Triage Squid Alerts Using Typosquatting Algorithm

Date of Publish: 2019-3-18

Contents

Triage Squid Events

Security event triage rules determine which events require further follow up and which events can be archived without further investigation. HCP processes many events every day so effective triage helps analysts focus on the most important events.

The two components of security event triage are:

• Determine if the event is an alert.
• If the event is an alert, assign a score. If the event is not an alert, it is not scored.

Triage Squid Using the Typosquatting Algorithm

For this example, we use a simple triage rule to detect typosquatting. Typosquatting uses common domain misspellings to install malicious web content.

Procedure

1. Determine the number of possible typosquat permutations.

   To configure the Bloom filter you need to specify roughly how many elements are going into the Bloom filter and what kind of false positive probability you want. You can use the CONSOLE output mode of the flatfile_summarizer.sh to count the number of typosquatted domains across the entire document.

   a) Create an extractor_count.json file at $METRON_HOME/config and populate it with the following:

```
{
  "config" : {
    "columns" : {
        "rank" : 0,
        "domain" : 1
    },
    "value_transform" : {
        "domain" : "DOMAIN_REMOVE_TLD(domain)"
    },
    "value_filter" : "LENGTH(domain) > 0",
    "state_init" : "0L",
    "state_update" : {
        "state" : "state + LENGTH( DOMAIN_TYPOSQUAT( domain ))"
                        },
    "state_merge" : "REDUCE(states, (s, x) -> s + x, 0)",
    "separator" : ","
  },
  "extractor" : "CSV"
}
```

   where

columns	Indicates the schema of the CSV. There are two columns, rank at the first position and domain at the second position.
separator	Use a comma to separate the columns.
value_transform	For each row, transform each domain column by removing the TLD.
value_filter	Only consider non-empty domains.
state_init	Initialize the state, a long integer, to 0.

state_update	For each row in the CSV, update the state, which is the running partial sum, with the number of typosquatted domains for the domain.
state_merge	For each thread, we have a partial sum, we want to merge the partial sums into the total.

b) Run the extractor_count.json file:

```
$METRON_HOME/bin/flatfile_summarizer.sh -i ~/top-10k.csv -e ~/
extractor_count.json -p 5 -om CONSOLE
```

The output should look similar to the following:

```
WARN extractor.TransformFilterExtractorDecorator: Unable to setup
 zookeeper client - zk_quorum url not provided. **This will limit some
 Stellar functionality**

Processing /root/top-10k.csv
17/12/22 17:05:20 WARN resolver.BaseFunctionResolver: Using System
 classloader
Processed 9999 - \
3496552
```

2. Generate the Bloom filter on HDFS.

a) Create an extractor_filter.json file at $METRON_HOME/config and populate it with the following:

```
{
  "config" : {
    "columns" : {
        "rank" : 0,
        "domain" : 1
    },
    "value_transform" : {
        "domain" : "DOMAIN_REMOVE_TLD(domain)"
    },
    "value_filter" : "LENGTH(domain) > 0",
    "state_init" : "BLOOM_INIT(3496552, 0.001)",
    "state_update" : {
        "state" : "REDUCE( DOMAIN_TYPOSQUAT( domain ), (s, x) ->
  BLOOM_ADD(s, x), state)"
                      },
    "state_merge" : "BLOOM_MERGE(states)",
    "separator" : ","
  },
  "extractor" : "CSV"
}
```

Most of the parameters are same as the extractor_count.json file, but there are three different parameters:

state_init	We have changed our state to be a bloom filter, initialized with:
	3496552 - The size calculated in the previous step
	0.001 - The false positive probability (0.1%)
state_update	Update the bloom filter (the state variable) with each typosquatted domain,
state_merge	Merge the bloom filters generated per thread into a final, single bloom filter to be written.

4

    b) Generate the Bloom filter in HDFS at /tmp/reference/alexa10k_filter.ser:

```
$METRON_HOME/bin/flatfile_summarizer.sh -i ~/top-10k.csv -o /tmp/
reference/alexa10k_filter.ser -e ~/extractor_filter.json -p 5 -om HDFS
```

3. Apply your new filter to domains from the squid telemetry.

    a) Display the Management UI.

    b) Select the Squid sensor from the list of sensors on the main window.

    c) Click the pencil icon in the list of tool icons

for the sensor.

The Management UI displays the Squid sensor panel.

    d) Click the **Advanced** button.

    e) Click

(expand window) next to the **RAW JSON** field.

    f) Replace the JSON information in the **SENSOR ENRICHMENT CONFIG** section with the following JSON information:

```
{
  "enrichment": {
   "fieldMap": {
    "geo": [
     "ip_dst_addr"
    ],
    "stellar": {
     "config": [
      "domain_without_tld :=
 DOMAIN_REMOVE_TLD(domain_without_subdomains)",
      "is_potential_typosquat := BLOOM_EXISTS(OBJECT_GET('/tmp/reference/
alexa10k_filter.ser'), domain_without_tld)",
      "domain_without_tld := null"
     ]
    }
   },
   "fieldToTypeMap": {},
   "config": {}
  },
  "threatIntel": {
   "fieldMap": {
    "stellar": {
     "config": [
      "is_alert := (exists(is_alert) && is_alert) ||
  is_potential_typosquat"
     ]
    }
   },
   "fieldToTypeMap": {},
   "config": {},
   "triageConfig": {
    "riskLevelRules": [
      {
       "name": "Alexa 10k Typosquat Bloom",
```

```
      "comment": "Inspect a bloom filter with potentially typosquatted
 domains from the top Alexa 10k",
      "rule": "is_potential_typosquat != null && is_potential_typosquat",
      "score": 50,
      "reason": "FORMAT('%s is a potential typosquatted domain from the
 top 10k domains from alexa', domain_without_subdomains)"
    }
  ],
  "aggregator": "MAX",
  "aggregationConfig": {}
 }
 },
 "configuration": {}
}
```
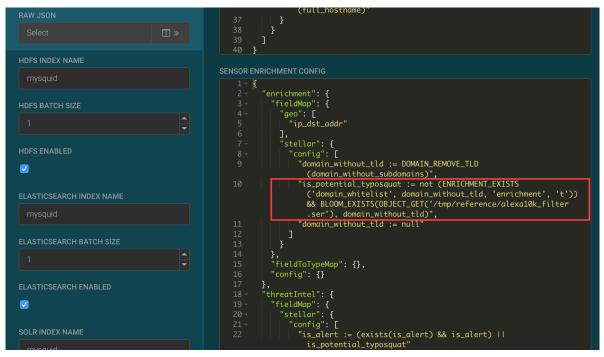

g) Click **SAVE** below the JSON information.

h) Click **SAVE** at the bottom of the Squid sensor configuration panel.

4. After you identify a potential typosquatted domain, investigate it, and determined that it is legitimate, you can stop future alerts by using a domain whitelist enrichment.

a) In the Management UI, click the pencil icon next to the mysquid sensor.

The Management UI displays the sensor configuration form.

b) Click the **Advanced** button.

c) Click

(expand window button) next to the **RAW JSON** field.

d) Replace the **is_potential_typosquat** field value with the following:

```
"is_potential_typosquat := not (ENRICHMENT_EXISTS('domain_whitelist',
 domain_without_tld, 'enrichment', 't')) && BLOOM_EXISTS(OBJECT_GET('/
 tmp/reference/alexa10k_filter.ser'), domain_without_tld)",
```

e) Click **SAVE** below the JSON information.

f) Click **SAVE** at the bottom of the Squid sensor configuration panel.

5. Ensure that the results appear in the Alerts UI.

a) Enter cnn.com or nsp.com in the browser connected to the HCP proxy.

b) Display the Alerts UI.

In the Score column, you should see events with non-zero scores and the **is_alert** field set to **true**.

If you want to view the columns as they appear in the screen shot, click the gear icon to the left of the **Actions** button and unselect all fields except **Score**, **id**, **timestamp**, **source:type**, **domain_withoutsub_domains**, and **is_alert** fields, then click **Save**.

c) Click the **Score** header to sort the events ascending by Score.

Click again to sort descending by Score. A downward arrow appears next to the **Score** header when sorted descending by Score.

d) Click between the columns of one of the Scored alerts to view the alert details.

The fields beginning with **threat:triage:rules** show the results of all the triage rules. The **threat:triage:score** field is the aggregated score of the event. If there is more than one triage rule, this field will contain the score combining the results from all the rules. The **is_alert** field is set only if the triage rules indicate the event is an alert.
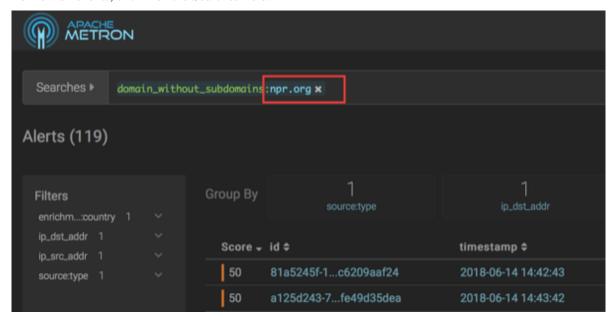
e) To see all the alerts for a particular domain, click the domain name.

The Alerts UI displays only the alerts with the selected domain name.

f) To remove a filter, click **x** next to the filter.

To view all events, click **x** on the Searches field.

Improve Scoring with a Domain Whitelist

Once you have identified and investigated a potential typosquatted domain and found that it is legitimate, you can stop future alerts by using a domain whitelist enrichment.

Procedure

1. Display the Management module UI.
2. Select the Squid sensor from the list of sensors on the main window.

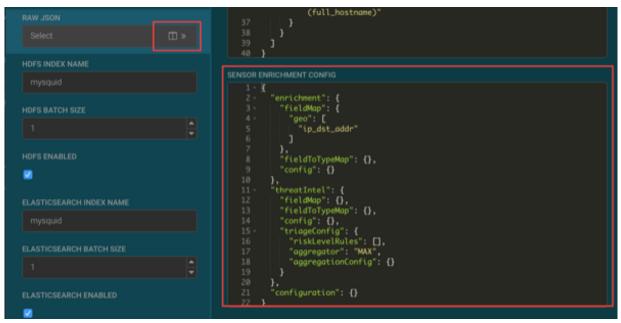**3.** Click the pencil icon in the list of tool icons

for the Squid sensor.
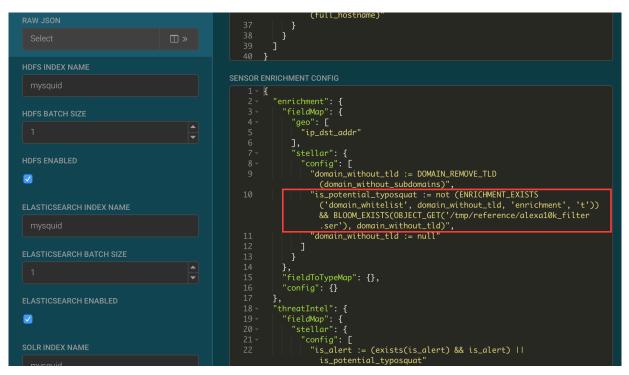
4. Click **Advanced**.

5. Click

(expand window button) next to the **RAW JSON** field.

6. Replace the is_potential_typosquat information with the following:

```
"is_potential_typosquat := not (ENRICHMENT_EXISTS('domain_whitelist',
 domain_without_tld, 'enrichment', 't')) && BLOOM_EXISTS(OBJECT_GET('/tmp/
reference/alexa10k_filter.ser'), domain_without_tld)",
```

7. Click **SAVE** below the JSON panel.

8. Click **SAVE** at the bottom of the Squid sensor configuration panel.

9. Open cnn.com or npr.com in the browser connected to the HCP proxy.

10. Open the Alerts UI.

11. Click on the **timestamp** column header until the events are sorted descending by timestamp.

   Proxy events to cnn.com and npr.org are no longer alerts.