Capstone Engagement

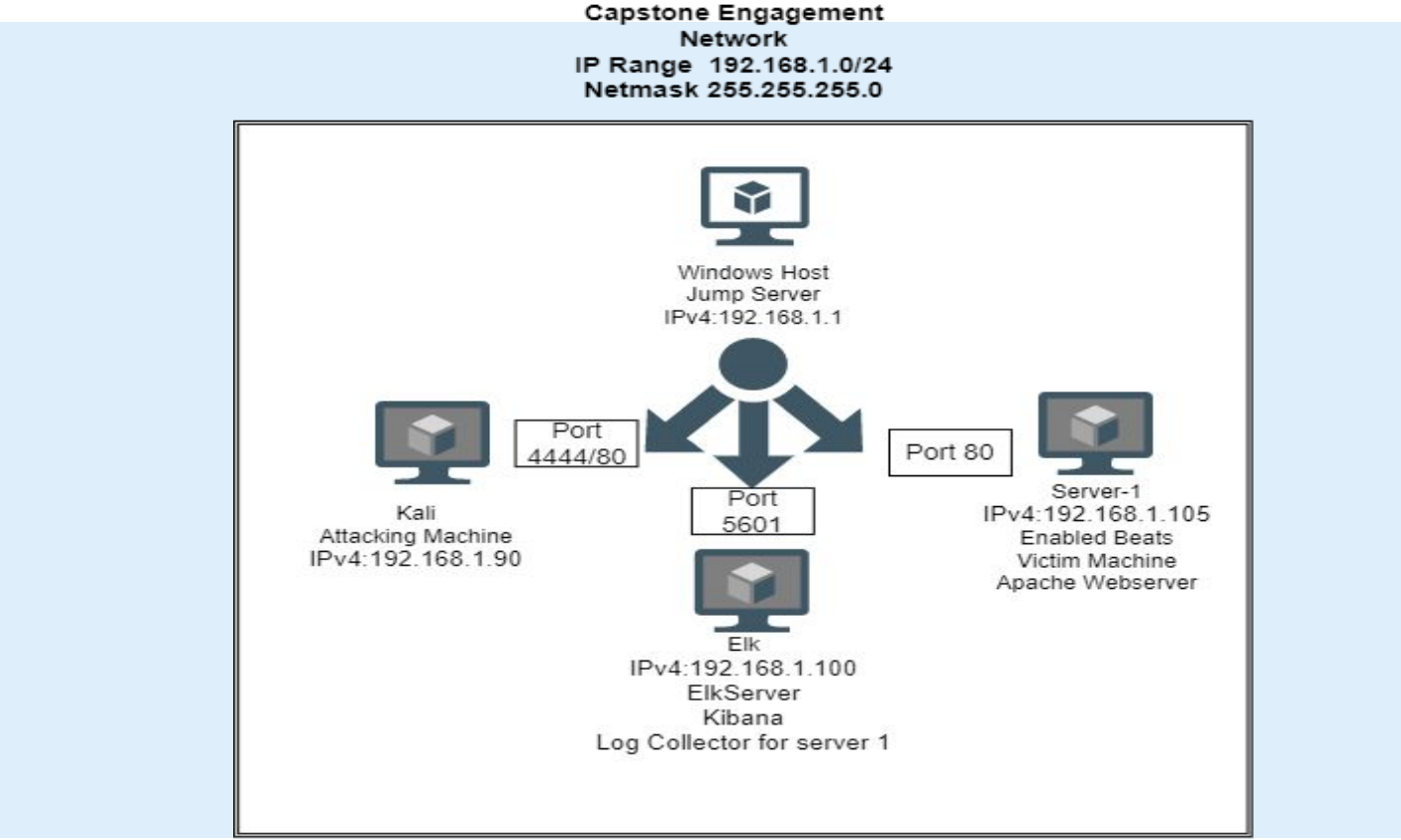
Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

Network Topology

Network Topology



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

Machines
IPv4:192.168.1.90
OS:kali
Hostname:Kali

IPv4:192.168.1.105
OS:Ubuntu 18.04.1 LTS
Hostname:server1

IPv4:192.168.1.100
OS:Linux
Hostname:Elk

IPv4:192.168.1.1
OS:Windows
Hostname:
ML-RefVm-684427

Red Team

Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Machines Kali	IPv4:192.168.1.90	Attacking Machine
Server1	IPv4:192.168.1.105	Enabled beats to Elk server Apache Webserver (capstone) Victim machine
Elk	IPv4:192.168.1.100	Elk Server(Kibana) -Log Collector for server1
Windows Host	IPv4:192.168.1.1	Jump Server

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Reverse Shell TCP	*Allows attacker to gain access to shell on victim's machine with equal privileges, lets attacker establish a meterpreter session.*	*You can access files, traverse directories, download content (screenshot, read/write, remote delete files, execute scripts, stop server).*
Weak Passwords	Easily found using password dictionary.	Puts computer and network at risk of being compromised.
Brute Force Login	Login to a password protected directory, by providing a variety of combination	Identity theft, install malware, loss of data, downtime
Port Scan	Used NMAP to scan open ports on the network.	This allowed me to find vulnerable ports so that I could further exploit.

Exploitation: Port Scan

01

Tools & Processes

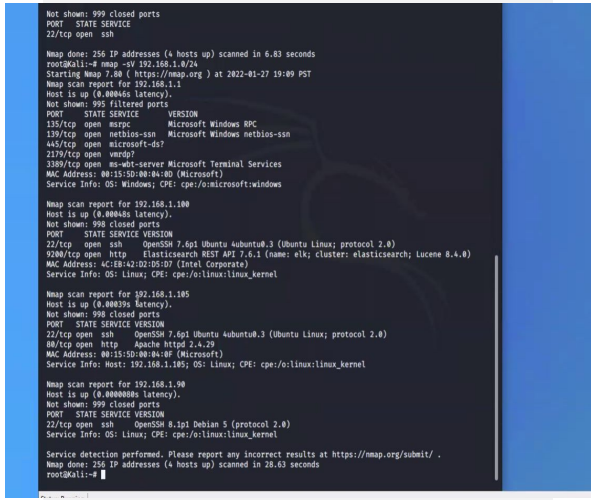
In order to find IP address of the machine, I used Nmap to scan network.

02

Achievements

From Nmap we see that port 80 is open Apache Web Server. I typed IP address in web browser and located hidden directory.

03



```
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 256 IP addresses (4 hosts up) scanned in 6.83 seconds
root@Kali:~# nmap -sV 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-01-27 19:09 PST
Nmap scan report for 192.168.1.1
Host is up (0.00046s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
2179/tcp open  vmrdp?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.1.100
Host is up (0.00046s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
9200/tcp open  http    Elasticsearch REST API 7.6.1 (name: elk; cluster: elasticsearch; Lucene 8.4.0)
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.105
Host is up (0.00039s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.90
Host is up (0.00000080s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.1p1 Debian 5 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (4 hosts up) scanned in 28.63 seconds
root@Kali:~#
```

Exploitation: Brute Force

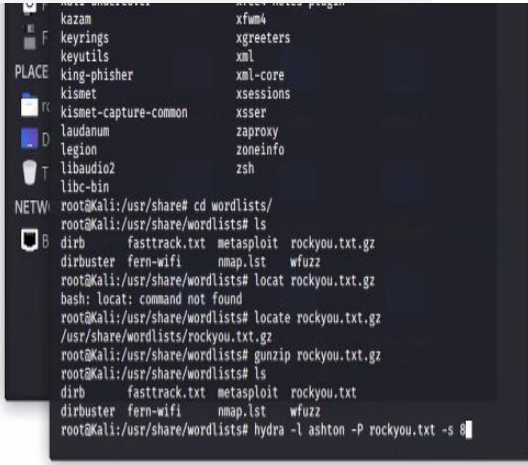
01

Tools & Processes
Used Hydra attack against directory.

02

Achievements
I was able to gain privileges through finding username “ashton” and password “leopoldo.”

03



- hydra -l ashton -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder

Exploitation: Reverse TCP

01

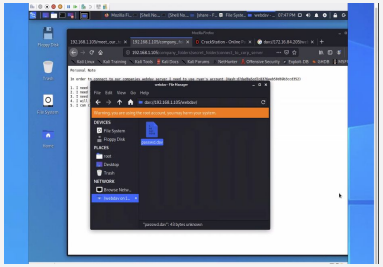
Tools & Processes

Using the newly acquired login credentials, I opened a file called “connecting_to_webdav.” In that file I found username and a hashed password. Using CrackStation I solved the hash and gained the resulting password “linux4u.” Next I connected to the VM’s WebDAV directory and entered newly gained credentials.

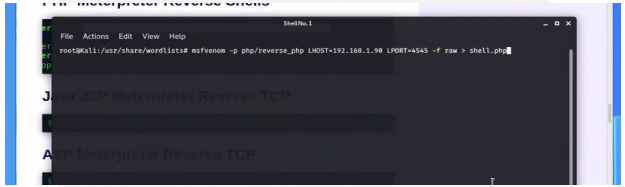
02

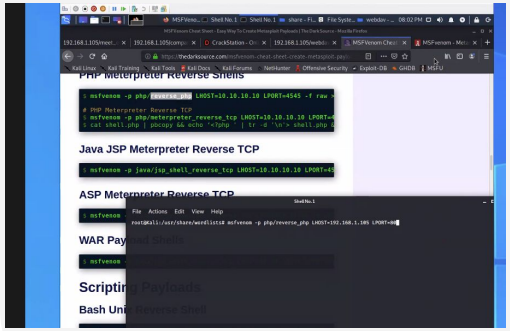
Achievements

Uploaded PHP reverse shell payload in order to set up a listener command on Meterpreter and uploaded PHP file to WebDAV directory.



03

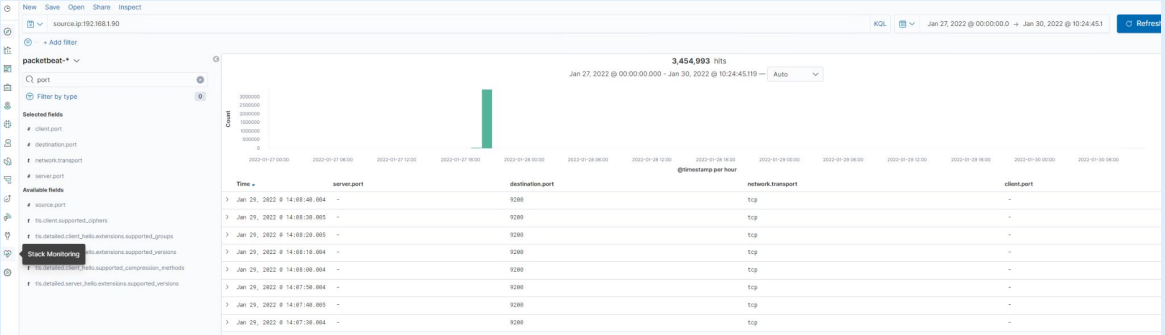




Blue Team

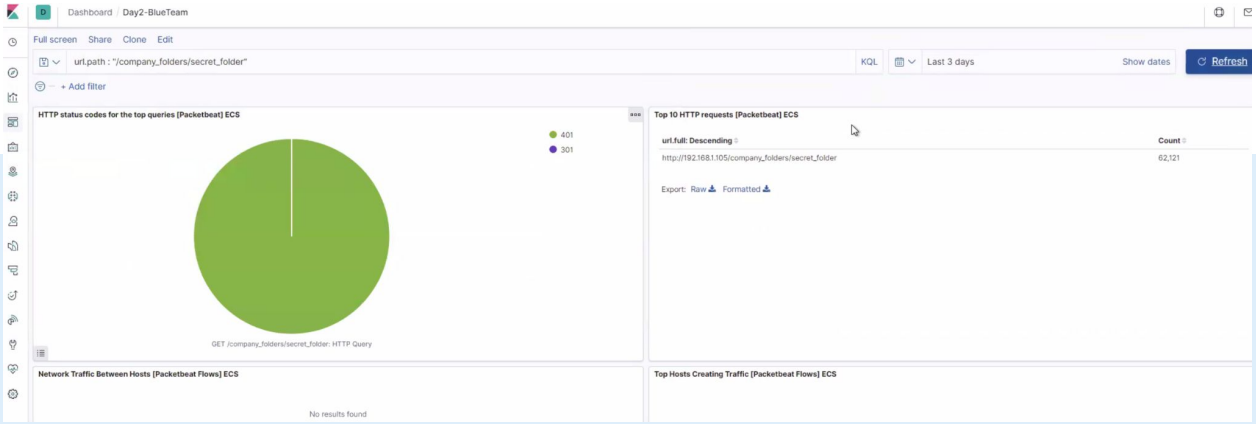
Log Analysis and Attack Characterization

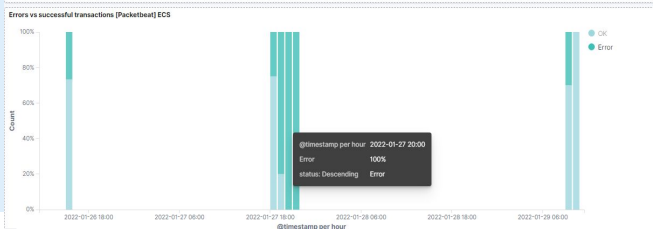
Analysis: Identifying the Port Scan



- Port Scan occurred Jan 27, 2022
- Source IP 192.168.1.90
- The large quantity of scans in a short amount of time indicates an attack

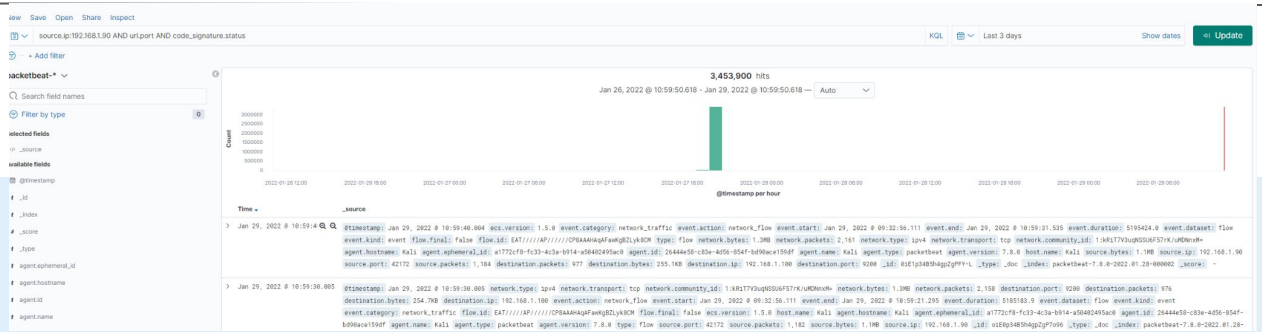
Analysis: Finding the Request for the Hidden Directory





- 62,121 requests made at 13:20:13
- Secret_folder requested containing WebDAV access instructions

Analysis: Uncovering the Brute Force Attack

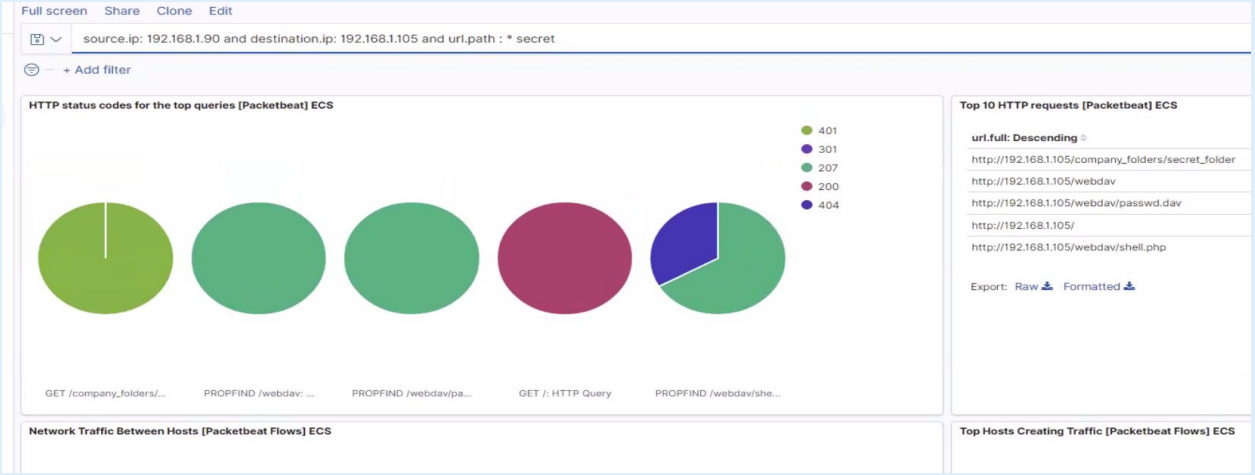


- 3,453,900 Hits
- 469,523 requests made before the attacker discovered the password

Analysis: Finding the WebDAV Connection

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	469,523
http://192.168.1.105/webdav	2



- 2 WebDAV requests
- Shell.php and Passwd.dav requested

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

Create alarm when threshold of 500 ports scanned in 5 minutes occurs.

System Hardening

Set configurations on the host to mitigate port scans by blocking all ports not in use by configuring settings and setting up new rules. In Linux one option is run “sudo apt install firewalld”, “sudo service firewalld status, sudo firewall-cmd -
-remove-port=22/tcp
- -permanent”, “sudo firewall-cmd
- -remove-port=22/udp - -permanent,” for example. There are lots of ways to set firewalls and close ports.

Mitigation: Finding the Request for the Hidden Directory

Alarm

An alarm can be set to detect future unauthorized access when a non-Whitelist IP tries to access network. Set the threshold at 5 in an hour.

System Hardening

Set configuration on the host to block unwanted access by only allowing people from the Whitelist. In Linux you could run from root:
“# iptables -A INPUT -s <ip address> -j ACCEPT”

For all Whitelisted IPs.

Mitigation: Preventing Brute Force Attacks

Alarm

An alarm can be set to detect future brute force attacks by setting a threshold of 3 failed logins in a minute block. We could also add CAPTCHA and two-factor authentication.

System Hardening

We could add whitelisted IP's or use BotGaurd. Additionally we could do a lock out after too many failed attempts.
Ex: auth required <account> deny=3 even_deny_root unlock_time=600 oner=fail
account required <account>

Mitigation: Detecting the WebDAV Connection

Alarm

Alarm on non-Whitelisted IP on WebDAV (threshold 1) and make sure my firewall blocks all others.

System Hardening

In Ubuntu I would run the following command: $ iptables -A INPUT -s 192.168.1.0/24 -j DROP

Mitigation: Identifying Reverse Shell Uploads

Alarm

Do not allow upload of code files including PHP (threshold 1). Also trigger alarm for all attempts on port 4444.

System Hardening

Set access to read only and have whitelisted IPs. Also insure that only necessary ports are open.

the End