Blue Team: Summary of Operations

Table of Contents

Network Topology

TODO: Fill out the information below.

The following machines were identified on the network:

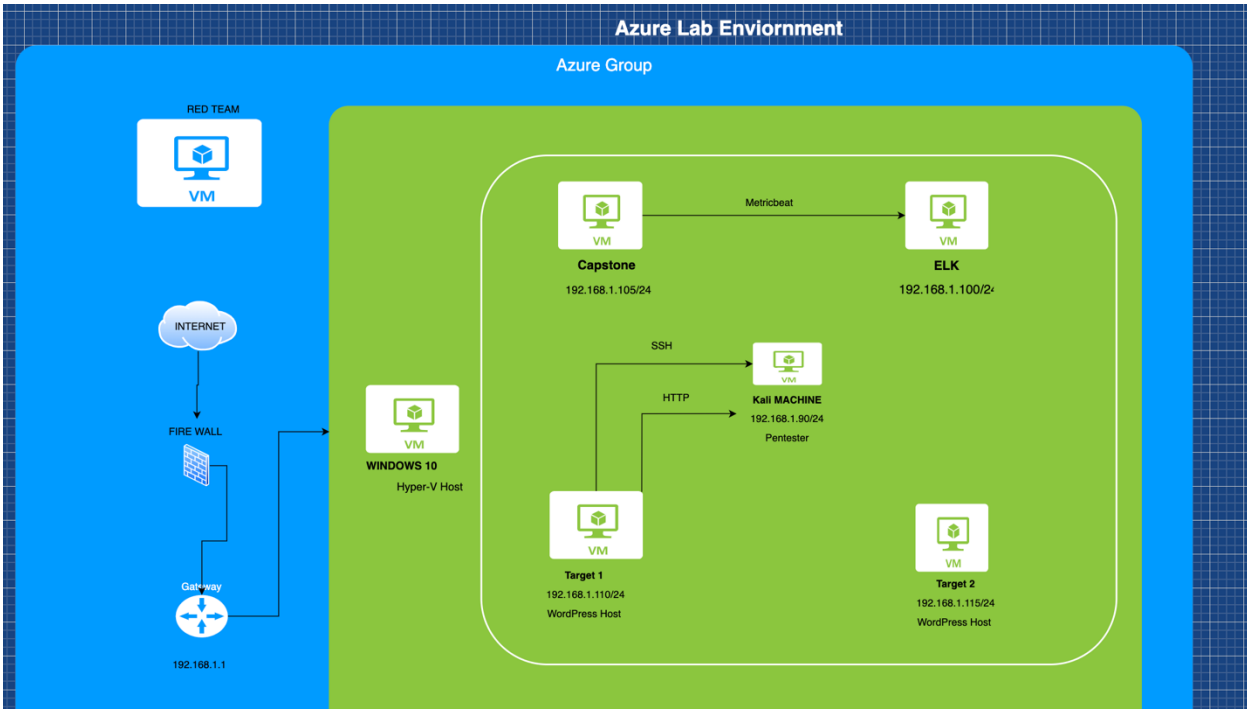
- VM1 KALI MACHINE
    - **Operating System**: Kali Linux 5.4.0
    - **Purpose**: Penetration tester
    - **IP Address**:192.168.1.90
- VM 2 ELK
    - **Operating System**: Ubuntu 18.04
    - **Purpose**: ELK and Kibana
    - **IP Address**:192.168.1.100
- VM3 TARGET 1
    - **Operating System**: Linux
    - **Purpose**: The host of WordPress
    - **IP Address**:192.168.1.110
- VM3 CAPSTONE
    - **Operating System**: Ubuntu 18.04
    - **Purpose**: The vulnerable Web Server
    - **IP Address**:192.168.1.105

Network Diagram:

Address Range: 192.168.1.0 Mask: 255.255.255.0

Gateway: 192.168.1.1

Cloud Services: Azure Lab



Description of Target

The vulnerable virtual machines to attack are Target 1 with Ip address 192.168.1.110 and Target 2 with Ip address 192.168.1.115. In this specific case only, the machine is Target 1, it is covered and was attacked.

Target 1 is an Apache web server and has SSH enabled, so ports 80 and 22 are possible ports of entry for attackers.

Monitoring the Targets

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

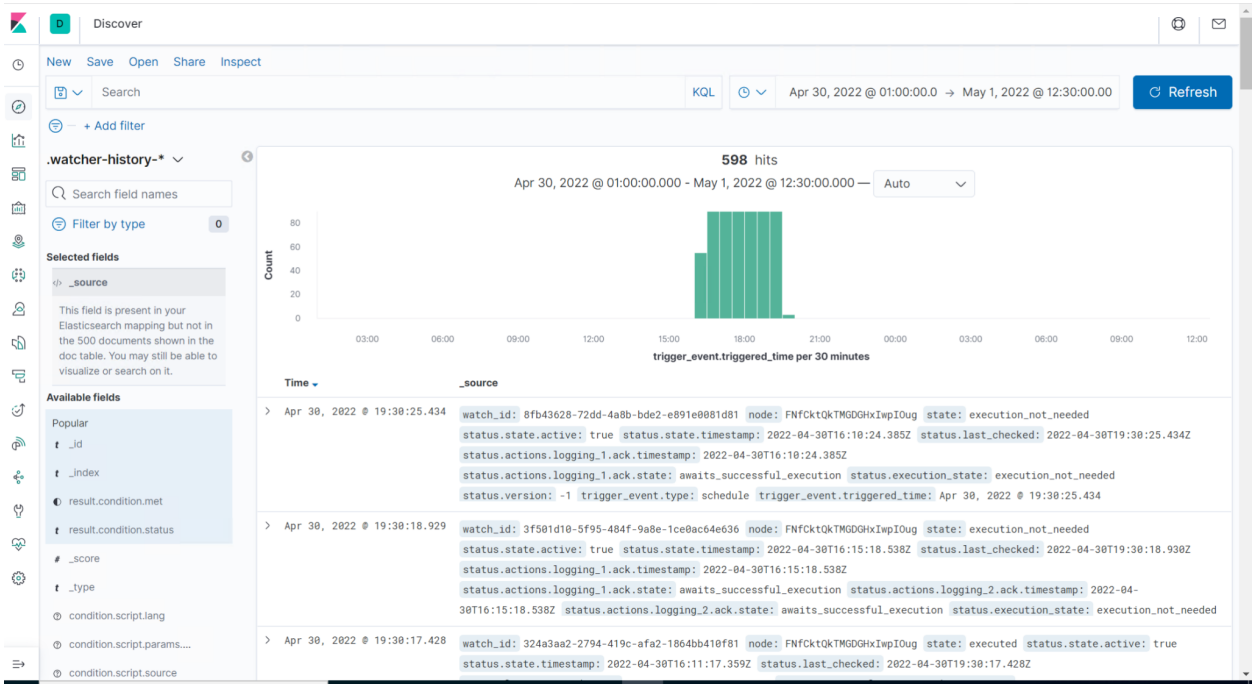
Target 1

Command: Nmap -sV 192.168.1.110

Port: 22/tcp open ssh

Port: 80/tcp open http

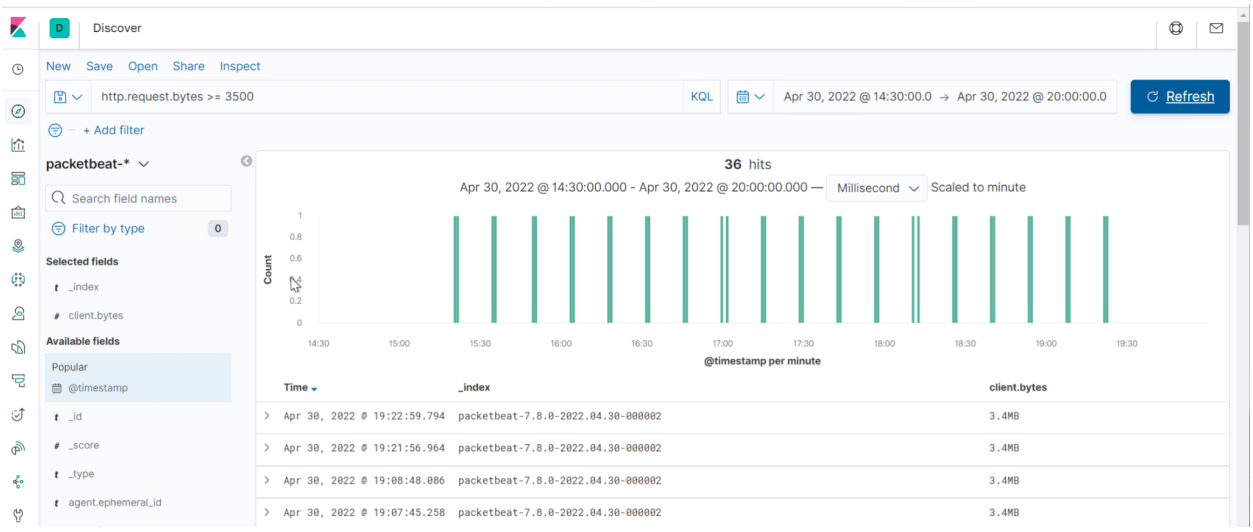
Alert 1: HTTP Request Size Monitor



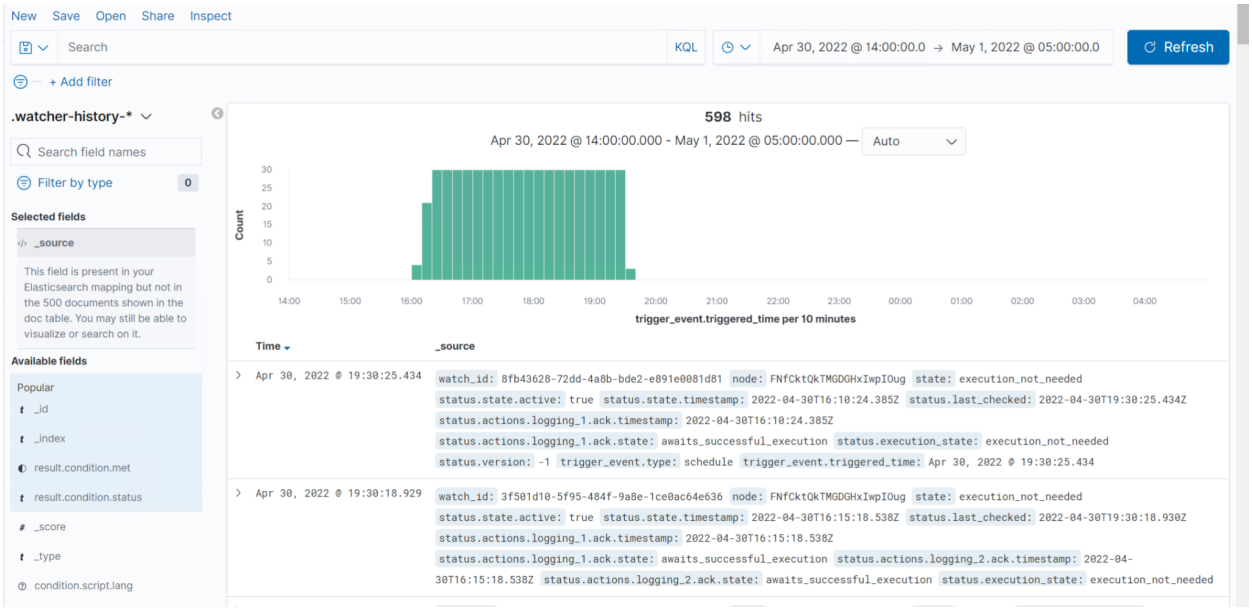
- **Metric**: It was used Metricbeat to collect data and monitoring the cluster in Kibana
- **Threshold**: reached its maximum point when a sum of 3500 for the last minute.
- **Vulnerability Mitigated**: DDos attacks

- **Reliability**: This rule is useful for monitoring logs generated by HTTP requests for identifying DDos attacks**.** So, you can build a visualization that monitors the average amount of bytes served to users over time.

- **Log:** WHEN sum() OF http.request.byes OVER all documents IS ABOVE 3500 DOR THE LAST 1 minute.



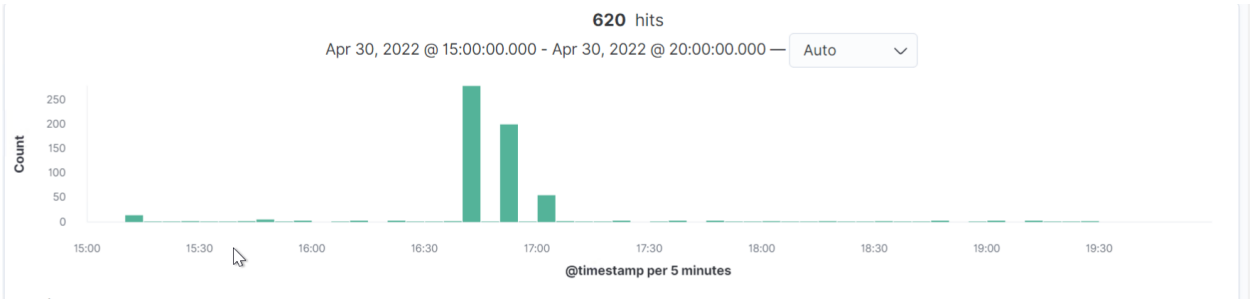
Alert 2: Excessive HTTP Errors



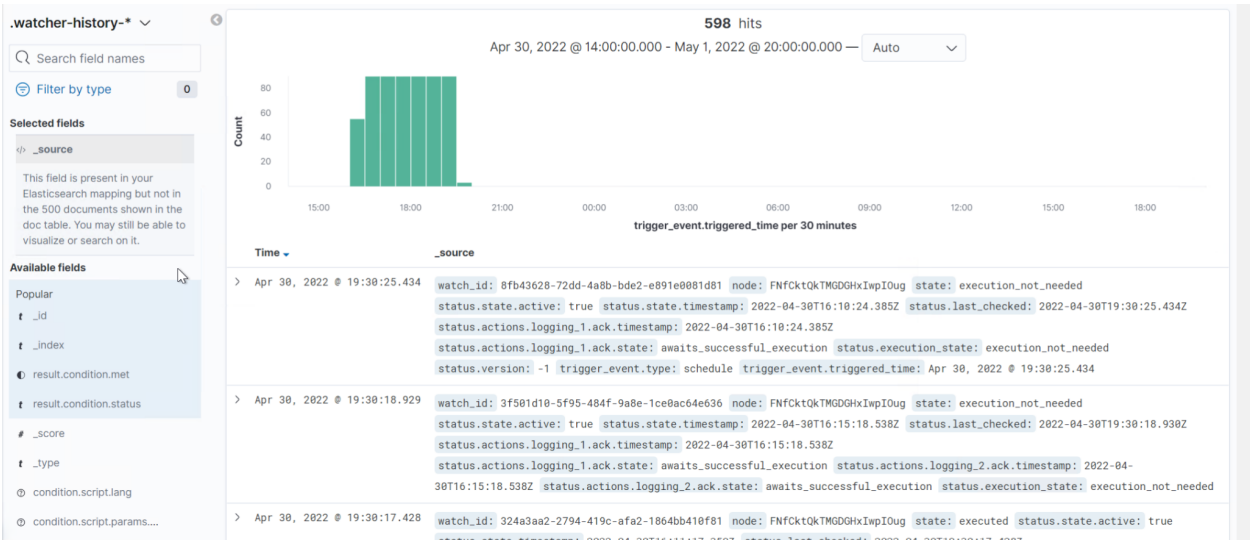
- **Metric**: It was used Metricbeat to show the range of errors occurred in HTTP
- **Threshold**: Reached its maximum point above a count 400 for the last 5 min.
- **Vulnerability Mitigated**: DDos attacks and Enumeration

- **Reliability**: This feature allows you to make a connection between the errors and possible attacks, by linking you to the specific high amount of 404 errors and allows to see the whole trade

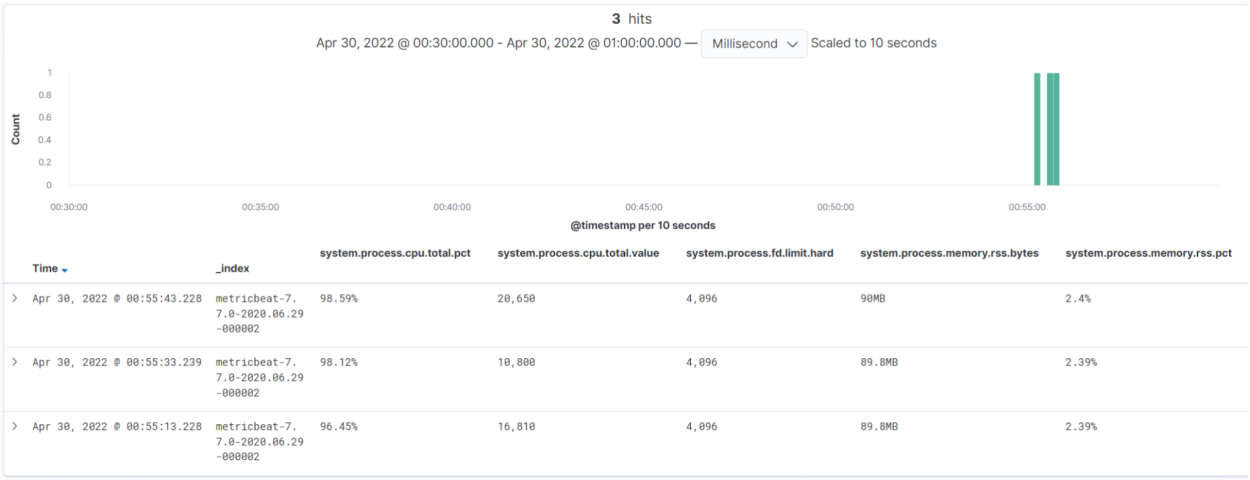
- **Log:** WHEN count () GROUPED OVER top 5 http. response. status_code IS ABOVE 400 FOR THE LAST 5minutes



Alert 3: CPU Usage Monitor



- **Metric**: This rule helps to monitoring the CPU usage based on the report statistics of CPU utilization using system.process.cpu.total.pct with metricbeat.
- **Threshold**: Reached its maximum point when remains above 0.5 over all process for the last 5 minutes.
- **Vulnerability Mitigated**: Denial of Service attack, C2
- **Reliability**. The statistics that we collect help to decide about determine anomalies with such small values that are not interesting for the investigation.
- **Log:** WHEN max () OF system.process.cpu.total.pct OVER all documents IS ABOVE FOR THE LAST 5 minutes.



Suggestions for Going Further (Optional)

As a result of the work in the vulnerabilities identified by the alerts before being exposed, the blue team takes these solutions as suggested to protect the network.

- Vulnerability 1 Hardening Against Vulnerable Ports 22 and 80 on Target 1
  - **Patch**: Close port 22 use port 443 withs http instead 80
  - **Why It Works:** Port 22 will prevent open ssh access to the machine. Using port 443 will provide a layer of security using ssh instead of the open port.
  - Each command should be run one at a time and checked status with sudo ufw status verbose.
  - Command use for shutting down the Ports 22 and 80

    sudo ufw deny PORT 80

    sudo ufw deny PORT 22

    sudo ufw allow port 443

- Vulnerability 2: Hardening Against Weak/ Insecure Passwords on Target 1
  - **Patch**: the policy of the users should change passwords to protect their accounts will require at least 16 characters, no dictionary words, special characters, numbers, and symbols must be included too. After 5 unsuccessful attempts it will lock out for an hour. Multi-factor authentication will need to be used.
  - **Why It Works**: The complexity of the passwords will make it harder and difficult for hackers to crack or brute force, the implementation of the lock outs after 5 unsuccessful attempts help to prevent it too. Generating notification alerts about logins will further the protection of the accounts.
- Vulnerability 3: Hardening Against Python Privilege Escalation Target 1
  - **Patch**: Python privileges should be removed for users with vulnerabilities like ssh to the ones that are not authorized for roots privileges.
  - **Why It Works**: Access restringing should be stronger at the moment to remove the phyton privileges.

    Command: vi/etc/sudoers:

    this line must be deleted:

    steven ALL+(ALL) NONPASSWORD: /usr/bin/python

- Vulnerability 3: Hardening Against Enumerate WordPress Site on Target 1
  - **Patch**: Deploy the Ansible Playbook to update the WordPress last version with Stop User enumeration plug-in and firewall to block enumerating traffic.

    - **Why It Works**: Updates versions of WordPress with appropriate plugins that won't allow enumeration.
    - https:// wordpress.org/plugins/stop-user-enumeration/
- Vulnerability 5 Hardening Against Apache 2.4.1 CVE-2016-4975 on Target 1
    - **Patch**: Update Apache server to the last version
    - **Why It Works**: Users required to update the last version to keep in constantly to encourage and prior the vulnerabilities already fix it in past versions, there have been significant changes in authorization configuration and some other changes that will help to be protected.