**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**:
    - Debian Kali 5.4.0
  + **Purpose**:
    - Penetration testing
  + **IP Address**:
    - **192**.168.1.90
* ELK
  + **Operating System**:
    - Ubuntu 18.04
  + **Purpose**:
    - Elasticsearch and Kibana Stack – system logging
  + **IP Address**:
    - 192.168.1.100
* Capstone
  + **Operating System**:
    - Ubuntu 18.04
  + **Purpose**:
    - Web Server
  + **IP Address**:
    - 192.168.1.105
* Target 1
  + **Operating System**:
    - Debian Linux 8
  + **Purpose**:
    - WordPress Host
  + **IP Address**:
    - 192.168.1.110

**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:

* CPU Usage Monitor
* Excessive HTTP Errors
* HTTP Request Size Monitor

**Monitoring the Targets**

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

**CPU Usage Monitor**

The usage monitor is implemented as follows:

Text

Description automatically generated

* **Metric**: CPU Usage percentage
* **Threshold**: The percentage goes above 0.5% over a 5 minute period
* **Vulnerability Mitigated**: TODO
* **Reliability**: Highly dependent on any services implemented on the server. The threshold may need to be adjusted based on how resource intensive the services are.

**Excessive HTTP Errors**

The Error alert is implemented as follows:

Graphical user interface, text, application, email

Description automatically generated

* **Metric**: The number of http response codes over 400
* **Threshold**: An error count over 5 over a 5-minute period
* **Vulnerability Mitigated**: The use of brute force attacks and attempted site enumeration via DNS navigation
* **Reliability**: This is a reliable method for mitigation as frequent error codes are unlikely to occur if the system is functioning properly. However, if the servers go down for whatever reason, this monitor will send out false positives though this could also indicate DoS attacks if the system goes down and recovers under the radar.

**HTTP Requst Size Monitor**

The monitor is implemented as follows:

Graphical user interface, text, application, email

Description automatically generated

* **Metric**: The request size of HTTP requests in bytes
* **Threshold**: Over 3500 bytes per request over the span of a minute
* **Vulnerability Mitigated**: Malicious software uploading
* **Reliability**: Very reliable depending on typical server traffic. It will pick up false positives if users upload otherwise innocuous files to the server (e.g. a site admin uploads a picture file to the website), however if this traffic is infrequent then the threshold errors are acceptable