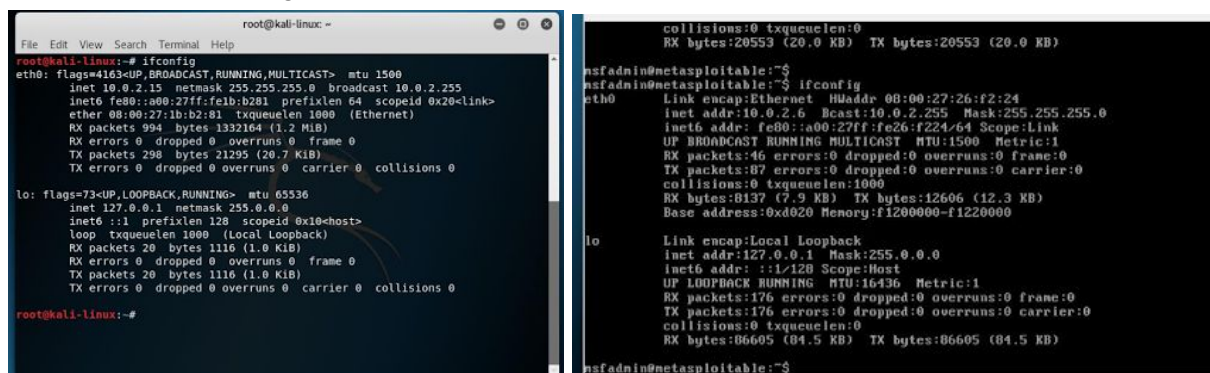


Lab 3: Pentesting

Goal: To use Kali to perform penetration testing towards Metasploitable. I will try to do my best based on what I can remember from the zoom lecture.

- The procedure is the same as the last lab. We first have to run “**ifconfig**” on both machines to figure out their IP addresses.



The image shows two terminal windows side-by-side. The left window is Kali Linux, and the right window is Metasploitable. Both show the output of the 'ifconfig' command.

```
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 1110 (1.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20 bytes 1110 (1.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@kali-linux:~#
```

```
nsfadmin@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:12686 (12.3 KB)
    Base address:0xd020 Memory:f1200000-f1220000

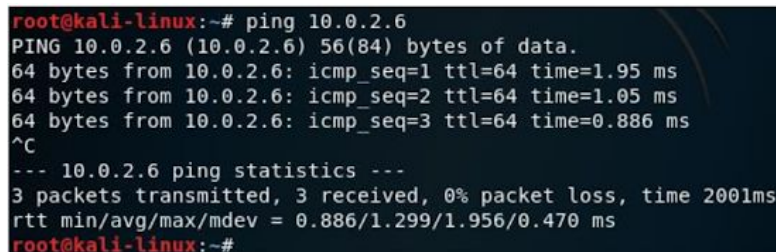
lo
    Link encap:Local Loopback
    inet addr:127.0.0.1 Mask:255.0.0.0
    inet6 addr: ::1/128 Scope:Host
    UP LOOPBACK RUNNING MTU:16436 Metric:1
    RX packets:176 errors:0 dropped:0 overruns:0 frame:0
    TX packets:176 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:0
    RX bytes:86695 (84.5 KB) TX bytes:86695 (84.5 KB)

nsfadmin@metasploitable:~$
```

Kali: 10.0.2.15

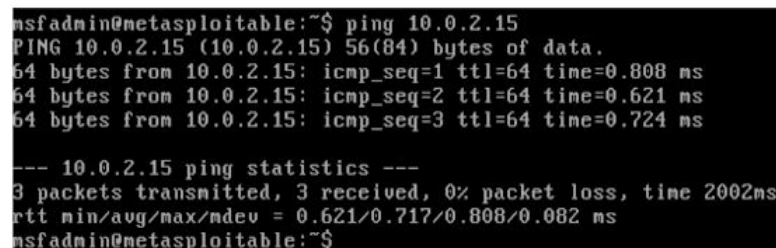
Metasploitable: 10.0.2.6

- Next is to run the **ping** command to see if the machines can communicate with one another.



```
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:~#
```

The ping to Metasploitable is successful and is demonstrated by pinging Metasploitable’s IP address: **10.0.2.6**



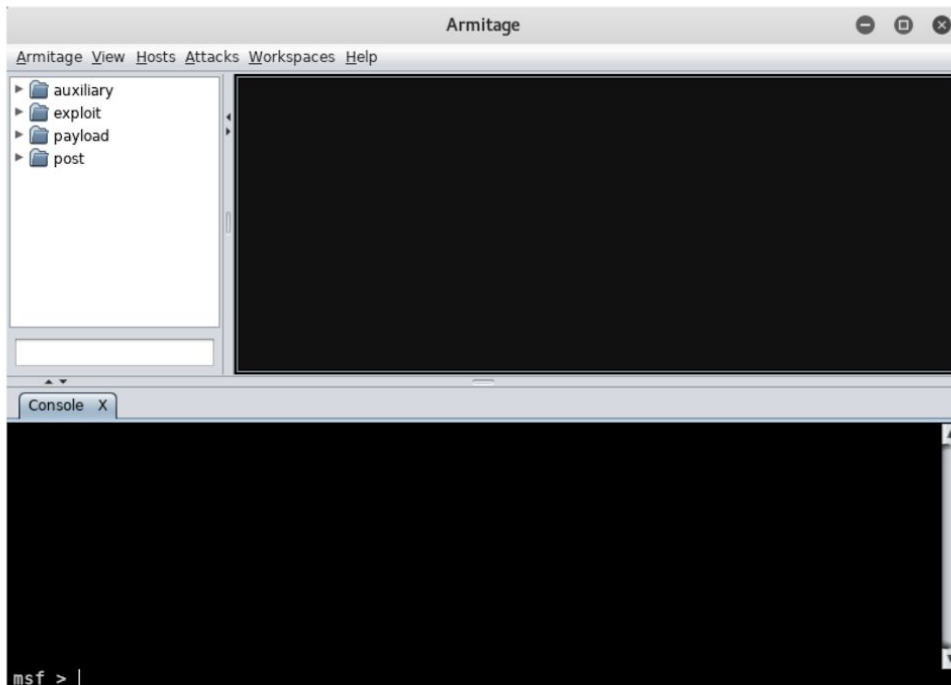
```
nsfadmin@metasploitable:~$ ping 10.0.2.15
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.
 64 bytes from 10.0.2.15: icmp_seq=1 ttl=64 time=0.808 ms
 64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=0.621 ms
 64 bytes from 10.0.2.15: icmp_seq=3 ttl=64 time=0.724 ms

--- 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
nsfadmin@metasploitable:~$
```

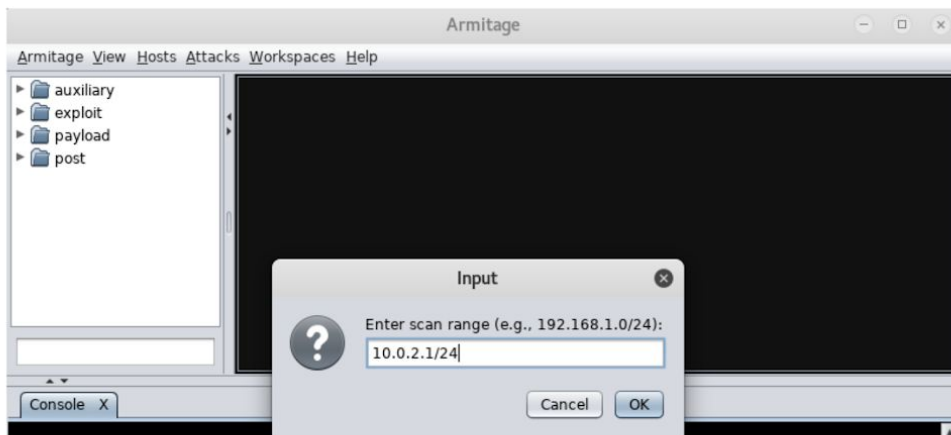
The ping to Kali is successful and is demonstrated by pinging Kali’s IP address:

10.0.2.15

- Next Step is to run the command “**service postgresql start**”, which is followed by “**armitage**”. After some trial and error, we will finally have the graphical user interface of armitage read for us to use. This allows us to do attack without having to run commands through terminal.

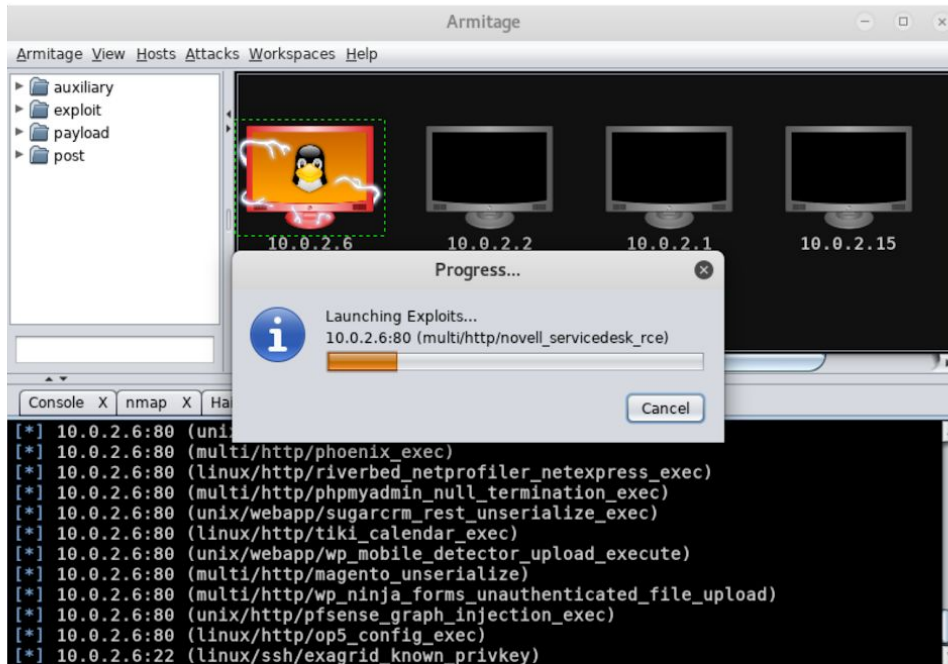


- Next is to go to the menu “**hosts**”, then “**nmap scan**”, then “**quick scan**” and set the IP range of which metasploitable belongs to. In this case, **10.0.2.1/24**. Nmap is used to find our victims.



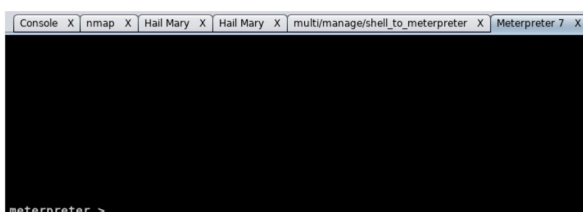
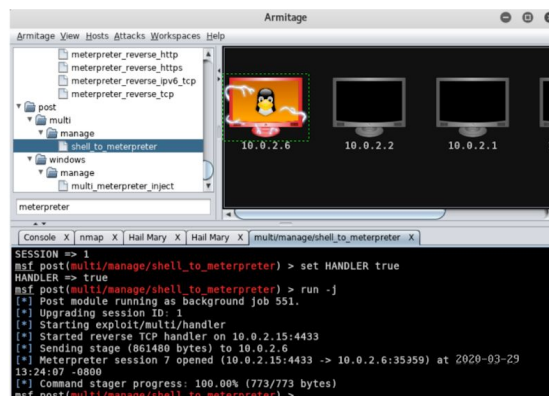
This scan succeeded in finding 4 machines and one of them being the one with open ports. That victim's IP address is 10.0.2.6

- Next is to go **“attack”**, go to **“hail mary”**. This will highlight out our victim with the open ports in red, **10.0.2.6**



After launching the **“Hail Mary”** attack, the victim’s machine is flooded by brute force attacks. This not a very stealthy approach because we are sending many exploits towards the victim’s machine. Hence the name. In the case that we were to fail with our hail mary, we will have to keep running it until we find active sessions. Eventually, I was able to find 3 sessions. Once found we will use Meterpreter.

- Next is to search for **“meterpreter”** in the left column, and choose **“post-multi-manage-shell_to_meterpreter”**, and run it.

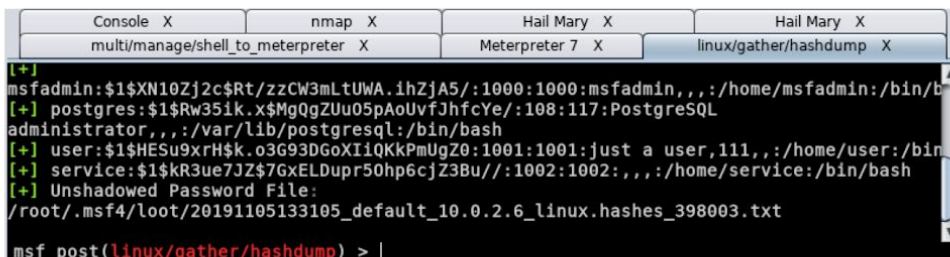


This resulted in 7 that is opened. Next, we right-click the hacked victim machine then go to **meterpreter 7**, interact meterpreter shell. If we succeed we can run root administrative commands. If not, we will choose another and try again.

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

Because we were successful we are able to run commands like “**shell**” and what this does is gives us an interactive OS shell. Also, I ran “**whoami**” which you can see returned root meaning we have gained privilege inside the victim’s machine.

Next, we can locate **hashdump** by going to **linux/gather/hashdump**. Below you can see that it is working and we are now viewing the hashdumps of the passwords on the victim’s machine.



```
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.1hZjA5/:1000:1000:msfadmin,,:/home/msfadmin:/bin/b
[+] postgres:$1$Rw351k.x$MgQgZUu05pAoUvfJhfcYe/:108:117:PostgreSQL
administrator,,:/var/lib/postgresql:/bin/bash
[+] user:$1$HESu9xrH$k.o3G93DGoXI1QKkPmUgZ0:1001:1001:just a user,111,,:/home/user:/bin/b
[+] service:$1$KR3ue7JZ$7GxELDupr50hp6cjZ3Bu//:1002:1002,,:/home/service:/bin/bash
[+] Unshadowed Password File:
/root/.msf4/loot/20191105133105_default_10.0.2.6_linux.hashes_398003.txt
msf post(linux/gather/hashdump) > |
```

Also, “**ps**” lets us see which processes are currently running on the victim’s machine.

Next is to run some meterpreter commands from the cheat sheet.

```
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 khal-1---
```