Final Engagement

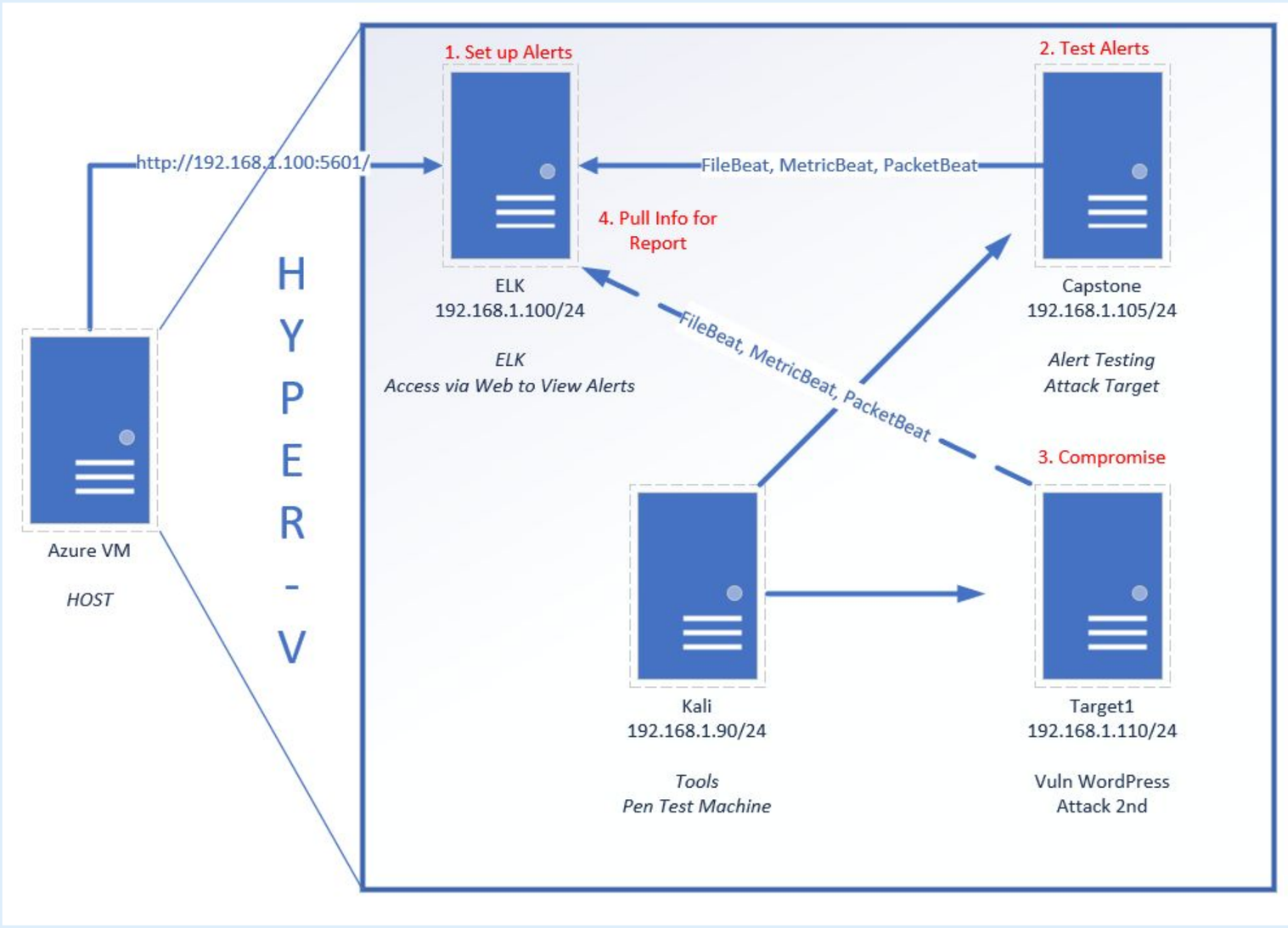
Attack, Defense & Analysis of a Vulnerable Network

Table of Contents

This document contains the following resources:

Network Topology & Critical Vulnerabilities

Network Topology



Network
Address Range: **192.168.1.0/24**
Netmask: **255.255.255.0**
Gateway: **192.168.1.1**

Machines
IPv4: **192.168.1.105**
OS: **Linux**
Hostname: **Capstone**

IPv4: **192.168.1.90**
OS: **Linux**
Hostname: **Kali**

IPv4: **192.168.1.110**
OS: **Linux**
Hostname: **Target 1**

IPv4: **192.168.1.100**
OS: **Linux**
Hostname: **ELK**

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

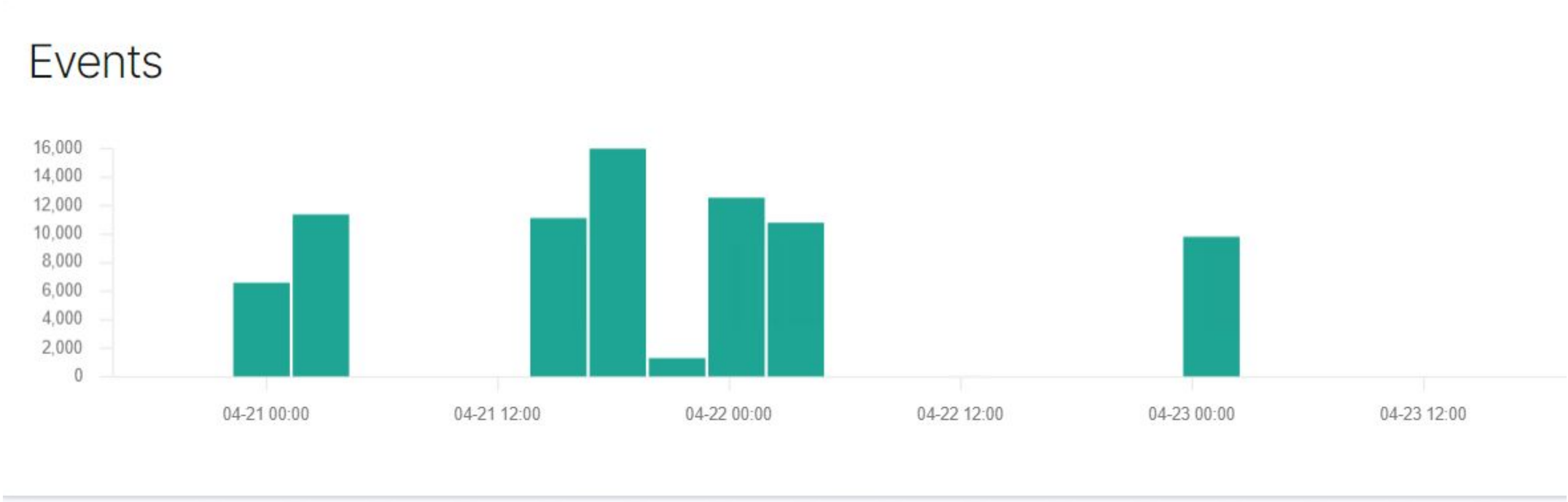
Vulnerability	Description	Impact
CVE-2008-5161	SSH remote login active at the user level; open port 22	The attacker gained access to user accounts.
CVE-2017-7760	Exposed username which allowed for brute force of user account information.	User access to the wp-config.php file, exposing wordpress credentials.
CVE-2012-6707	Weak MD5-based password hashes in wordpress database	The attacker was able to easily crack user account credentials using and gain root access.

Alerts Implemented

Excessive HTTP errors

- This metric monitors HTTP response status codes over time.

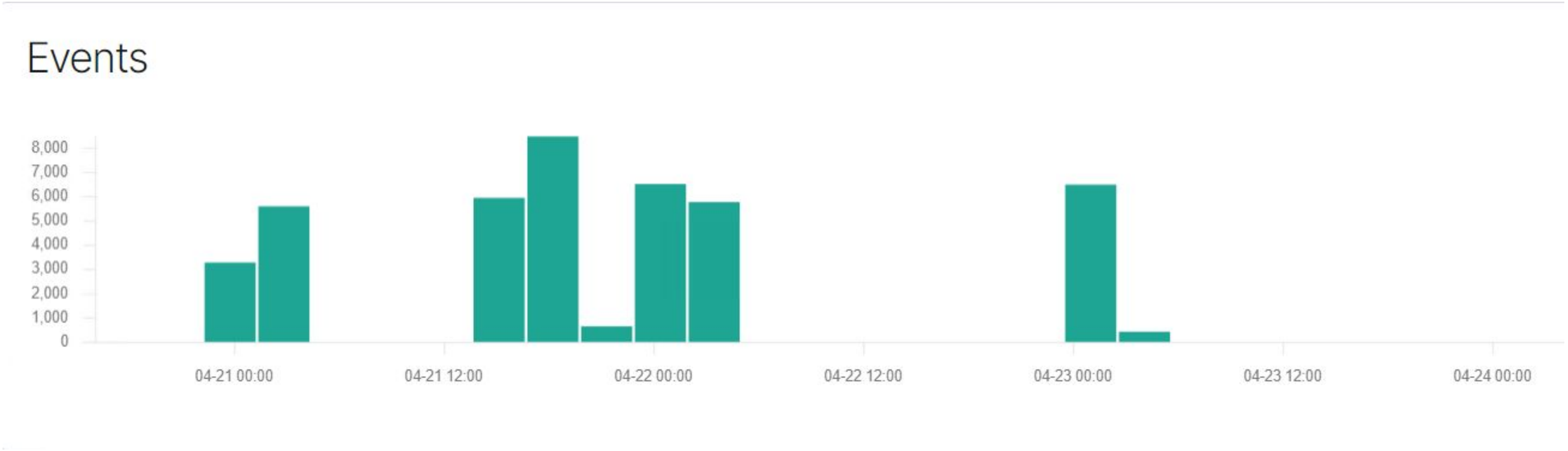
- It will fire when the amount of error codes exceeds the predetermined threshold.



HTTP Request Size Monitor

- This metric monitors the size of HTTP Request packets.

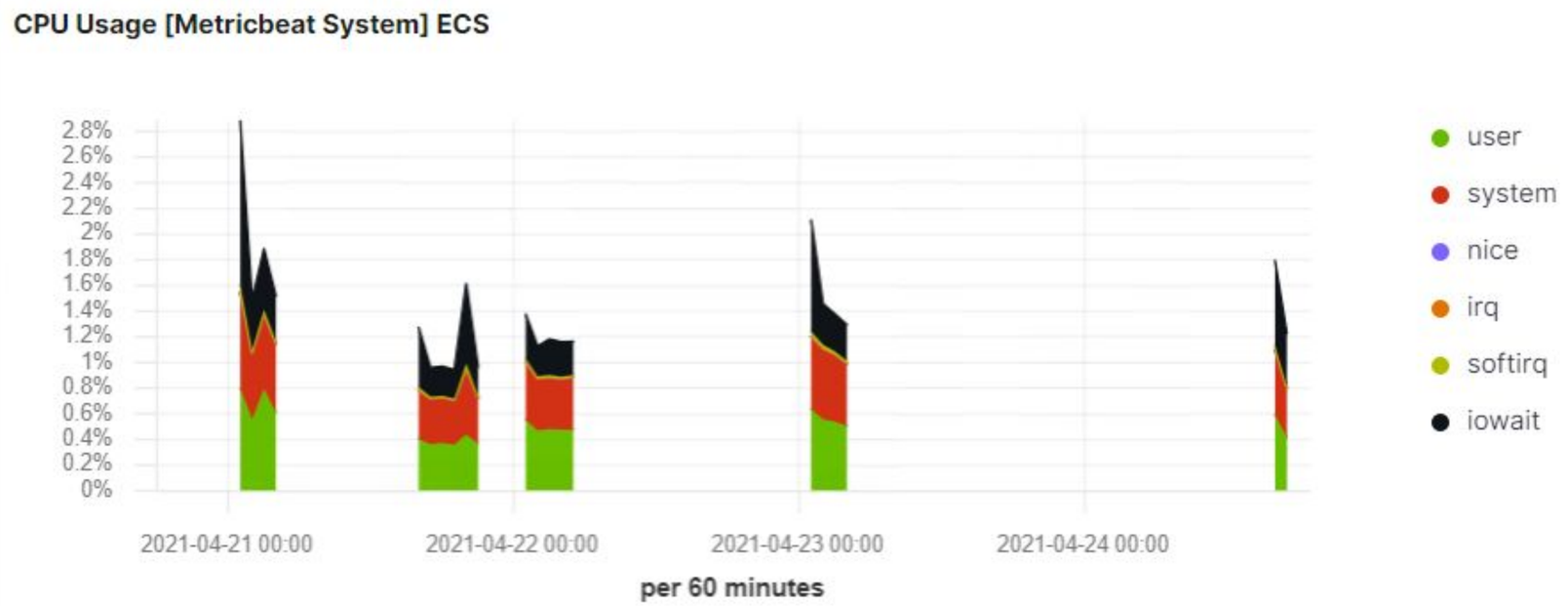
- It will fire when the predetermined threshold has been exceeded.



CPU Usage Monitor

- This metric monitors CPU usage over time.

- It will fire when the CPU usage exceeds a predetermined limit.



Hardening

Hardening Against Open SSH port on Target 1

- One solution to this issue is to redefine firewall rules so that the port will be closed to access outside the internal network.

- Another solution would be to use a custom SSH port. This can be configured as follows:
    - nano -w /ec/ssh/ssh_config
    - Scroll down to ports and set a custom port in place of port 22.

Hardening Against Weak Login Credentials

- The password for user account Michael was the same as the username, allowing quick and easy access into the account via SSH.

- It is recommended to always use complex passwords for user accounts, and credentials should **NEVER** be the same for both username and password.

Hardening Against Wordpress Access

- In this activity, the attacking team was able to gain valuable information about Wordpress (login credentials) from the wp-config.php file. In order to prevent outsiders from gaining access to this valuable file, it is necessary to secure the file.
- One of the best measures is to protect the wp-config.php file using a .htaccess configuration file. To do this, connect to the webpage using an FTP client and download the .htaccess file. Edit the file using the below syntax. Then, upload it to root directory to overwrite the old file. Individual configurations can then be edited as needed.

```
# protect wpconfig.php
 <files wp-config.php>
 order allow,deny
 deny from all
 </files>
```

Implementing Patches

Implementing Patches with Ansible

Playbook Overview

```
- hosts: all
  vars:
  allowed_ssh_networks:
- 192.168.1.110
- 10.10.10.0/24
  - name: Add local user
    user:
    name:admin
    group:admin
    shell: /bin/bash
    home: /home/admin
    create_home: yes
    state: present
-name: Add SSH public key for user
authorized_key:
    user: admin
    key: “{{ lookup(‘file’, ‘~/.ssh/id_rsa.pub’) }}”
-name: Add hardened SSH config
copy:
    dest: /etc/ssh/sshd_config
    src: etc/ssh/sshd_config
    owner: root
    group: root
    mode: 0600
notify: Reload SSH
-name: Add SSH port to internal zone
firewalld:
zone: internal
service: ssh
state: enabled
immediate: yes
permanent: yes
```

Ansible Playbook Continued

```
-name: Add permitted networks to internal zone
firewalld:
    zone: internal
    source: “{{ item }}”
    state:enabled
    immediate: yes
    permanent: yes
with_items: “{{ allowed_ssh_networks }}”
-name: Drop ssh from the public zone
firewalld:
    zone: public
    service: ssh
    state: disabled
    immediate: yes
    permanent: yes
```

```
handlers:
    -name: Reload SSH
    service:
      name: ssh
      state: reloaded
```