Final Engagement

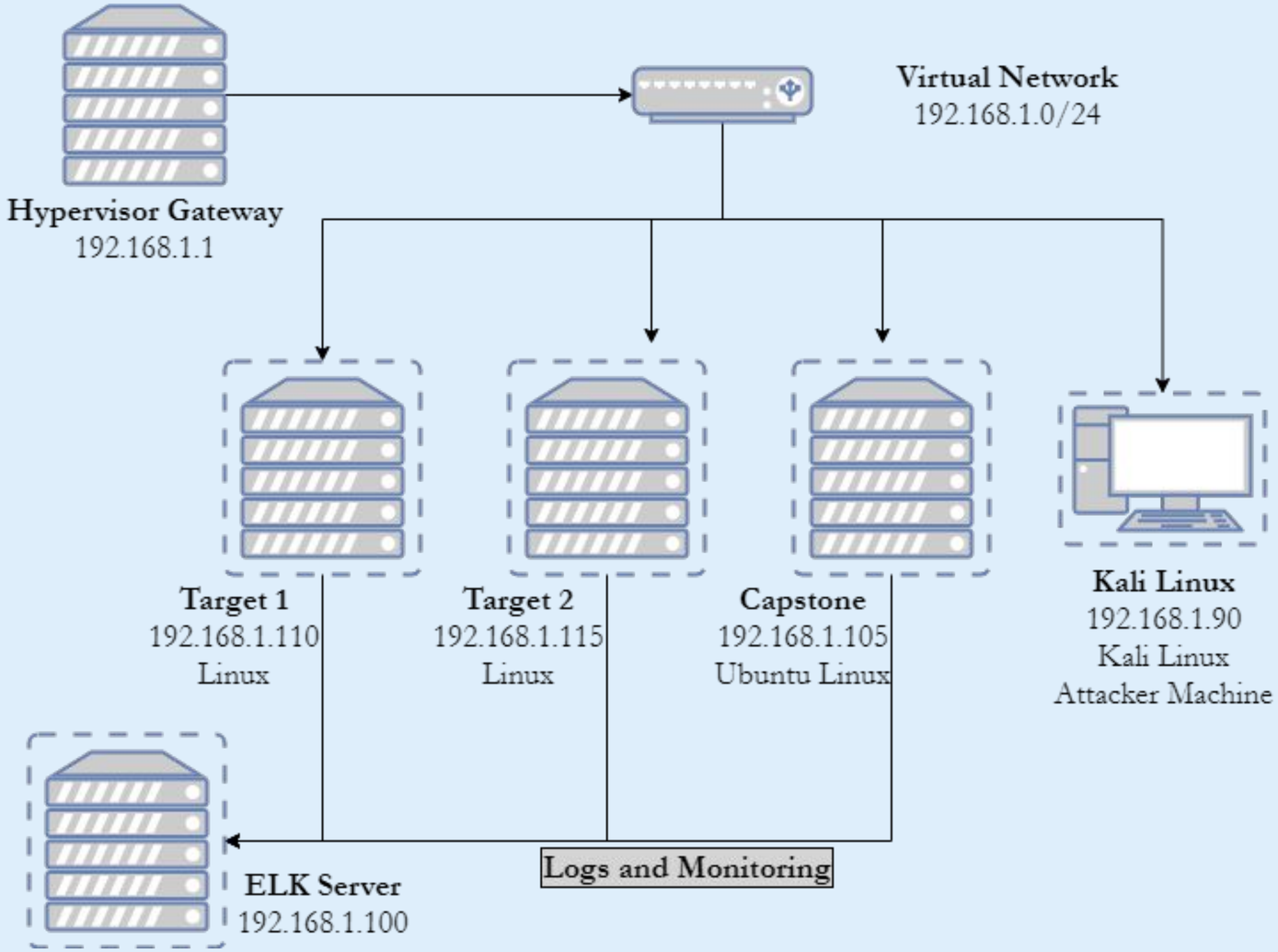
Attack, Defense & Analysis of a Vulnerable Network

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: 192.168.1.100
OS: Ubuntu Linux
Hostname: ELK

IPv4: 192.168.1.110
OS: Linux
Hostname: Target 1

IPv4: 192.168.1.115
OS: Linux
Hostname: Target 2

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

Critical Vulnerabilities: Target 1

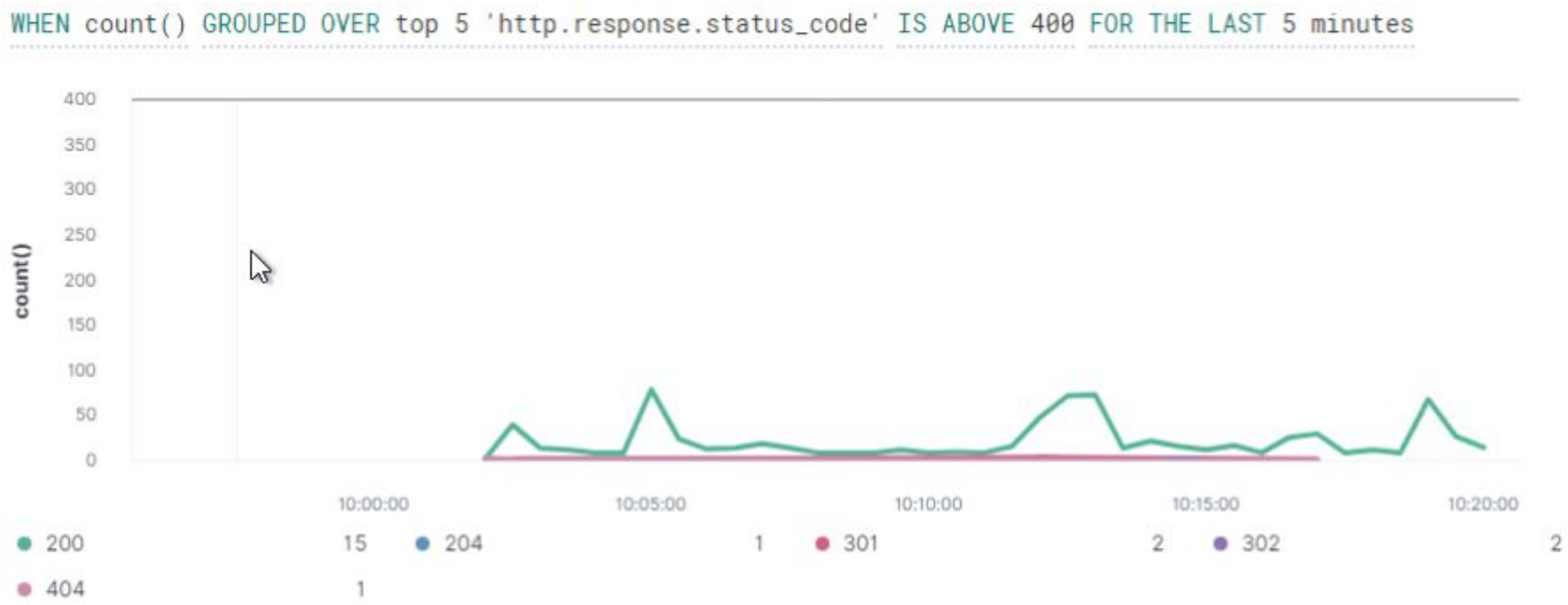
Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Open access to SSH	An attacker can try SSH logins if the port is left open	Open SSH is vulnerable to brute force attacks
Brute Force Vulnerability	Attackers can perform rapid attempts to guess user names and passwords	Brute force will, given enough time, gain access to the system
Enumerate Wordpress	Unsecured Wordpress allows for information gathering and vulnerability assessment	Wordpress stored usernames, greatly reducing the time required to brute force

Alerts Implemented

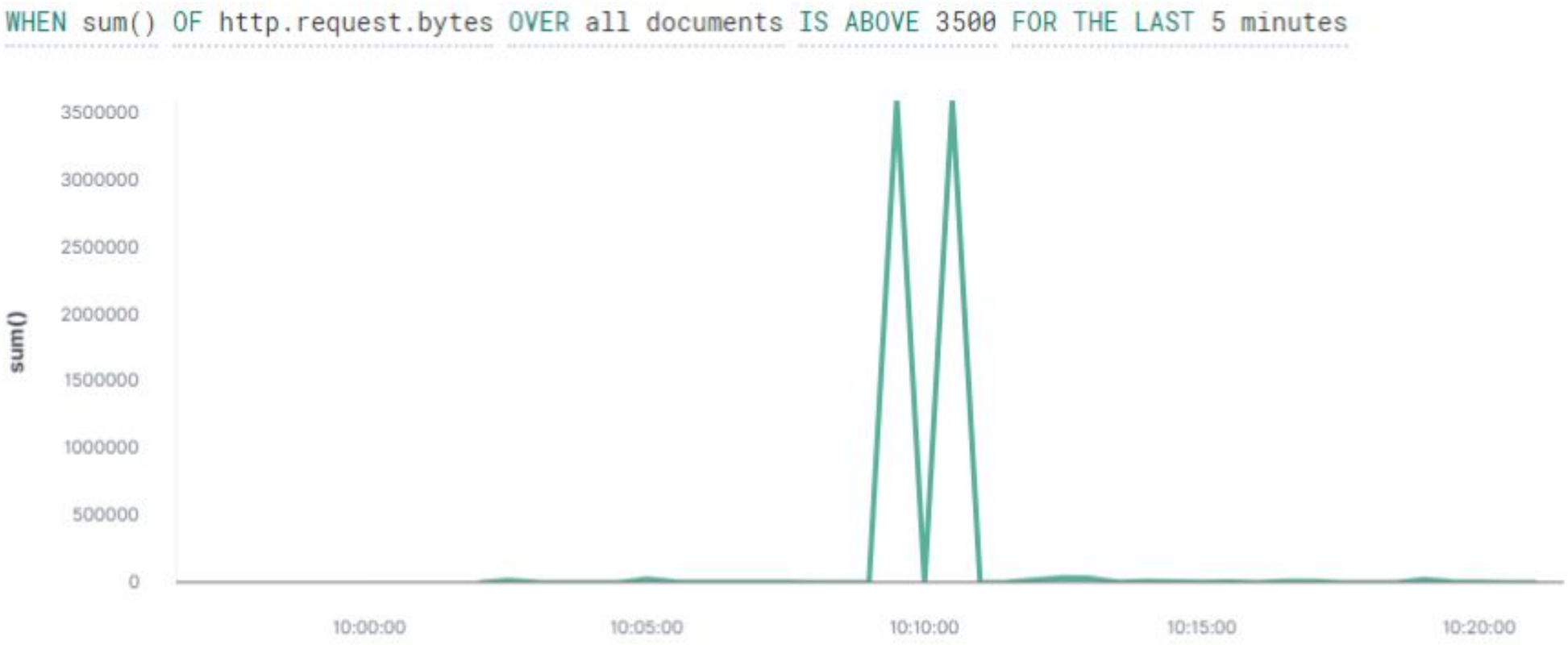
Excessive HTTP Errors

- Utilise Packetbeat to monitor *http.response.status_code*
- Implement a threshold of 400 for the last 5 minutes



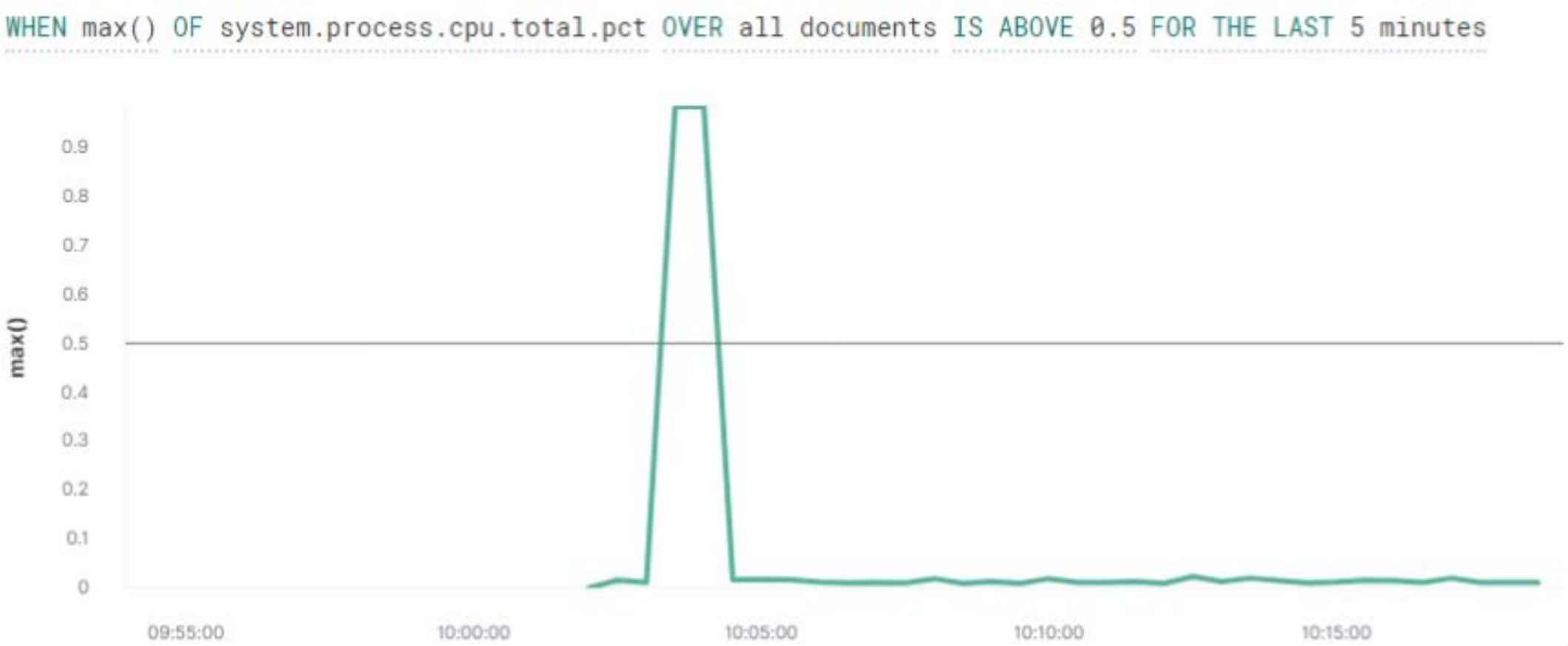
HTTP Request Size Monitor

- Utilise Packetbeat to monitor *http.request.bytes*
- Implement a threshold of 3500 for the last 5 minutes
- Provide a screenshot of the alert in action.



CPU Usage Monitor

- Utilise Metricbeat to monitor *system.process.cpu.total.pct*
- Implement a threshold of 0.5 for the last 5 minutes



Hardening

Hardening Against Open SSh on Target 1

There are a wide variety of hardening techniques for SSH. These can include:

- Set a custom TCP port by editing the */etc/ssh/sshd_config* file.
- Filter the SSH port through the firewall
- Implement SSH Passwordless Login. Uses keys to allow for login and removes the password prompt
- Disable empty passwords
- Set a custom SSH login banner. Doesn't stop logins but is a warning of active monitoring
- Keep SSH updated

Hardening Against Brute Force Vulnerability on Target 1

The simplest defense against a brute force attack is to implement an account lockout policy, however this leads to other forms of attack such as a Denial of Service or username harvesting. Other defenses include

- Implement a strong password policy
- Multi factor authentication
- Captcha
- Asking a ‘secret question’ after two failed login attempts

Hardening Against Wordpress Enumeration on Target 1

As Wordpress is a website builder with a large range there are many security aspects to consider:

- Keep Wordpress up to date.
- Disable **REST API** and **XML-RPC** if they are not being used
- Configure your web server to block requests to **/?author=<number>**
- Don’t expose **/wp-admin** and **/wp-login.php** directly to the public Internet

Implementing Patches

Implementing Patches with Ansible

Playbook Overview

Ansible will be used to automatic the update process on each machine. This process can be scheduled to run on a regular basis through cron job. The following updates the Linux-based webserver:

name: System Update

hosts: webservers

apt: update_cache=yes force_apt_get=yes cache_valid_time=3600

- name: reboot

- register: reboot_required_file

- stat: path=/var/run/reboot-required get_md5=no