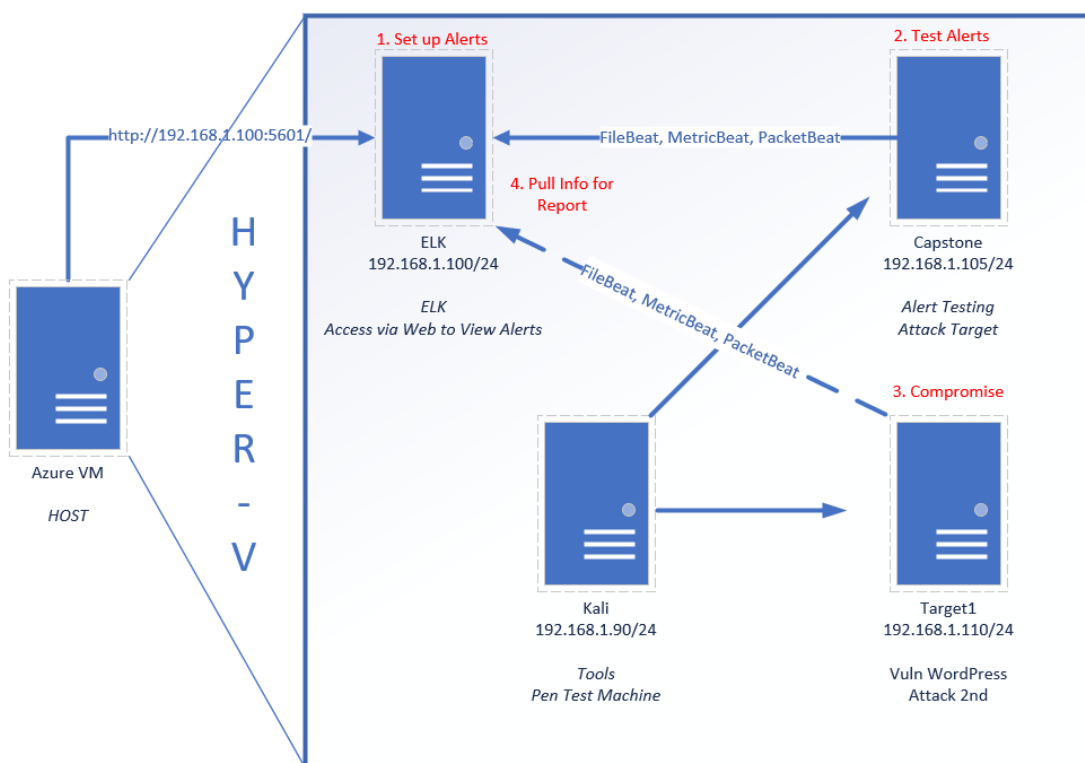


Defensive: Summary of Operations

Table of Contents

- Network Topology
- Description of Targets
- Monitoring the Targets
- Patterns of Traffic & Behavior
- Suggestions for Going Further

Network Topology



```
File Actions Edit View Help
root@Kali:~# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-02-26 10:43 PST
Nmap scan report for 192.168.1.110
Host is up (0.0024s 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 13.13 seconds
root@Kali:~#
```

The following machines were identified on the network:

- Name of VM 1: **Target 1**
 - **Operating System:** **Linux**
 - **Purpose:** **Word press server**
 - **IP Address:** **192.168.1.110**
- Name of VM 2
 - **Operating System:**
 - **Purpose:**
 - **IP Address:**
- Etc.

Description of Targets

The target of this attack was: Target 1 **ip. 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

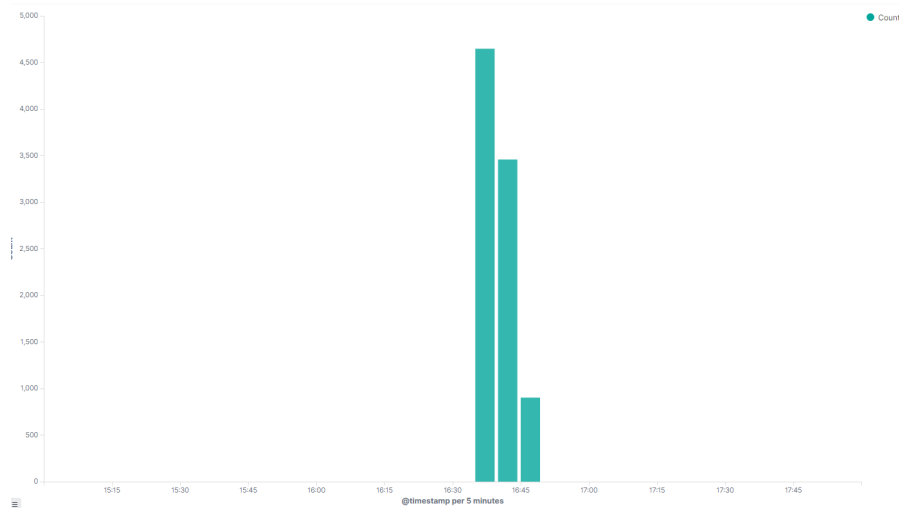
Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

Name of Alert 1

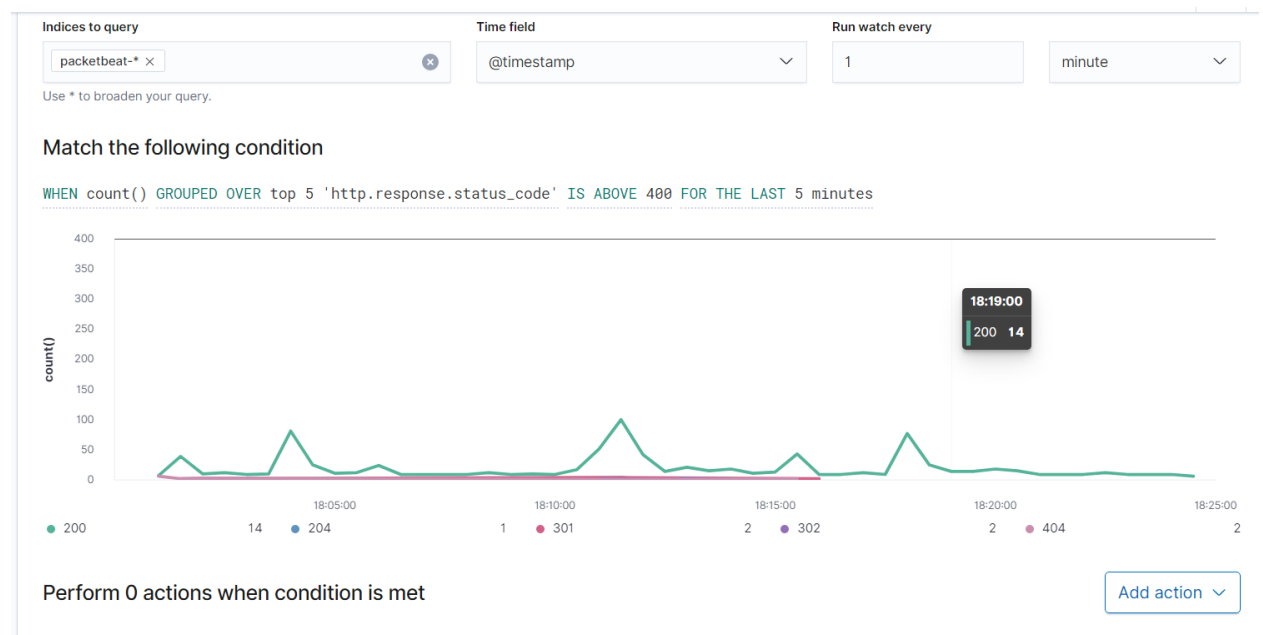
Excessive HTTP Errors is implemented as follows:

- **Metric:** `http.response.status_code`
- **Threshold:** `http://response.status_code > 400`
- **Vulnerability Mitigated:** Brute Force Attack
- **Reliability:** This alert does not generate false positives. I feel that it is a highly reliable alert for monitoring a brute force attack.

●



●



●

Name of Alert 2

HTTP Request Size Monitor is implemented as follows:

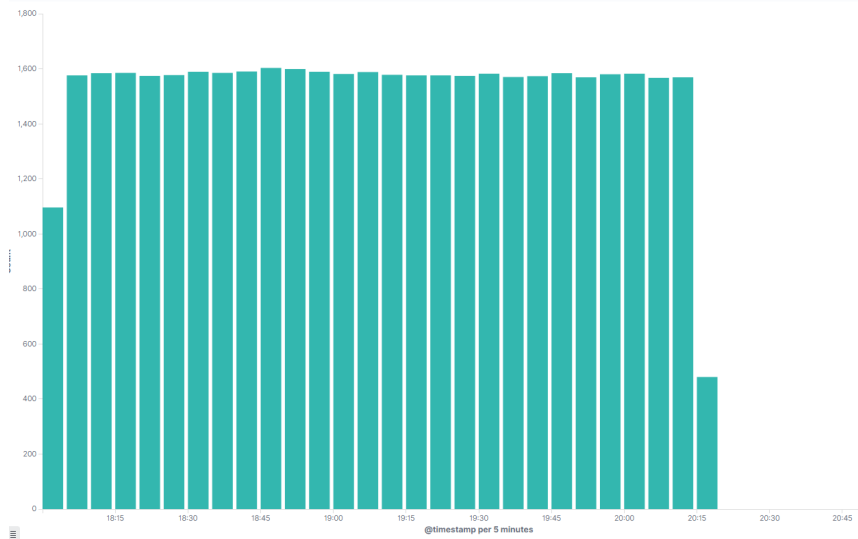
- **Metric:** `http.request.bytes`
- **Threshold:** 3500 hits in 1 min
- **Vulnerability Mitigated:** DDOS
- **Reliability:** No false positives. I feel the reliability is medium due to the number of hits over 3500 but not much higher than 5000.



Name of Alert 3:

CPU Usage Monitor is implemented as follows:

- **Metric:** `system.process.cpu.total.pct`
- **Threshold:** 0.5 usage every 5 minutes
- **Vulnerability Mitigated:** Malware
- **Reliability:** TODO: This alert will generate a lot of false positives. I would rate this alert low.



Indices to query: [+](#)

Time field: [v](#)

Run watch every: [minutes](#) [v](#)

Use * to broaden your query.

Match the following condition

WHEN `max()` OF `system.process.cpu.total.pct` OVER all documents IS ABOVE 0,5 FOR THE LAST 5 minutes

`max()`

A line chart showing the maximum value of `system.process.cpu.total.pct` over time. The y-axis is labeled 'max()' and ranges from 0 to 0.5. The x-axis shows time intervals from 18:10:00 to 18:30:00. The chart displays a green line representing the data points. The values are relatively stable, fluctuating between approximately 0.05 and 0.1.

Perform 0 actions when condition is met [Add action v](#)

[✓ Create alert](#) [Cancel](#) [Show request](#)