**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:

* **Kalin Linux**
  + Purpose: An attacker virtual machine used to compromise other machines.
  + Operating System: Linux
  + IP Address: 192.168.1.90
* **Capstone**
  + Purpose: A target (victim) virtual machine that is used to test alerts and forwards the logs to the ELK machine via Filebeat and Metricbeat.
  + Operating System: Linux
  + IP Address: 192.168.1.105
* **ELK**
  + Purpose: ELK Stack that contains the Kibana dashboards.
  + Operating System: Linux
  + IP Address: 192.168.1.100
* **Target 1** 
  + Purpose: A target (victim) virtual machine that reveals a vulnerable WordPress server and also forwards the logs to the ELK machine.
  + Operating System: Linux
  + IP Address: 192.168.1.110
* **Target 2**
  + Purpose: A target (victim) virtual machine that reveals a vulnerable WordPress server and also forwards the logs to the ELK machine.
  + Operating System: Linux
  + IP Address: 192.168.1.115

**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 ports of entry for attackers.

As such, the following alerts have been implemented:

* CPU Usage Monitor
* Excessive HTTP Errors
* SSH Authentication Failure
* Authorization SUDO Command Usage Monitor
* Excessive SSH Authorization

**Monitoring the Targets**

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

* **CPU Usage Monitor**

CPU Usage Monitor is implemented as follows:

* + Metric: Metricbeat
  + Threshold: When max of ‘system.process.cpu.total.pct’ over all documents is above 0.5 for the last 5 minutes.
  + Vulnerability Mitigated: This can mitigate an application(s), video/graphic program(s) or outdated software that is using high CPU usage.
  + Reliability: This alert has a high reliability and can generate lots of false positives due to the many applications/programs that can be executing simultaneously.

* **Excessive HTTP Errors**

Excessive HTTP Errors is implemented as follows:

* + Metric: Metricbeat
  + Threshold: When the count grouped over top 5 ‘http.response.status\_code’ is above 400 for the last 5 minutes.
  + Vulnerability Mitigated: This can mitigate or reduce failed authentication attempts which can be a sign of a brute force attack.
  + Reliability: This alert has a high reliability and can generate lots of false negatives due to the different HTTP status codes that may actually be the result of an attack to the system.

**Monitoring the Targets (con’t)**

* **Excessive SSH Authorization**

Excessive SSH Authentication is implemented as follows:

* Metric:  Filebeat
* Threshold:  When the count grouped over top 3 ‘system.auth.ssh.event’ is above or equals 50 for the last 1 minute.
* Vulnerability Mitigated:  This can mitigate unauthorized users from connecting to a remote host via SSH.
* Reliability:  This alert has a medium reliability and can generate lots of false positives because the request can be valid.
* **Authorization SUDO Command Usage Monitor**

Excessive SSH Authentication is implemented as follows:

* + Metric: Filebeat
  + Threshold: When the count grouped over top 10 ‘system.auth.sudo.user’ is above or equals 5 for the last 1 minute.
  + Vulnerability Mitigated: This can mitigate unauthorized users from escalating root privilege by monitoring all sudo commands executed.
  + Reliability: This alert has a medium reliability and can generate lots of false positives because the user can actually be authorized to use root or execute a sudo command.
* **Excessive SSH Authorization**

Excessive SSH Authorization is implemented as follows:

* + Metric: Filebeat
  + Threshold: When the count grouped over top 3 ‘system.auth.ssh.event’ is above or equals 10 for the last 1 minute.
  + Vulnerability Mitigated: This can mitigate unauthorized users from connecting to a remote host via SSH.
  + Reliability: This alert has a medium reliability and can generate lots of false positives because the request can be valid.

**Suggestions for Going Further**

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 – Outdated Resources/Software**

* Patch: Use the latest version of a software/application (ie. WordPress). Block WordPress username enumeration by editing the theme’s functions.php. Install WordPress plugins. Block the author scans by modifying the .htaccess code.
* Why It Works: Prevents exploitation of a system/network. Outdated resources are vulnerable to attacks.

**Vulnerability 2 – Excessive Authentication Attempts**

* Patch: Password policy that includes password complexity, multi-factor authentication and restriction of login attempts. Enable CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart). Require user to answer secret questions in addition to regular login.
* Why It Works: These mitigation techniques can reduce or prevent brute force attacks.

**Vulnerability 3 – Unauthorized Remote Host Access**

* Patch: Enable multi-factor authentication. Restrict the number of login attempts. Login attempts should be limited to a specific IP address/range. Create an allow list to allow only certain users that can connect via SSH. Change the SSH 22 port used to a different port to prevent these types of attacks.
* Why It Works: Educating employees can reduce threats to the organization. This prevents unauthorized access to the system.

**Vulnerability 4 – Privilege Escalation**

* Patch: Prevent sudo rights to commands that can elevate privilege. Restrict usage of root account. Use the principle of least privilege. Perform system audits. Use a monitoring system.
* Why It Works: This prevents exploitation of the network by unauthorized root users.