Blue Team: Summary of Operations

Table of Contents

Network Topology

The following machines were identified on the network:

- **ML-RefVm-684427/Hyper-V Virtual Ethernet Adapter**
    - **Operating System**: Microsoft Windows 10.0.18363.900 (ipconfig /all)
    - **Purpose**: NATSwitch
    - **IP Address**: 192.168.1.1
    - **Subnet Mask:** 255.255.255.0
- **Kali**
    - **Operating System**: Kali Linux, 5.4.0-kali3-amd64 (uname -r)
    - **Purpose**: Attack Machine
    - **IP Address**: 192.168.1.90
    - **Subnet Netmask:** 255.255.255.0
    - **Broadcast:** 192.168.1.255
- **Target 1**
    - **Operating System**: Debian Linux
    - **Purpose**: Target Machine
    - **IP Address**: 192.168.1.110
- **ELK**
    - **Operating System**: Linux
    - **Purpose**: log correlation and real-time analytics
    - **IP Address**: 92.168.1.100

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 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

Alert 1 is implemented as follows:

- **Metric**: packetbeat
- **Threshold**: WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes
- **Vulnerability Mitigated**: This alert aims to mitigate HTTP errors such as 401 unauthorized errors.
- **Reliability**: This alert can potentially generate false negatives, but with a healthy alert threshold set, generated triggers can be inspected for further insight into the activity.
    - High reliability alert

HTTP Request Size Monitor

Alert 2 is implemented as follows:

- **Metric**: packetbeat
- **Threshold**: WHEN sum() of http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute
- **Vulnerability Mitigated**: This alert aims to mitigate HTTP request smuggling, which is a technique attackers interfere with sites' ability to process HTTP requests.
- **Reliability**: This alert could potentially generate false negatives if attackers stay below the threshold set by the alarm. That could lead to bypassed security controls and data is potentially compromised.
    - Medium reliability

CPU Usage Monitor

Alert 3 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 alert observes the total CPU percent used since the process started or was updated. It aims to mitigate malware, as CPU usage soars when malware starts performing malicious actions.
- **Reliability**: This alert could potentially generate false positives due to certain programs consuming CPU usage in the background, but set at an appropriate level measuring higher than the average usage, it should produce reliable results.
    - High reliability alert

Alert 1: This alert detects HTTP errors such as code 401, Unauthorized or “unauthenticated”. This signals that the client request was not completed due to lack of valid credentials for the requested source. Setting our threshold to 400 in the last 5 minutes signals the potential for malicious activity as that range is abnormal. Attackers may bypass triggering this alert by executing exploits over a longer period of time, thus evading the threshold trigger.

Alert 2: This alert measures the total size of request data and fire when there are greater than 35000 bytes of request data. Noting the size of request data is vital to maintaining boundaries between requests and preventing misinterpretation of requests that can interfere with the application request process. Attackers may bypass triggering this alert by modifying scripts to send out fewer requests, but this could also reduce accuracy.

Alert 3: This alert inspects the total server CPU usage, triggering usage higher than .5. Processes in execution and CPU usage are in constant flux. There can be several contributors to the state of usage, but our focus with this alert is background processes such as malware. Malware initially won’t utilize high resources if it’s hidden in the background but upon execution it will spike usage due to data transfer. For this reason it’s vital to monitor usage as unexpected events can occur, as well as for system optimization.