Blue Team: Summary of Operations

Table of Contents

Network Topology

The following machines were identified on the network:

[Target 1]

- Operating System: **Linux 3.2 -4.9**
- Purpose: **Raven Web server**
- IP Address: 198.168.1.110

[Target 2]

- Operating System: **Linux 3.2 - 4.9**
- Purpose: **Raven Web server**
- IP Address: **198.168.1.115**

[Capstone]

- Operating System: **Linux**
- Purpose: **Logging Metricbeats and Filebeats**
- IP Address: **198.168.1.105**

[JumpBox]

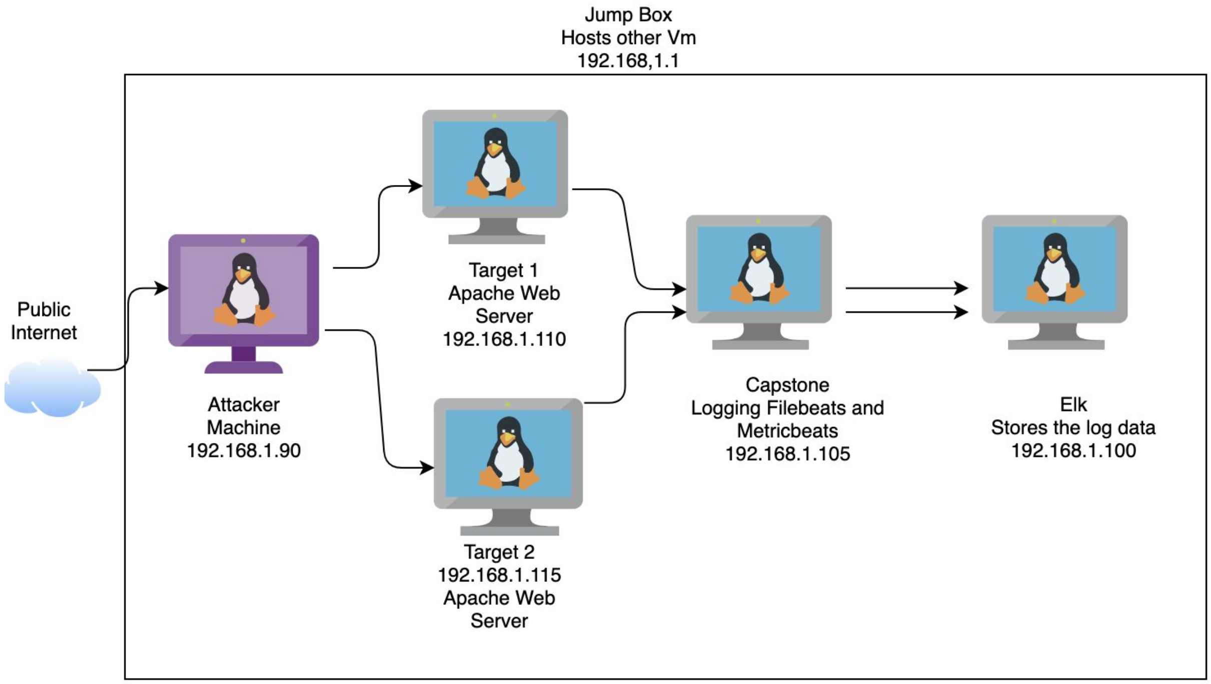
- Operating System: **Microsoft Windows 10 Pro**
- Purpose: **Host of all Virtual Machines**
- IP Address: **198.168.1.1**

[Elk]

- Operating System: **Linux 3.2 - 4.9**
- Purpose: **Collects the data from filebeats and Metricbeats**
- IP Address: **198.168.1.100**

[Kali Linux]

- Operating System: Linux 2.6.32
- Purpose: The Attacking Box
- IP Address: 198.168.1.90

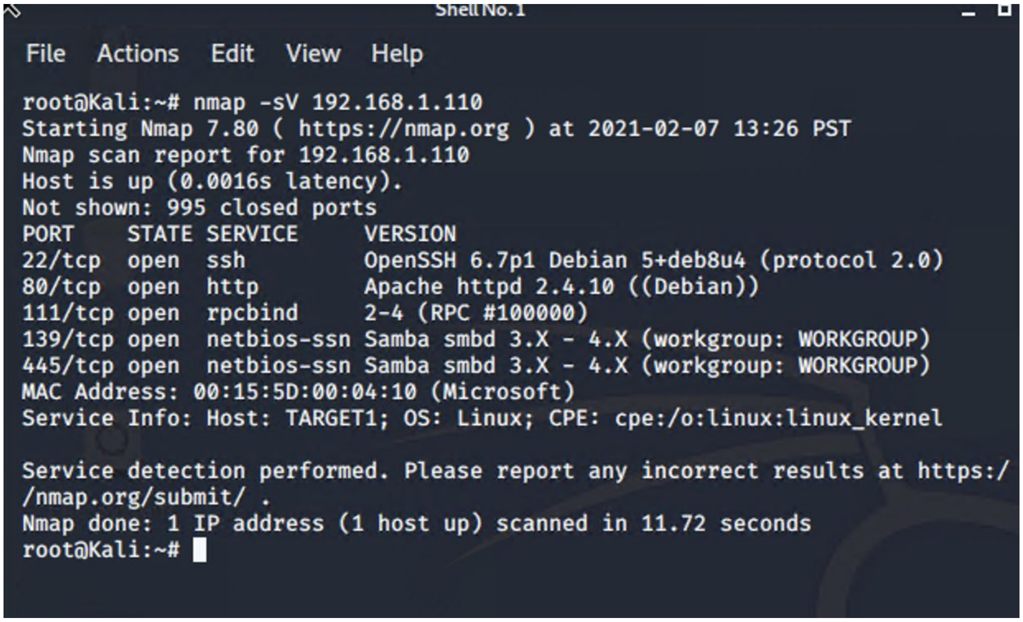

Jump Box
Hosts other Vm
192.168,1.1
Public Internet
Attacker Machine 192.168.1.90
Target 1 Apache Web Server 192.168.1.110
Target 2 192.168.1.115 Apache Web Server
Capstone Logging Filebeats and Metricbeats 192.168.1.105
Elk Stores the log data 192.168.1.100

Description of Targets

Fill in the following:

- Two VMs on the network were vulnerable to attack: Though there are two targets **Target 1** 198.168.1.110, and Target 2. Target 1 will be the main focus.
- Target 1 was hosted on a web server called Raven Security. By running a scan on the versions of each virtual machine we were able to gain information about the vulnerabilities.
- We found that the Target 1 and 2 were both susceptible to SSH attacks and HTTP attacks.
- When running the same attacks we found **Target 2** 198.168.1.115 is susceptible to the same attacks as **Target 1**

- Each VM functions as an Apache web server and has SSH enabled, so ports `80` and `22` are possible ports of entry for attackers.



```
root@Kali:~# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2021-02-07 13:26 PST
Nmap scan report for 192.168.1.110
Host is up (0.0016s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https:/
/nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.72 seconds
root@Kali:~# 
```

Monitoring the Targets

This scan identifies the services below as potential points of entry:

- **Raven Local 1**
    - OpenSSH
    - Apache HTTP 2.410 Debian
    - Samba smbd 3.X - 4.X

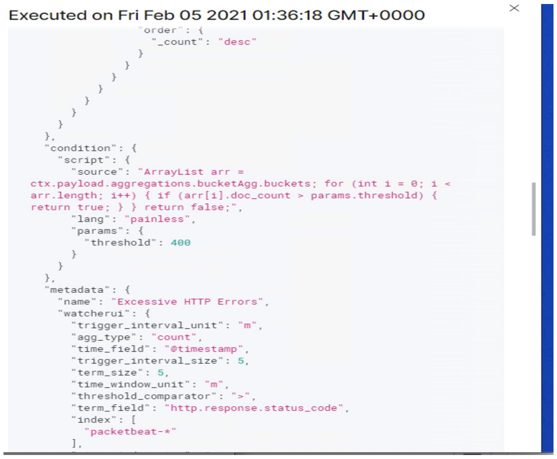
- **Raven Local 2**
    - OpenSSH
    - Apache HTTP 2.410 Debian
    - Samba SMBD 3.X - 4.X

Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

Excessive HTTP Errors

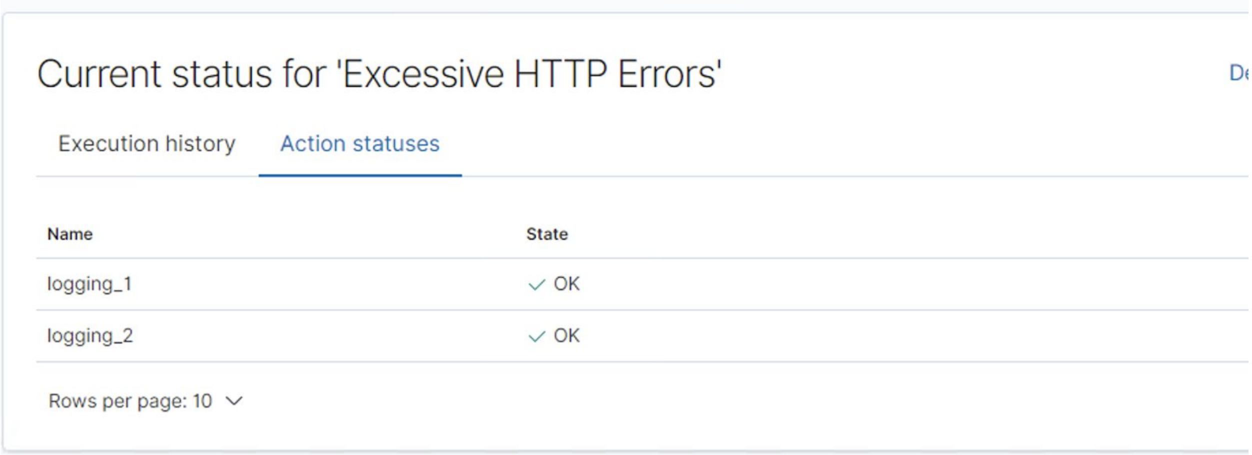
Excessive HTTP Errors is implemented as follows:

- Metric: **Packetbeat**
- Threshold: **HTTP status codes fires above 400 for the last 5 minutes**
- Vulnerability Mitigated: **Bad Request/ Requests Timed out**
- Reliability: This does not generate many false positives. I would rate this low



Executed on Fri Feb 05 2021 01:36:18 GMT+0000

```
                "order": {
                  "_count": "desc"
                }
              }
            }
          }
        }
      }
    }
  },
  "condition": {
    "script": {
      "source": "ArrayList arr =
ctx.payload.aggregations.bucketAgg.buckets; for (int i = 0; i <
arr.length; i++) { if (arr[i].doc_count > params.threshold) {
return true; } } return false;",
      "lang": "painless",
      "params": {
        "threshold": 400
      }
    }
  },
  "metadata": {
    "name": "Excessive HTTP Errors",
    "watcherui": {
      "trigger_interval_unit": "m",
      "agg_type": "count",
      "time_field": "@timestamp",
      "trigger_interval_size": 5,
      "term_size": 5,
      "time_window_unit": "m",
      "threshold_comparator": ">",
      "term_field": "http.response.status_code",
      "index": [
        "packetbeat-*"
      ],
```



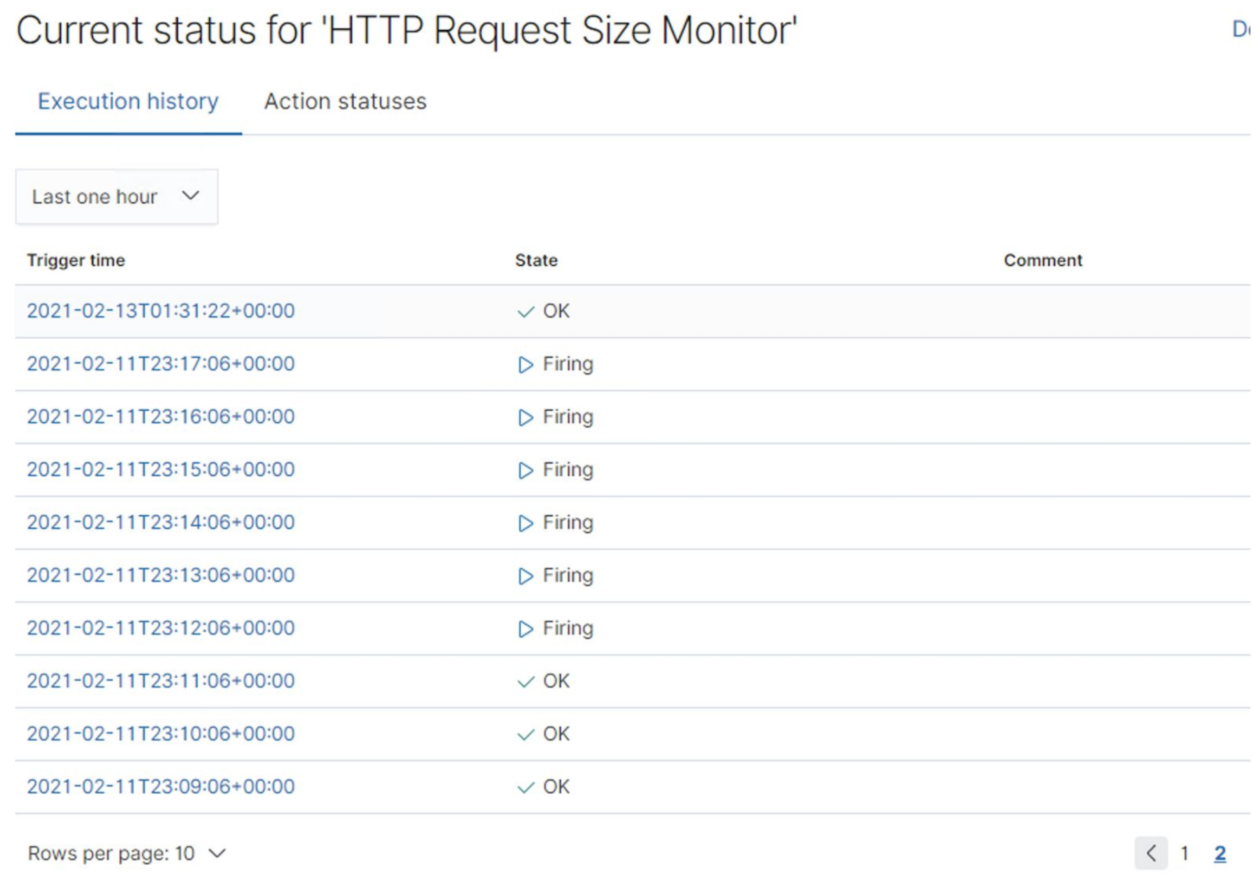
Current status for 'Excessive HTTP Errors'

Name	State
logging_1	✓ OK
logging_2	✓ OK

HTTP Request Size Monitor

HTTP Request Size Monitor is implemented as follows:

- Metric: **Packetbeat**
- Threshold: **Above 3500 for the last 1 minute**
- Vulnerability Mitigated: **HTTP Request Bytes**
- Reliability: **No it does not present false positives or false negatives, the monitor is behaving as it should.**

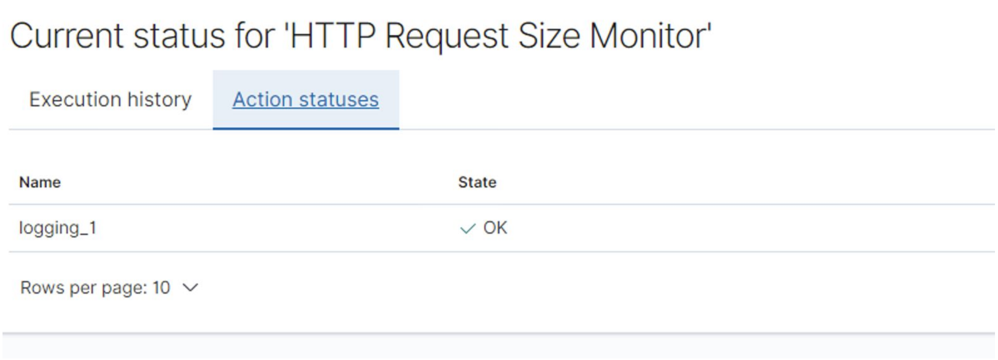


Current status for 'HTTP Request Size Monitor'

Trigger time	State	Comment
2021-02-13T01:31:22+00:00	✓ OK	
2021-02-11T23:17:06+00:00	▷ Firing	
2021-02-11T23:16:06+00:00	▷ Firing	
2021-02-11T23:15:06+00:00	▷ Firing	
2021-02-11T23:14:06+00:00	▷ Firing	
2021-02-11T23:13:06+00:00	▷ Firing	
2021-02-11T23:12:06+00:00	▷ Firing	
2021-02-11T23:11:06+00:00	✓ OK	
2021-02-11T23:10:06+00:00	✓ OK	
2021-02-11T23:09:06+00:00	✓ OK	

Rows per page: 10

< 1 2



Current status for 'HTTP Request Size Monitor'

Name	State
logging_1	✓ OK

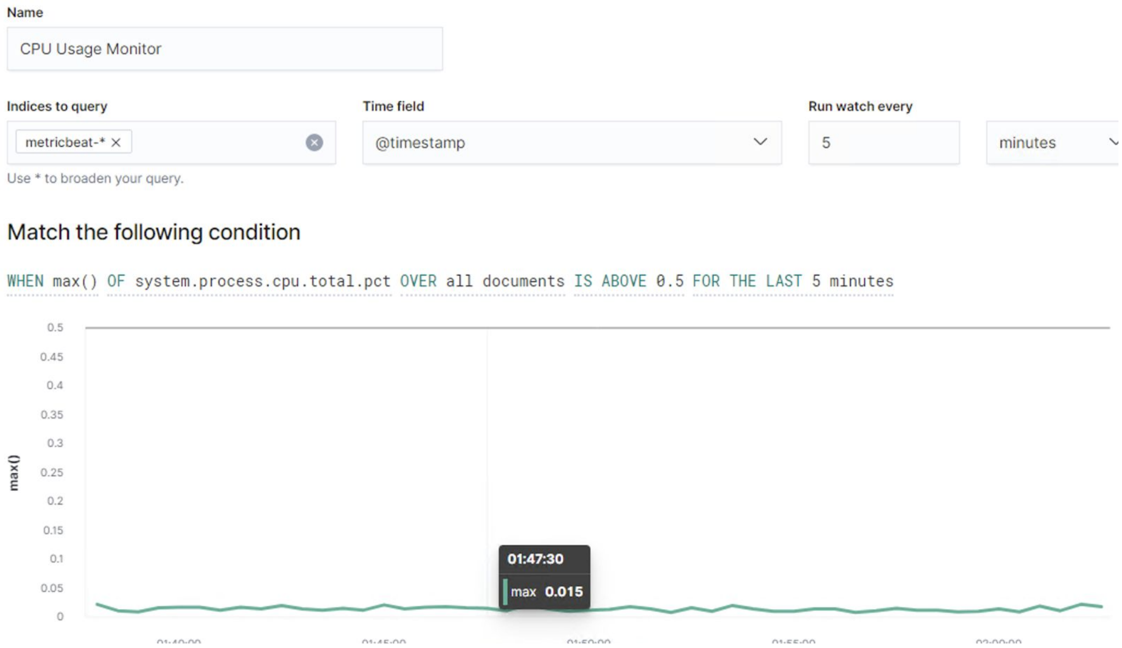
Rows per page: 10

CPU Usage Monitor

CPU Usage Monitor is implemented as follows:

- Metric: Metricbeat
- Threshold: Is above 0.5 for the last 5 minutes

- Vulnerability Mitigated: **Dos attacks Printers**
- Reliability: **This does not present any negative or positives because the threshold set too high the threshold needs to be lowered to around 2.5**



Name

CPU Usage Monitor

Indices to query

metricbeat-* ×

Use * to broaden your query.

Time field

@timestamp

Run watch every

5

minutes

Match the following condition

```
WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes
```

Suggestions for Going Further

The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats. In addition to watching for occurrences of such threats, the network should be hardened against them. The Blue Team suggests that IT implement the fixes below to protect the network:

HTTP Errors:

- Through backup of systems and database
- Run security patch updates
- Running the security patches and completing the backups will prevent the DoS attacks and from rendering the website unusable for users.

Request Size Monitor

- Patch: This patch will include using modules to harden the system
- Why It Works: This will allow the system to automatically update when new patches become available and it can target the HTTP requests based.

Vulnerability 3

- Patch: Update operating system, Update browser, and
- Why It Works: This will harden your print servers from leaking any information from the printers.