# **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:

* Target 1
  + **Operating System**:Debian GNU/Linux 8
  + **Purpose**: Exposes a vulnerable WordPress server. Sends logs to ELK
  + **IP Address**: 192.168.1.110
* Target 2
  + **Operating System:** Debian GNU/Linux 8
  + **Purpose:** a more difficult WordPress target
  + **IP Address:** 192.168.1.115
* Kali
  + **Operating System**: Kali GNU/Linux Rolling
  + **Purpose**:used to attack other machines
  + **IP Address**: 192.168.1.90
* Capstone
  + **Operating System:** Ubuntu 18.04.1 LTS server1
  + **Purpose:** used to test alerts. Filebeat and Metricbeat will forward logs to the ELK machine
  + **IP Address:** 192.168.1.105
* ELK
  + **Operating System:** Linux- Ubuntu Version 18.04.4 LTS (Bionic Beaver)
  + **Purpose:** holds the Kibana dashboards
  + **IP Address:** 192.168.1.100

### **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 points 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:

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

HTTP Request Size Monitor is implemented as follows:

* **Metric**: Total sum of bytes of http requests in a given time.
* **Threshold**: Above 3500 bytes in the last minute.
* **Vulnerability Mitigated**: HTTP request smuggling
* **Reliability**: This setting generates a lot of false positives. The threshold was met time and time again without even interacting with the target. We would rate as low, or even *very* low reliability.

#### **Excessive HTTP Errors**

Excessive HTTP Errors implemented as follows:

* **Metric**: Number of HTTP status codes over 400
* **Threshold**: Top 5 status codes are above 400 in the last five minutes.
* **Vulnerability Mitigated**: Brute force attacks.
* **Reliability**: Could potentially be very reliable, but not if the attacker knows your credentials or if you use default credentials. Very reliable in regards to the wpscan, as the user enumeration only works by reading login error messages.

#### **CPU Usage Monitor**

CPU Usage Monitor is implemented as follows:

* **Metric**: Percentage of total CPU Usage.
* **Threshold**: More than 0.5% of CPU usage for the last 5 minutes.
* **Vulnerability Mitigated**: Resource Exhaustion
* **Reliability**: This is generating a lot of false positives. The threshold appears to be set too low for it to be truly reliable. We would rate this at low reliability.

### **Suggestions for Going Further (Optional)**

The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats, identified by the alerts above. In addition to watching for occurrences of such threats, the network should be hardened against them. The Blue Team suggests that IT implement the fixes below to protect the network:

* Vulnerability 1 - Improper Error Handling
  + **Patch**: Sanitize error messages. A specific policy for how to handle errors should be documented, including the types of errors to be handled and for each, what information is going to be reported back to the user, and what information is going to be logged. All developers need to understand the policy and ensure that their code follows it.
  + **Why It Works**: In the implementation, ensure that the site is built to gracefully handle all possible errors. When errors occur, the site should respond with a specifically designed result that is helpful to the user without revealing unnecessary internal details.
* Vulnerability 2 - Default/Weak Credentials
  + **Patch**: Install some sort of PAM - apt-get install libpam-cracklib
  + **Why It Works**: A pluggable authentication module (PAM) allows you to set password complexity requirements on a machine: root/toor and michael/michael.
* Vulnerability 3 - Resource Exhaustion
  + **Patch**: Install CPULimit or something similar.
  + **Why It Works**: Allows the limitation of specific processes or process groups