# Blue Team: Summary of Operations

Table of Contents

- Network Topology

- Description of Targets

- Monitoring the Targets

- Patterns of Traffic & Behavior

- Suggestions for Going Further

## Network Topology

\_TODO: Fill out the information below.\_

The following machines were identified on the network:

- Name of VM 1 (ELK)

- \*\*Operating System\*\*: Ubuntu

- \*\*Purpose\*\*: To set up alerts and monitor network traffic

- \*\*IP Address\*\*:192.168.1.100

- Name of VM 2 (Azure VM)

- \*\*Operating System\*\*: Windows 10 Pro

- \*\*Purpose\*\*:Jumpbox (first layer of defense)

- \*\*IP Address\*\*:192.168.84.49

- Name of VM 3 Target 1

- \*\*Operating System\*\*: Debian Linux

- \*\*Purpose\*\*: Target machine for defensive security

- \*\*IP Address\*\*:192.168.1.110

- Name of VM 4 Target 2

- \*\*Operating System\*\*: Debian Linux

- \*\*Purpose\*\*: Target machine for defensive security

- \*\*IP Address\*\*:192.168.1.115

- Name of VM 5 Capstone

- \*\*Operating System\*\*: Ubuntu 18.04.4 LTS

- \*\*Purpose\*\*: Alert testing and attack target

- \*\*IP Address\*\*:192.168.1.105

- Name of VM 5 Kali Machine

- \*\*Operating System\*\*: Linux 5.4.0-Kali 3-AMD 64

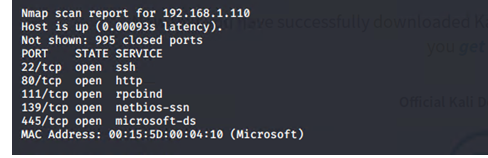
- \*\*Purpose\*\*: For offensive security

- \*\*IP Address\*\*:192.168.1.90

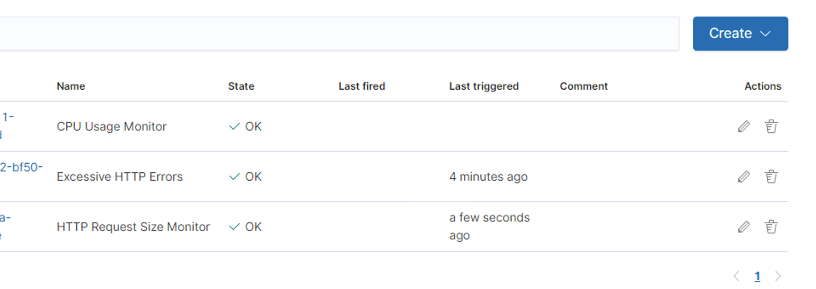
## Description of Targets

Answer the questions below.\_

The target of this attack was: `Target 1` (TODO: IP Address: 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-



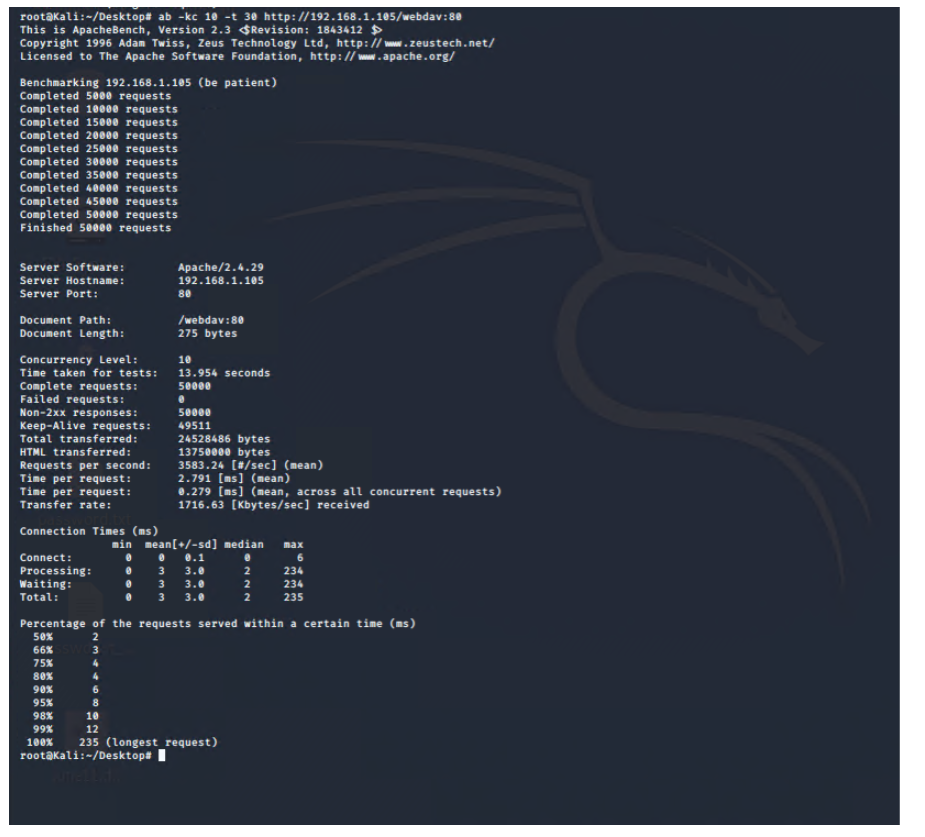
The three alerts created are:

1. CPU Usage
2. HTTP Request Size Monitor
3. Excessive HTTP Request errors

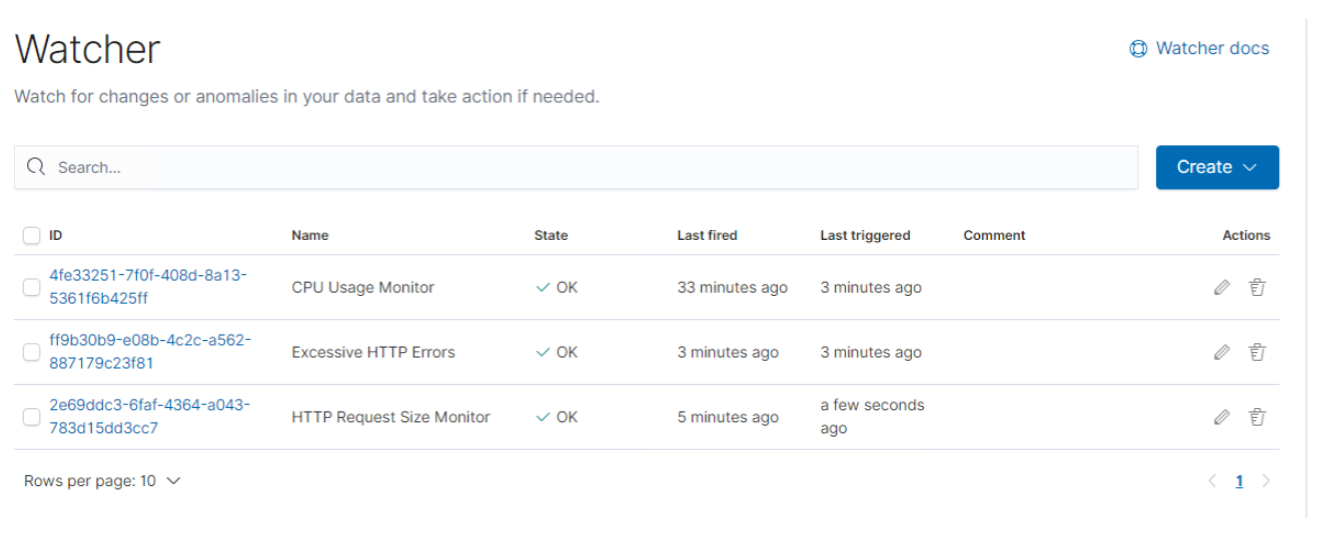
## Monitoring the Targets

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

Note- To test that the alerts worked, we used apache benchmark script to get the alerts triggered. See screen shot below.



When we triggered the alert, the screenshot below showed that the alerts worked, see screen shot below.



#### Alerts

1) CPU Usage Monitor is implemented as follows:

- \*\*Metric\*\*: Percentage of CPU usage over time

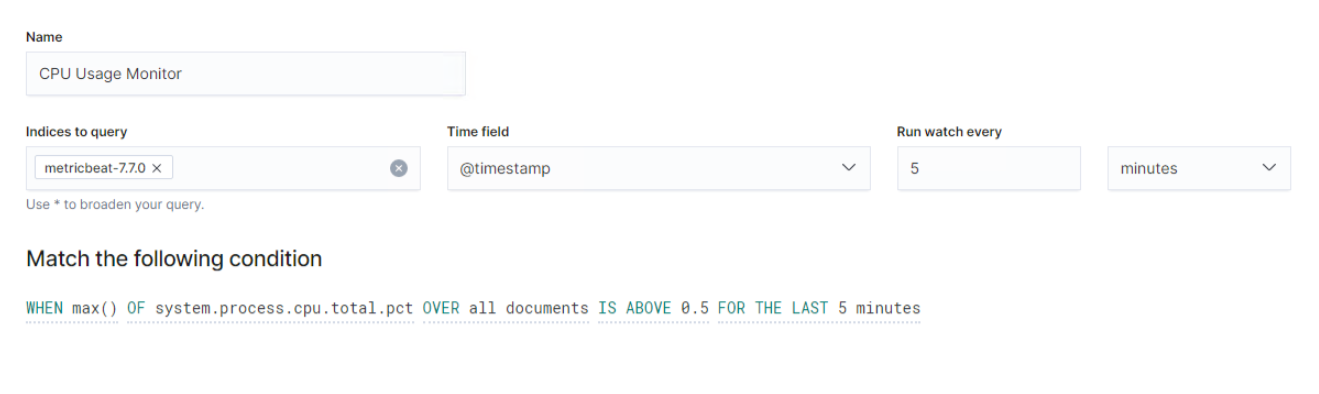
- \*\*Threshold\*\*: 50 percent (0.5)

- \*\*Vulnerability Mitigated\*\*: CPU usage due to DDOS attacks

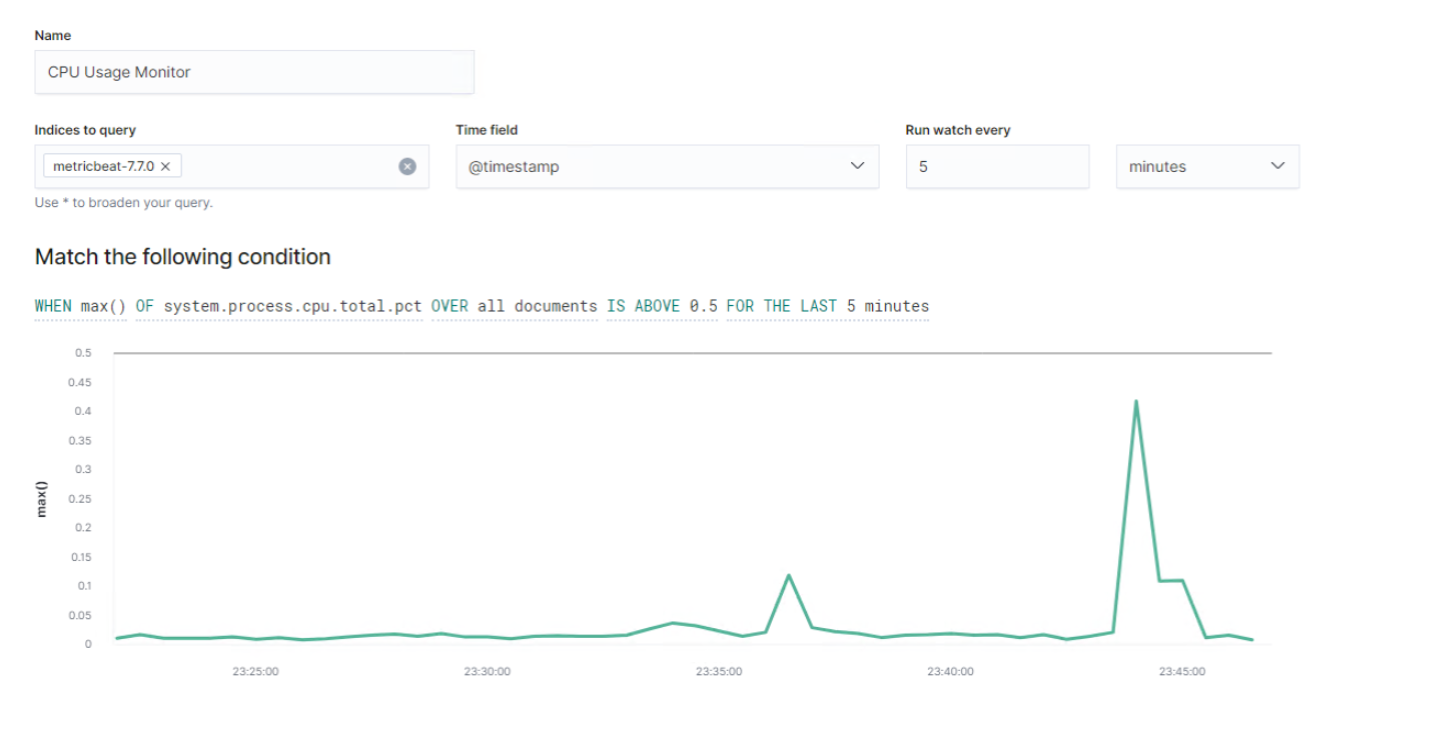
- \*\*Reliability\*\*: This is a good indicator when combined with other alerts

Does this alert generate lots of false positives/false negatives? I would say this is low

### Alert created



### Alert Triggered



2) HTTP Request Size Monitor implemented as follows:

- \*\*Metric\*\*: HTTP Request bytes

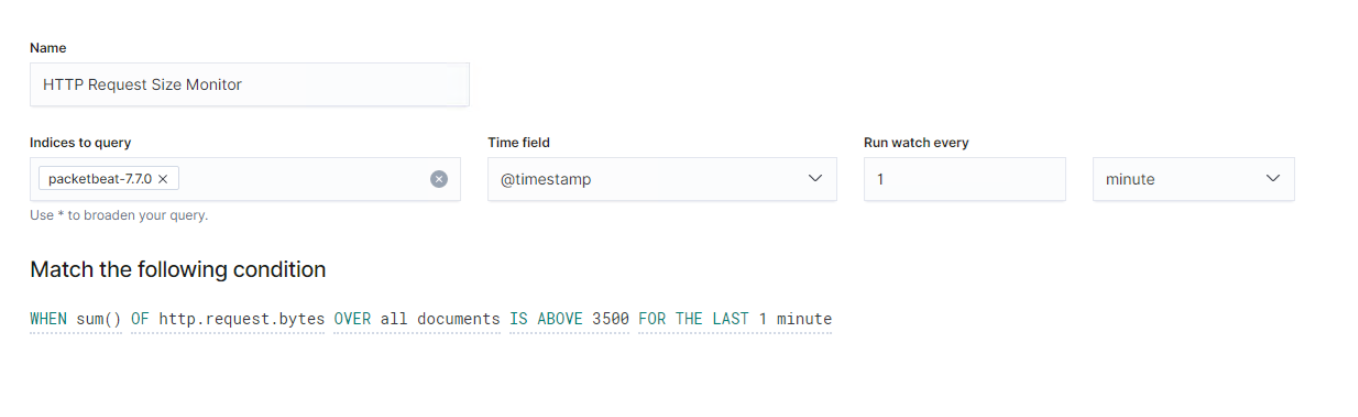
- \*\*Threshold\*\*: anything over > 3500 bytes

- \*\*Vulnerability Mitigated\*\*: Large payloads within timeframe (1 minute)

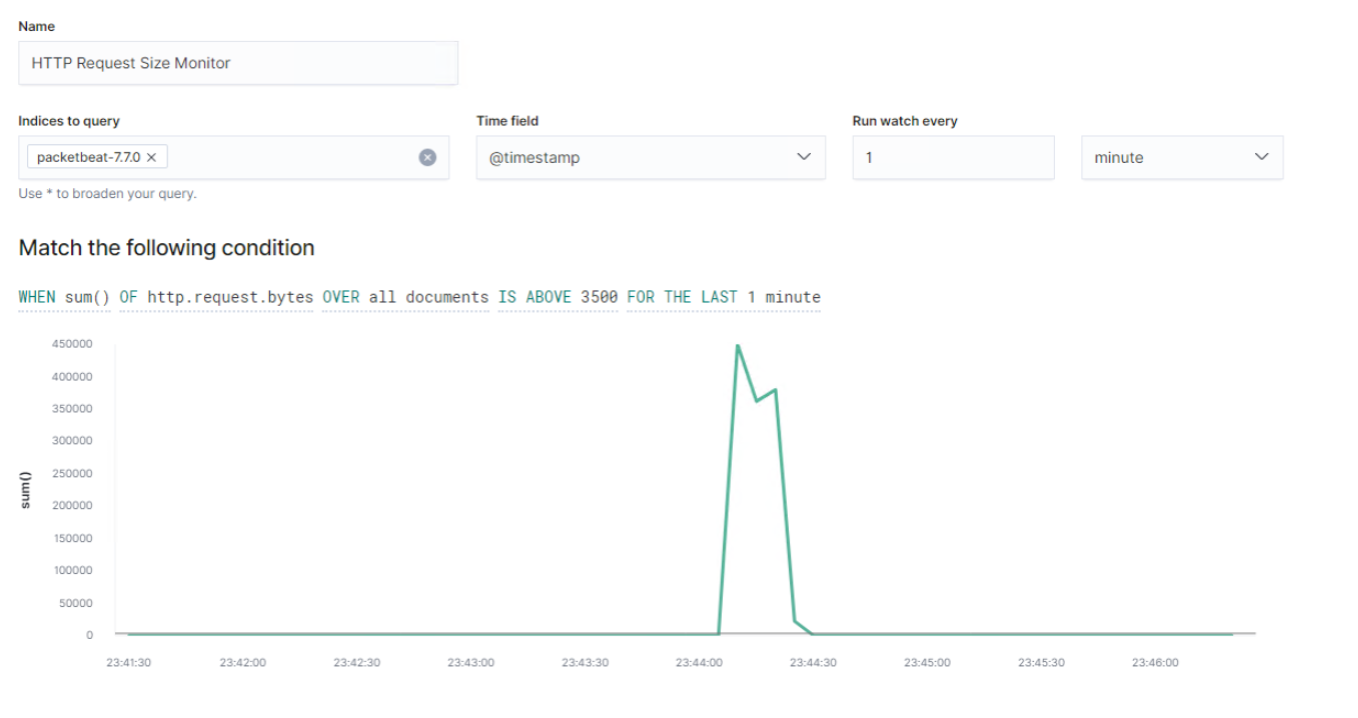
- \*\*Reliability\*\*: Possible as we do not get a lot of data to indicate the number of users trying to access

Does this alert generate lots of false positives/false negatives? There is a Low chance that it generates some false positives/false negatives

#### Alert Created



#### Alert Triggered



3) Excessive HTTP Request Errors implemented as follows:

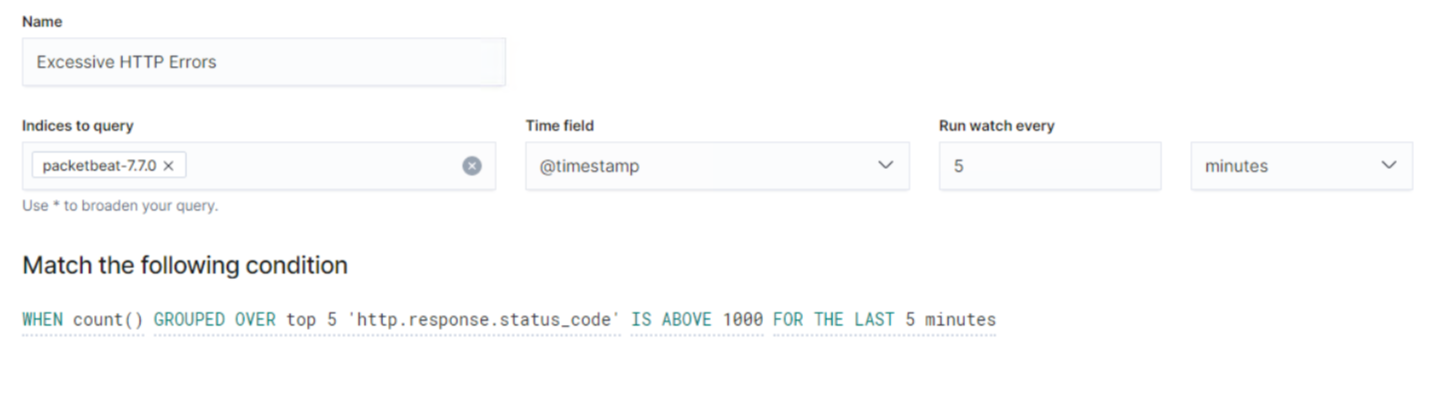
- \*\*Metric\*\*: Incoming error codes over time (400 series)

- \*\*Threshold\*\*: value > 400

- \*\*Vulnerability Mitigated\*\*: unauthorized login attempts

- \*\*Reliability\*\*: Does this alert generate lots of false positives/false negatives? Not the most reliable source, there is a medium chance that it generates some false positives/false negatives

#### Alert Created



#### Alert Triggered

