# Blue Team: Summary of Operations

## Table of Contents

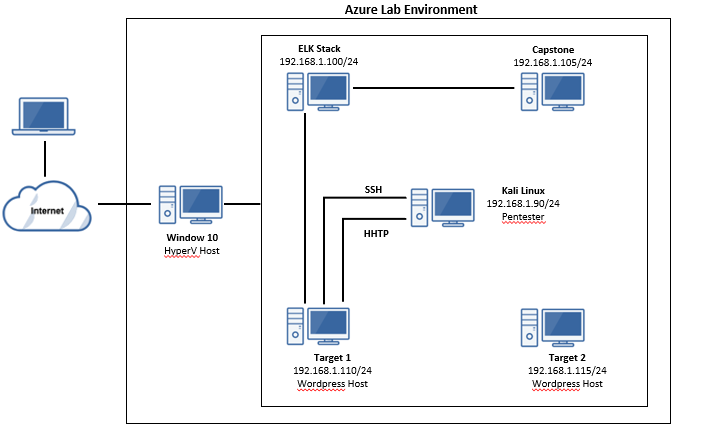
* Network Topology
* Description of Targets
* Monitoring the Targets
* Patterns of Traffic and Behavior
* Suggestions for Going Further

Prepared by Eric Loi

Team Members: Md Iftakher Hossain, Lucas Domanski, Eric Loi

May 6th, 2021

## Network Topology



The following machines were identified on the network:

**Kali Linux**

* Operating System: Debian Kali 5.4.0
* Purpose: Pentester
* IP Address: 192.168.1.90

**ELK Stack**

* Operating System: Ubuntu 18.04
* Purpose: The ELK (Elasticsearch and Kibana) Stack
* IP Address: 192.168.1.100

**Capstone**

* Operating System: Ubuntu 18.04
* Purpose: The Vulnerable Web Server
* IP Address: 192.168.1.105

**Target 1**

* Operating System: Debian GNU/Linux 8
* Purpose: Target 1 Machine (Wordpress host)
* IP Address: 192.168.1.110

**Target 2**

* Operating System: Debian GNU/Linux 8
* Purpose: Target 2 Machine
* IP Address: 192.168.1.115

## Description of Targets

Fill in the following:

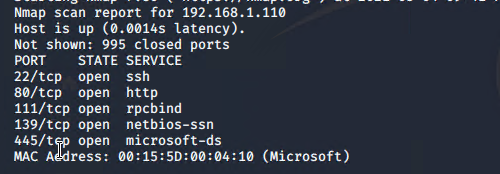
* Two VMs on the network were vulnerable to attack: Target 1 [192.168.1.110] and Target 2 [192.168.1.115].
* Each VM functions has an Apache web server and has SSH enabled, ports 80 and 22 are possible ports of entry for attackers.

Note: Pentest was only carried out for Target 1.

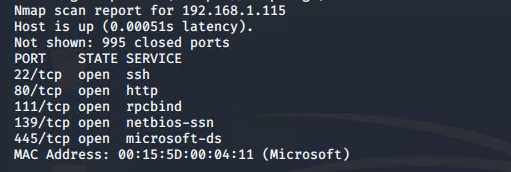
## Monitoring the Targets

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

* **Target 1**



* **Target 2** (nmap was performed for Target 2)

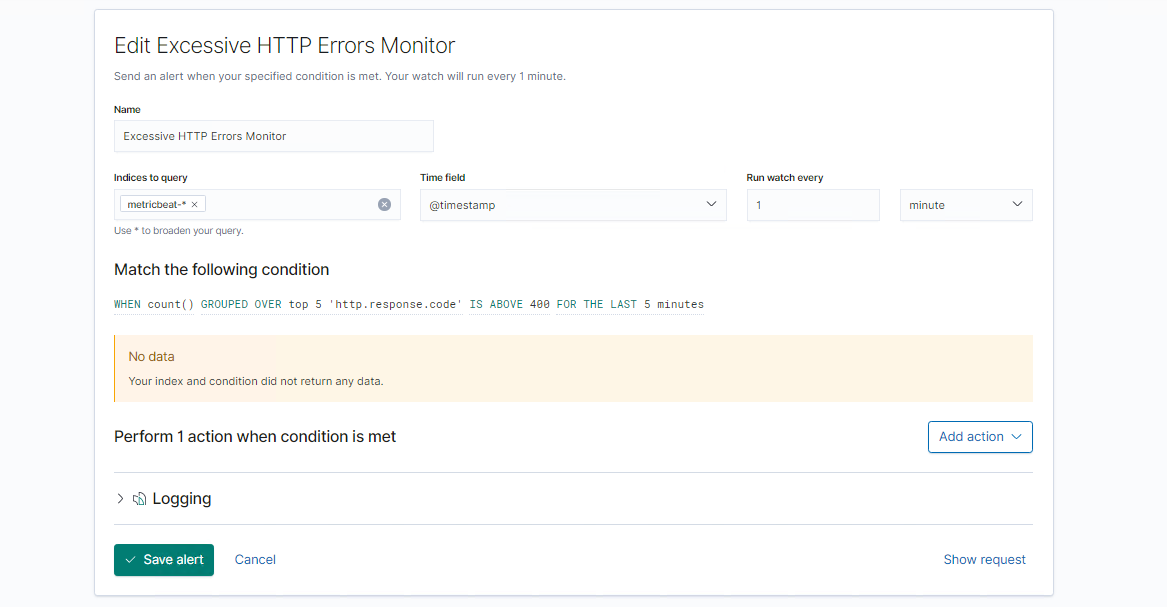


Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

**Excessive HTTP Errors Alert**

Excessive HTTP Errors Alert is implemented as follows:

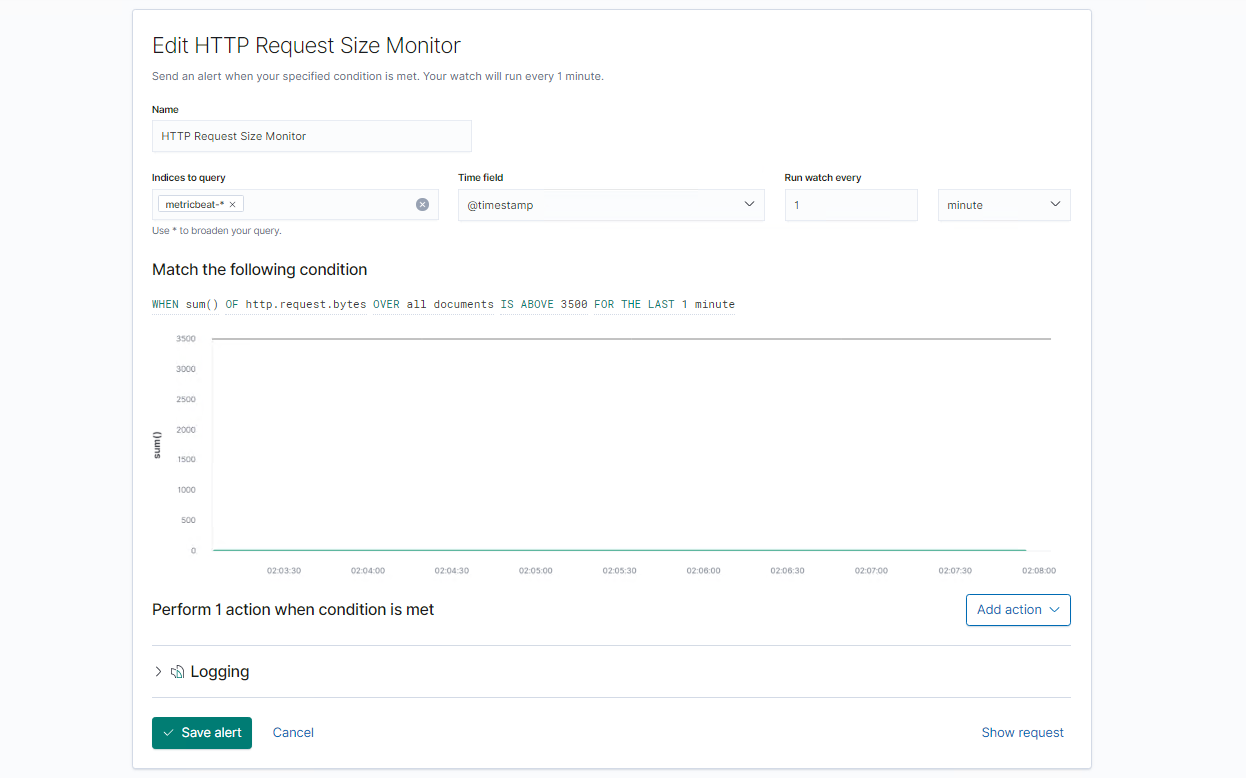
* **Metric:** WHEN count() GROUPED OVER top 5 'http.response.status\_code'
* **Threshold:** IS ABOVE 400
* **Vulnerability Mitigated:** Enumeration/Brute Force
* **Reliability:** The alert is highly reliable. Measuring by error codes 400 and above will filter out any normal or successful responses. 400+ codes are client and server errors which are of more concerning. Especially when taking into account these error codes occurs at a high rate.



**HTTP Request Size Monitor Alert**

HTTP Request Size Monitor is implemented as follows:

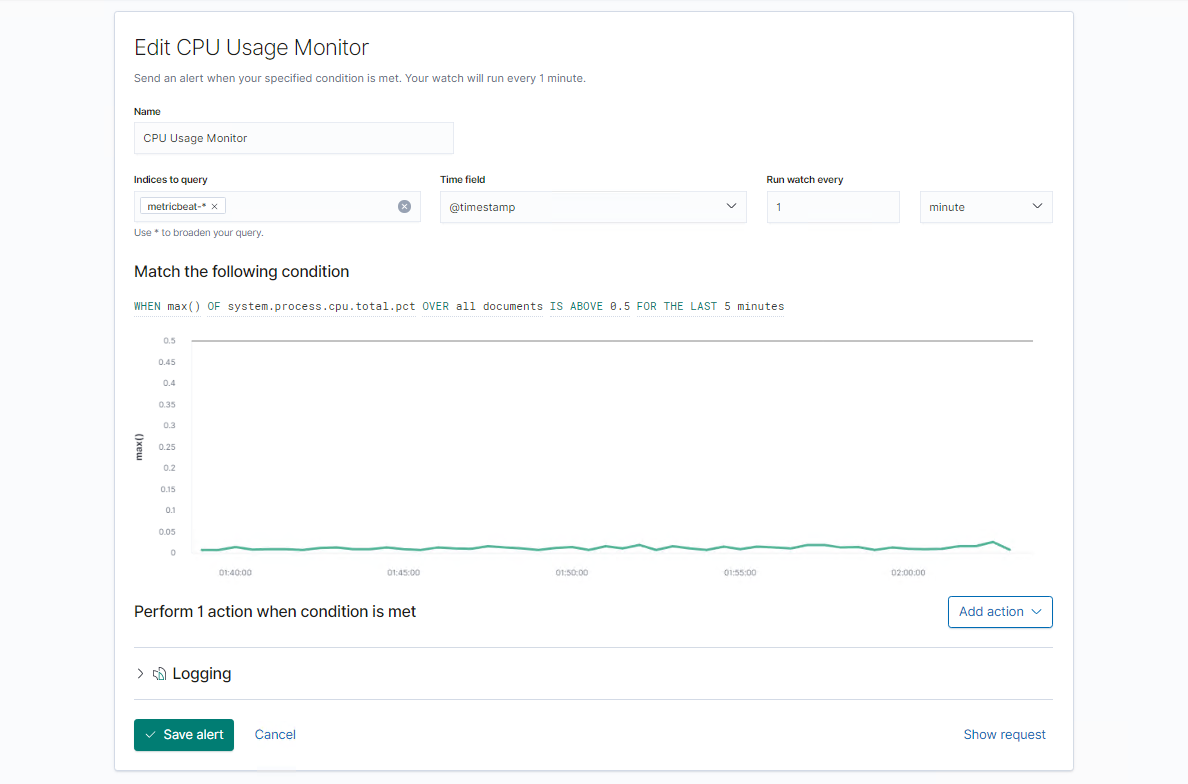
* **Metric:** WHEN sum() of http.request.bytes OVER all documents
* **Threshold:** ABOVE 3500
* **Vulnerability Mitigated**: Code injection in HTTP requests (XSS and CRLF) or DDOS
* **Reliability:** Alert could create false positives. It comes in at a medium reliability. There is a possibility for large volume of non-malicious HTTP requests or legitimate HTTP traffic.



**CPU Usage Monitor Alert**

CPU Usage Monitor Alert is implemented as follows:

* **Metric:** WHEN max() OF system.process.cpu.total.pct OVER all documents
* **Threshold:** ABOVE 0.5
* **Vulnerability Mitigated**: Malicious program taking up CPU resources
* **Reliability:** The alert is highly reliable. It is used for monitoring for a malicious program that is consuming CPU resources. For operation purpose, this alert can also be used for CPU usage monitoring.

****

## Suggestions for Going Further

**Mitigation for Vulnerability 1 with Excessive HTTP Errors Alert**

* Patch: WordPress Hardening
  + Implement regular updates to WordPress
    - WordPress Core
    - PHP version
    - Plugins
  + Install security plugin(s)
    - Ex. Wordfence (adds security functionality)
  + Disable unused WordPress features and settings like:
    - WordPress XML-RPC (on by default)
    - WordPress REST API (on by default)

Block requests to /?author= by configuring web server settings

* + Remove WordPress logins from being publicly accessible specifically:
    - /wp-admin
    - /wp-login.php
* Why It Works:
  + Keeping updated patches to WordPress, the PHP version and plugins is the easiest way to mitigate exploits/vulnerabilities.
  + Wordfence can also provide the following protection:
    - Malware scans
    - Firewall
    - IP options (to monitor/block suspicious traffic)
  + REST API is used by WPScan to enumerate users
    - Disabling REST API will help mitigate WPScan or enumeration.
  + XML-RPC uses HTTP as the method of data transport, disabling it to eliminate the vulnerability.
  + WordPress links (permalinks) can include authors (users)
    - Blocking “permalinks “request to view the all authors (users) helps to mitigate against user enumeration attacks.
  + Removal of public access to WordPress login helps reduce the attack surface.

**Mitigation of Vulnerability 2 with HTTP Request Size Monitor Alert**

* Code Injection/DDOS Hardening
  1. Input Validation and Sanitisation
  2. Applies Appropriate / Least User Privileges
  3. Deploys a Web Application Firewall (WAF)
  4. Using Error Messages Handling
* Why It Works:
  1. The most effective mitigation mechanism is to assume all user inputs are potentially malicious and perform data validation and sanitisation for all user-submitted input content before sending the queries to the server for execution. Input validation can help protect against malicious data attempts to send to the server via the website or application in/across a HTTP request.
  2. Applying minimum privileges can minimise impact to the server if the server does get compromised.
  3. WAF applies a set of rules to the communication within the HTTP/HTTPS/SOAP/XML-RPC/Web Service layers and help to filter out malicious data and requests.
  4. By not exposing detailed error messages, attackers cannot easily learn the system architecture from the error messages.

**Vulnerability 3: CPU Usage Monitor Alert**

* Virus or Malware hardening
  1. Using antivirus software and applying the latest virus signatures prevent potential virus infection.
  2. Scheduled scanning of server files, RAM as regular maintenance.
  3. Data backup to ensure date recovery
* Why It Works:
  1. Antiviruses specialize in removal, detection and overall prevention of malicious threats against computers.
  2. Schedule scanning can remove known malicious viruses.
  3. Backup can provide data recovered if a virus infection caused data corruption. In case of a ransom where, data restoring is the best mitigation.