# **Blue Team: Summary of Operations**

## **Table of Contents**

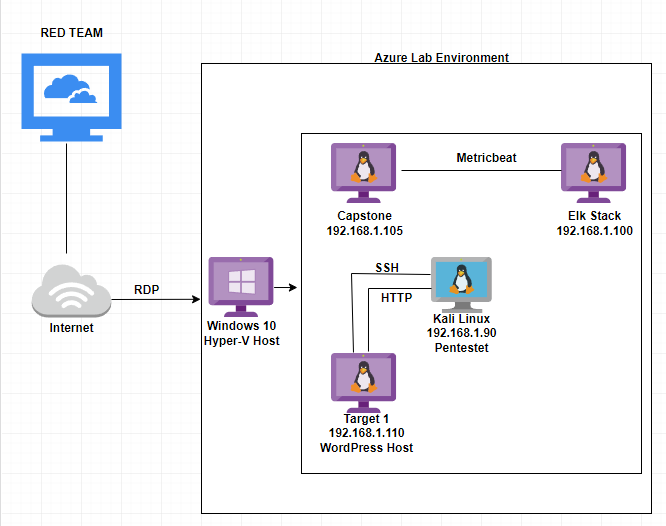
* Network Topology
* Description of Targets
* Monitoring the Targets
* Patterns of Traffic & Behavior
* Suggestions for Going Further

### **Network Topology**

The following machines were identified on the network:

* **Kali**
  + **Operating System**:Linux
  + **Purpose**: Penetration Tester
  + **IP Address**:192.168.1.90
* **Elk**
  + **Operating System**:Ubuntu
  + **Purpose**:ELK Stack ( Kibana & Elasticsearch )
  + **IP Address**:192.168.1.100
* **Capstone**
  + **Operating System**:Ubuntu
  + **Purpose**:Vulnerable Machine
  + **IP Address**:192.168.1.105
* **Target 1**
  + **Operating System**:Linux
  + **Purpose**:Wordpress Host
  + **IP Address**:192.168.1.110

**Network Diagram**



### **Description of Targets**

The target of this attack was: Target 1 (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:

### **Monitoring the Targets**

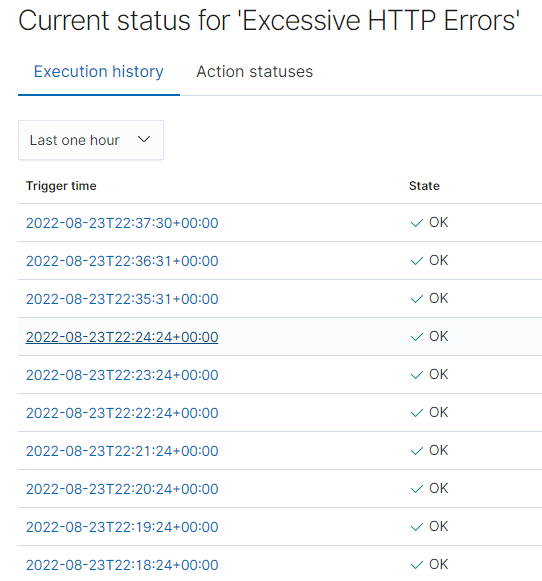
Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

#### **Excessive HTTP Errors**

WHEN count() GROUPED OVER top 5 'http.response.status\_code' IS ABOVE 400 FOR THE LAST 5 minutes

Alert 1 is implemented as follows:

* **Metric**:WHEN count() GROUPED OVER top 5 'http.response.status\_code'
* **Threshold**: IS ABOVE 400
* **Vulnerability Mitigated**: Enumeration and Brute Force Attacks
* **Reliability**: The Excessive HTTP error alert is highly reliable due to the fact that it can measure the amount of error codes of 400+ filtering out normal and successful responses.



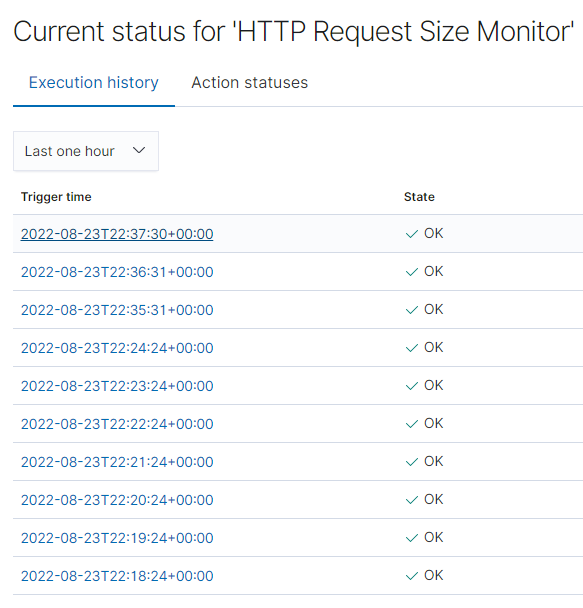
#### 

#### 

#### **HTTP Request Size Monitor**

Alert 2 is implemented as follows:WHEN sum() of http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute

* **Metric**: WHEN sum() of http.request.bytes OVER all document
* **Threshold**: IS ABOVE 3500
* **Vulnerability Mitigated**: DDOS and Code Injection via HTTP requests
* **Reliability**: The reliability for this alert is moderate (medium reliability). There is a possible margin open to larger non-malicious HTTP traffic altering it, or HTTP requests.



#### **CPU Usage Monitor**

Alert 3 is implemented as follows:WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes

* **Metric**: WHEN max() OF system.process.cpu.total.pct OVER all documents
* **Threshold**: IS ABOVE 0.5
* **Vulnerability Mitigated**: Tracking malware, viruses, and malicious software running taking up host resources.
* **Reliability**: The alert for CPU usage is a highly reliable alert. It can show how to optimize CPU usage even aside from tracking whether that is coming from a malicious source or not.

