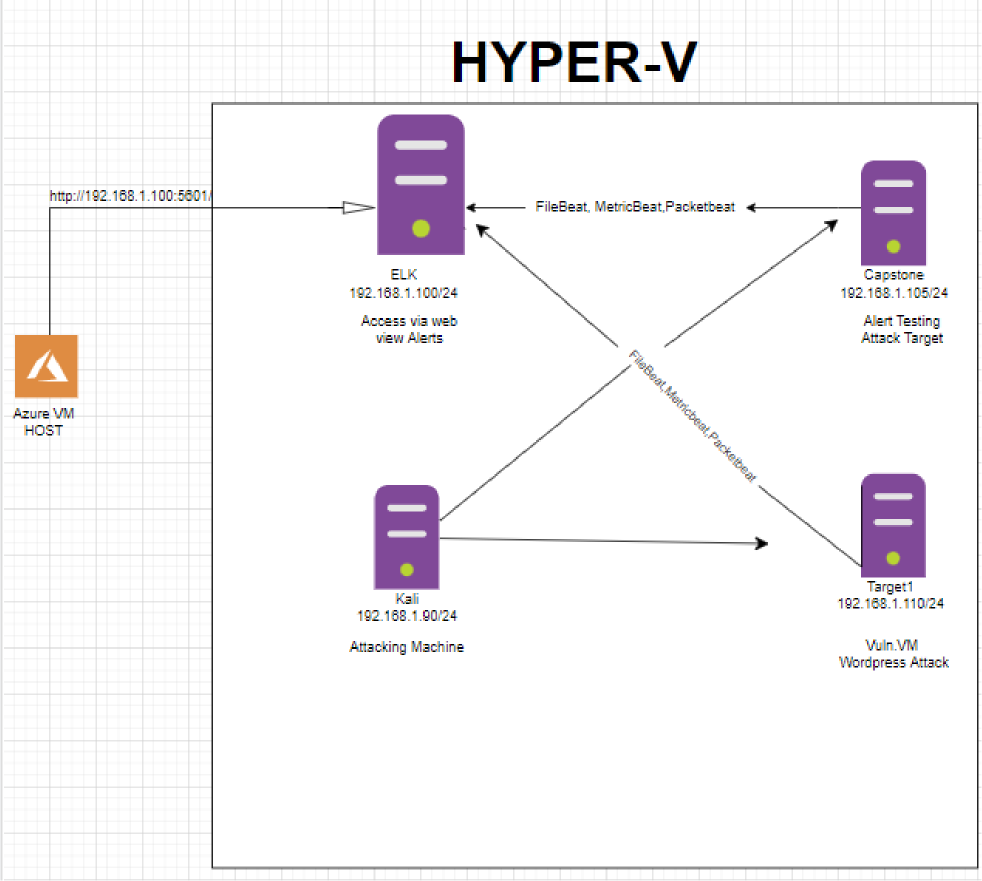
Blue Team: Summary of Operations

Table of Contents

Network Topology

The following machines were identified on the network:



- Name of VM 1 Kali
    - Operating System: Linux 5.4.0
    - Purpose: Kali is used as the attacking machine.
    - IP Address:192.168.1.90
- Name of VM 2 Target 1
    - Operating System: Linux 3.2
    - Purpose: Target 1 is used with WordPress as a vulnerable server.
    - IP Address:192.168.1.110
- Name of VM 3 ELK
    - Operating System: Linux
    - Purpose:It was used for gathering information from the victim machine with metricbeat,filebeat,and packetbeat
    - IP Address:192.168.1.100/24
- Name of VM 4 Capstone
    - Operating System: Linux
    - Purpose:Tests system for alerts
    - IP Address:192.168.1.105/24

Description of Targets

The target of this attack was: `Target 1` 192.168.1.110

Target 1 is an Apache web server and has SSH enabled, so ports 80 and 22 are possible ports of entry for attackers. As such, the following alerts have been implemented:

Monitoring the Targets

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

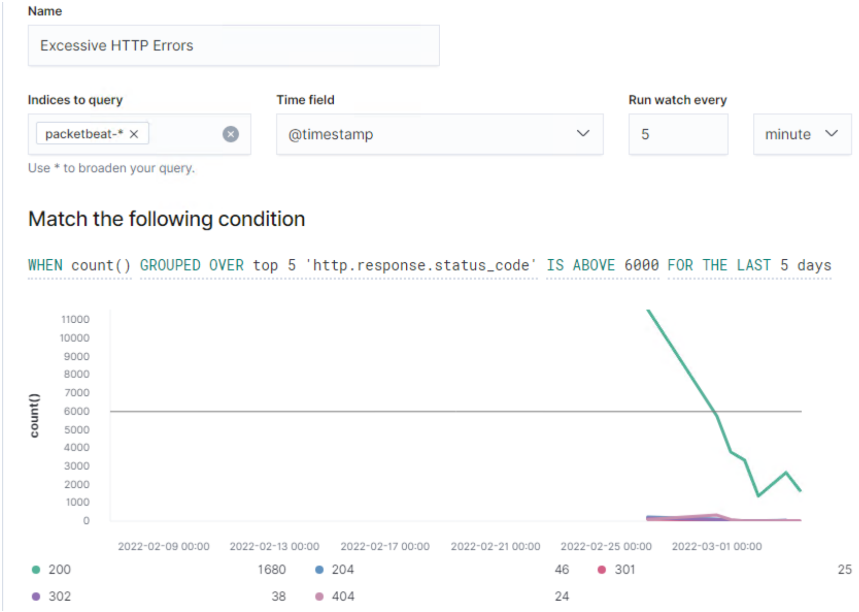
Name of Alert 1

Excessive HTTP Errors

Excessive HTTP Errors is implemented as follows:

- Metric: Packetbeat: http.response.status_code
- Threshold: grouped http response status codes above 6000 every 5 minutes

```
WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 6000
FOR THE LAST 5 days
```

- Vulnerability Mitigated: Port 22 needs to be either closed or disabled. Users should change their password to their accounts every 90 days.
- Reliability: Does this alert generate lots of false positives/false negatives? It’s High because of the high amount of Http errors. it does not create because its a reliable alert when you have a lot of error responses then most likely its managing traffic well.
- 

Name

Excessive HTTP Errors

Indices to query

packetbeat-* ×

Use * to broaden your query.

Time field

@timestamp

Run watch every

5 minute

Match the following condition

```
WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 6000 FOR THE LAST 5 days
```

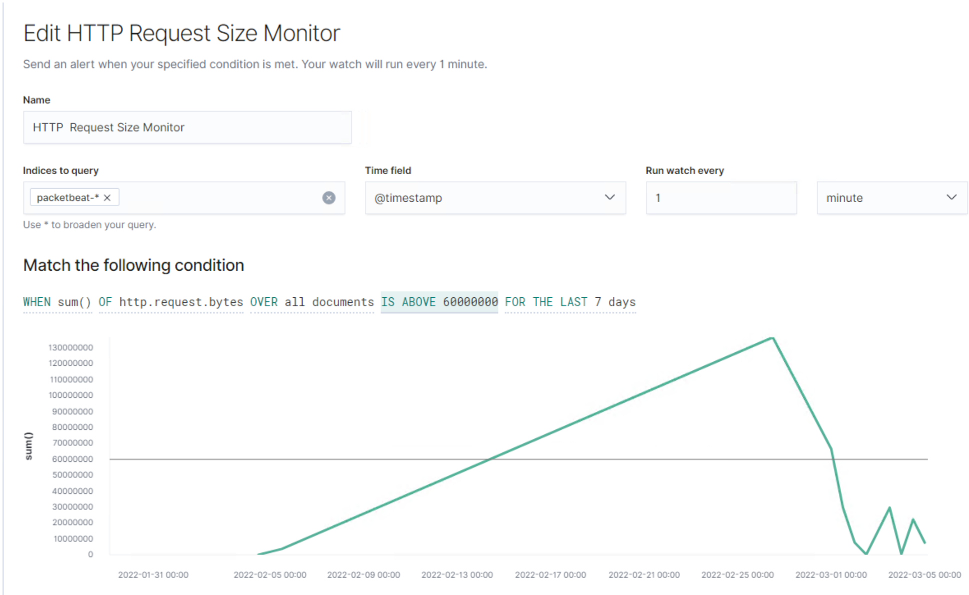
Name of Alert 2

HTTP Request Size Monitor implemented as follows:

- Metric: Packetbeat: http.request.bytes
- Threshold: The sum of the requested bytes is over 60000000 in 1 minute

```
WHEN sum() of http.request.bytes OVER all documents IS ABOVE 60,000,000
FOR THE LAST 7 days
```

- Vulnerability Mitigated: Generate an alert when bytes are over 60,000,000
- Reliability: TODO: Medium reliability because of the high amount of bytes.



Edit HTTP Request Size Monitor

Send an alert when your specified condition is met. Your watch will run every 1 minute.

Name: HTTP Request Size Monitor

Indices to query: packetbeat-*

Use * to broaden your query.

Time field: @timestamp

Run watch every: 1 minute

Match the following condition

```
WHEN sum() OF http.request.bytes OVER all documents IS ABOVE 60000000 FOR THE LAST 7 days
```

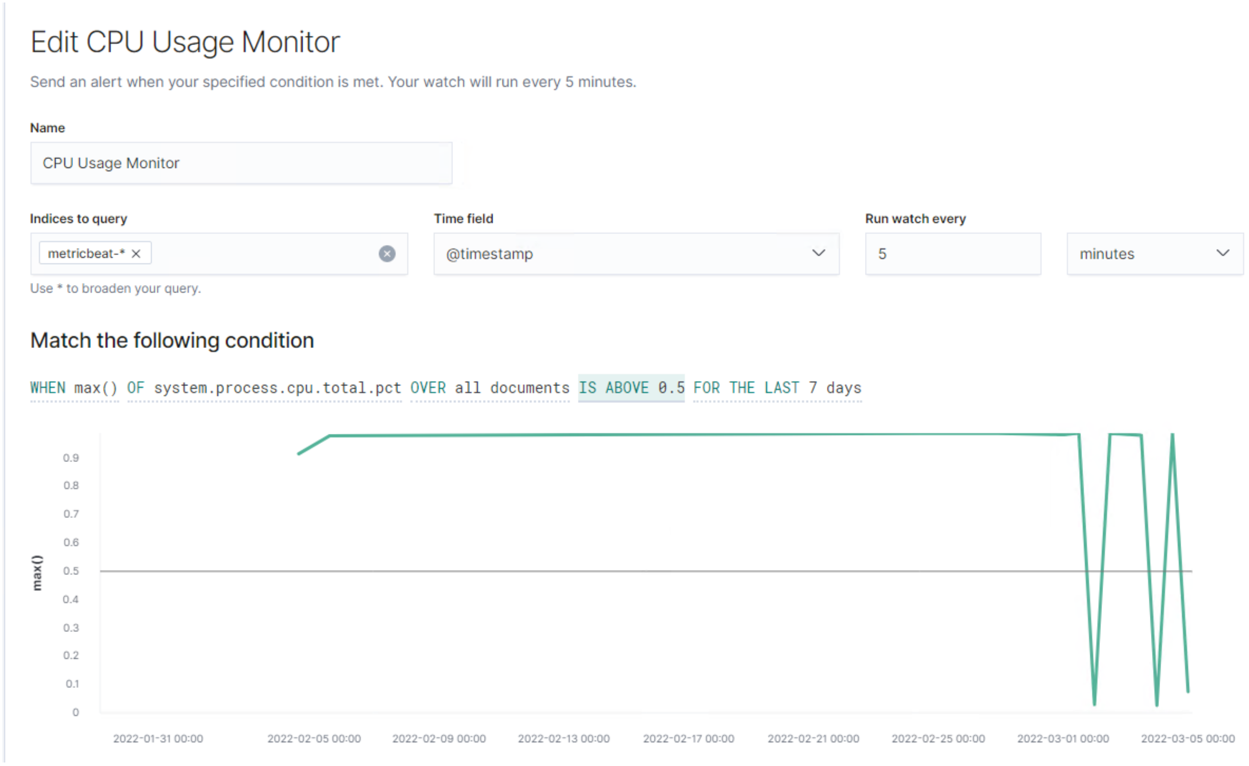
Name of Alert 3

CPU Usage Monitor is implemented as follows:

- Metric: Metricbeat: system.process.cpu.total.pct
- Threshold: The maximum cpu total percentage is over .5 in 7 days

```
WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE
0.5 FOR THE LAST 7 days
```

- Vulnerability Mitigated: Detects DOS
- Reliability: TODO: Does this alert generate lots of false positives/false negatives? Low because some legitimate software uses a lot of cpu.



Edit CPU Usage Monitor

Send an alert when your specified condition is met. Your watch will run every 5 minutes.

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 7 days
```