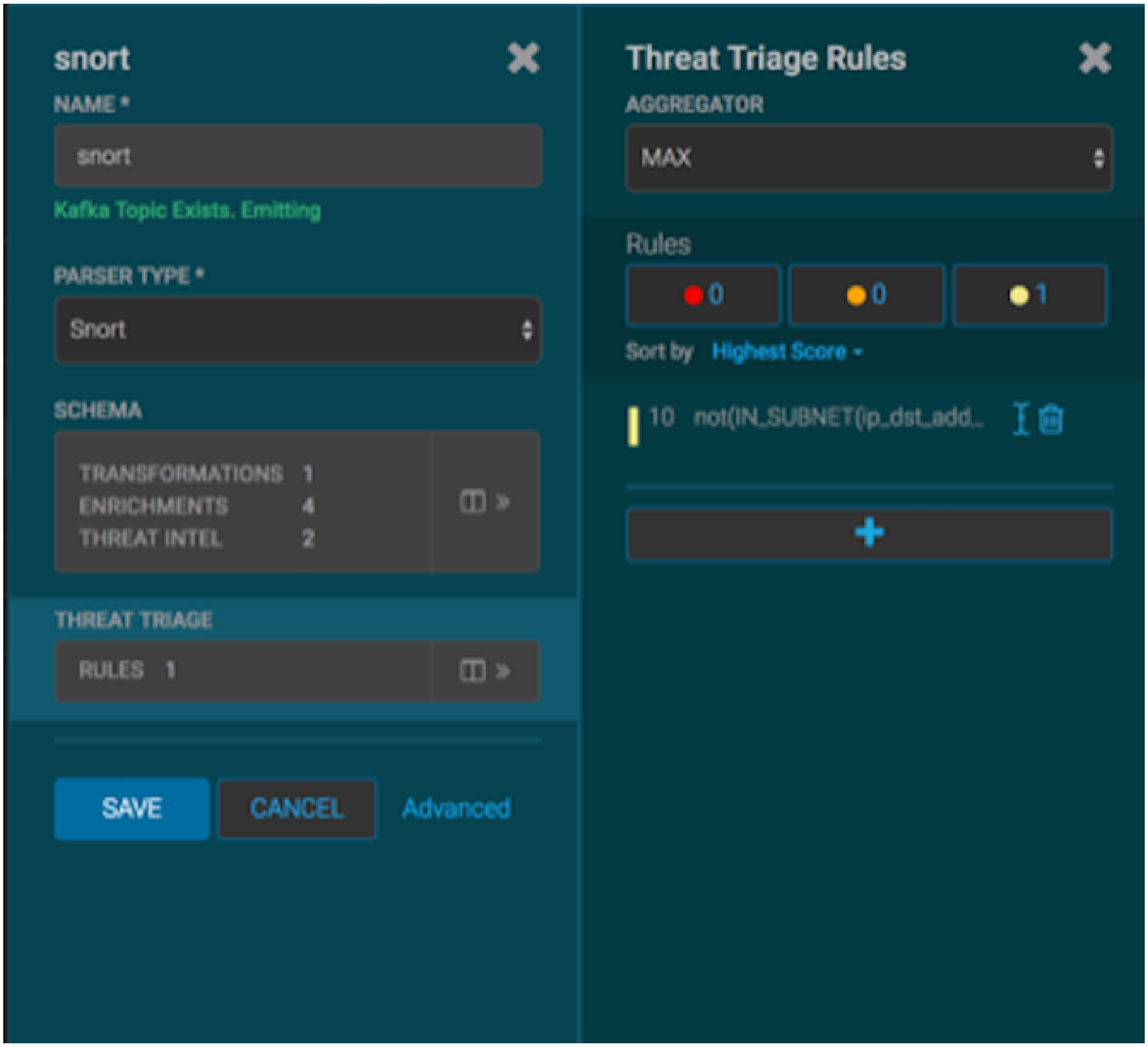
Ensure that the enrichment is working properly.

Procedure

1. On the sensor panel, in the Threat Triage field, click



.



snort

NAME *

snort

Kafka Topic Exists. Emitting

PARSER TYPE *

Snort

SCHEMA

TRANSFORMATIONS	1
ENRICHMENTS	4
THREAT INTEL	2

THREAT TRIAGE

RULES 1

SAVE CANCEL Advanced

Threat Triage Rules

AGGREGATOR

MAX

Rules

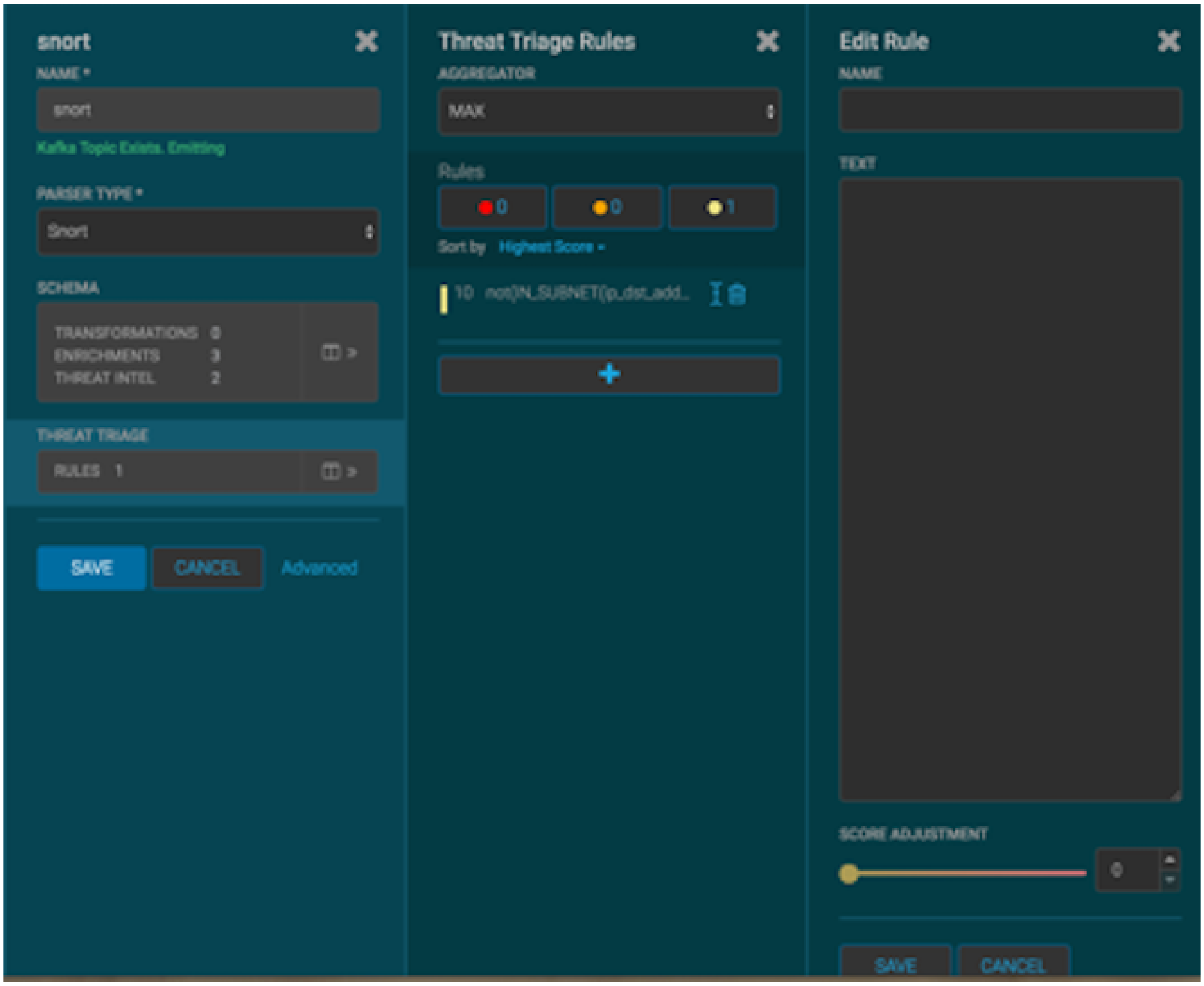
● 0 ● 0 ● 1

Sort by Highest Score ▾

10 not(IN_SUBNET(ip_dst_add...

+

2. To add a rule, click +.



3. Assign a name to the new rule in the NAME field.
4. In the Text field, enter the syntax for the new rule:

```
Exists(IsAlert)
```

5. Use the **SCORE ADJUSTMENT** slider to choose the threat score for the rule.
6. Click **SAVE**.

   The new rule is listed in the Threat Triage Rules panel.
7. Choose how you want to aggregate your rules by choosing a value from the Aggregator menu.

   You can choose among the following:

MAX	The maximum of all of the associated values for matching queries.
MIN	The minimum of all of the associated values for matching queries.
MEAN	The mean of all of the associated values for matching queries.
POSITIVE_MEAN	The mean of the positive associated values for the matching queries.

8. If you want to filter threat triage display, use the **Rules** section and the **Sort by** menu below it.

   For example, to display only high-levels alerts, click the box containing the red indicator. To sort the high-level alerts from highest to lowest, select **Highest Score** from the **Sort by** menu.
9. Click **SAVE**.

Configure Advanced Threat Triage Rules

To assign more detailed threat triage rules and scores, you must use the CLI or the Advanced JSON field in the Management user interface.

Procedure

1. Determine the rules you want to implement to prioritize alerts using the configuration guidelines provided in Understanding Threat Triage Rule Configuration.
2. Modify the configuration for the sensor in the enrichment topology.

   For example:

   ```
   “triageConfig” : {
      “riskLevelRules” : [
   {
   “name” : “zeusList is alerted"
   “comment” : “Threat intelligence enrichment type zeusList is alerted."
   “rule”:
    "exists(threatintels.hbaseThreatIntel.domain_without_subdomains.zeusList)”
   “score” : 5
   }
   {
   “name” : “Does not end with .com or .net"
   “comment” : “The URL ends with neither .com nor .net."
   “rule”: “not(ENDS_WITH(domain_without_subdomains, ‘.com’) or
    ENDS_WITH(domain_without_subdomains, ‘.net’))“ : 10
   “score” : 10
   }
   ]
         ,“aggregator” : “MAX”
          ,”aggregationConfig” : { }
   }
   ```
3. Log in as root user to the host on which Metron is installed.
4. Modify $METRON_HOME/config/zookeeper/sensors/$DATASOURCE.json to match the configuration on disk:

   Because the configuration in ZooKeeper might be out of sync with the configuration on disk, ensure that they are in sync by downloading the ZooKeeper configuration first:

   ```
   $METRON_HOME/bin/zk_load_configs.sh -m PULL -z $ZOOKEEPER_HOST:2181 -f -o
    $METRON_HOME/config/zookeeper
   ```
5. Validate that the enrichment configuration for the data source exists:

   ```
   cat $METRON_HOME/config/zookeeper/enrichments/$DATASOURCE.json
   ```
6. In the $METRON_HOME/config/zookeeper/enrichments/$DATASOURCE.json file, add the following to the triageConfig section in the threat intelligence section:

   ```
   "threatIntel" : {
       "fieldMap" : {
         "hbaseThreatIntel" : [ "domain_without_subdomains" ]
       },
   ```

```
     "fieldToTypeMap" : {
       "domain_without_subdomains" : [ "zeusList" ]
     },
     "config" : { },
     "triageConfig" : {
       "riskLevelRules" : {

  "exists(threatintels.hbaseThreatIntel.domain_without_subdomains.zeusList)" :
  5
               , "not(ENDS_WITH(domain_without_subdomains, '.com') or
  ENDS_WITH(domain_without_subdomains, '.net'))" : 10
                                 }
         ,"aggregator" : "MAX"
         ,"aggregationConfig" : { }
                            }
                        }
   }
```

7. Ensure that the aggregator field indicates MAX.
8. Push the configuration back to ZooKeeper:

```
$METRON_HOME/bin/zk_load_configs.sh -m PUSH -z $ZOOKEEPER_HOST:2181 -i
 $METRON_HOME/config/zookeeper
```

View Triaged or Scored Alerts

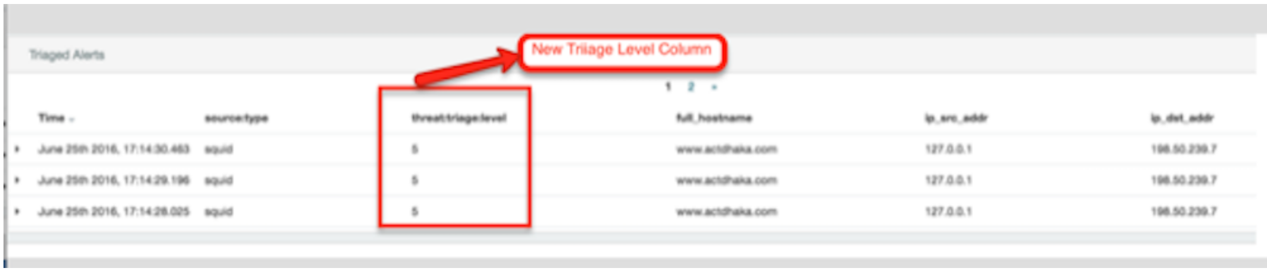
You can view triaged alerts in the indexing topic in Apache Kafka or in the triaged alert panel in the HCP Metron dashboard.

An alert in the indexing topic in Kafka looks similar to the following:

```
> THREAT_TRIAGE_PRINT(conf)
#######################################################################################
# Name # Comment # Triage Rule # Score # Reason #
#######################################################################################
# Abnormal DNS Port # # source.type == "bro" and protocol == "dns" and
 ip_dst_port != 53 # 10 # FORMAT("Abnormal DNS Port: expected: 53, found:
 %s:%d", ip_dst_addr, ip_dst_port) #
##########################################################
```

The following shows you an example of a triaged alert panel in the HCP Metron dashboard

Investigation Module Triaged Alert Panel



Triaged Alerts

New Triiage Level Column

1 2 »

	Time	source:type	threat:triage:level	full_hostname	ip_src_addr	ip_dst_addr
▸	June 25th 2016, 17:14:30.463	squid	5	www.actdhaka.com	127.0.0.1	198.50.239.7
▸	June 25th 2016, 17:14:29.196	squid	5	www.actdhaka.com	127.0.0.1	198.50.239.7
▸	June 25th 2016, 17:14:28.025	squid	5	www.actdhaka.com	127.0.0.1	198.50.239.7

For URLs from cnn.com, no threat alert is shown, so no triage level is set. Notice the lack of a **threat.triage.level** field.