

# CSE 3501

## INFORMATION SECURITY ANALYSIS & AUDIT



### Lab Assessment – 3

L9+L10 | PLBG04

FALL SEMESTER 2021-22

by

**SHARADINDU ADHIKARI**

19BCE2105

## Experiment 1: Firewall

1. **Aim:** To demonstrate a Firewall implementation in Cisco Packet Tracer®.
2. **Description:** A firewall is a network security device that monitors incoming and outgoing network traffic and decides whether to allow or block specific traffic based on a defined set of security rules.

Firewalls have been a first line of defence in network security for over 25 years. They establish a barrier between secured and controlled internal networks that can be trusted and untrusted outside networks, such as the Internet.

3. **Benefits of Firewall:**

- Monitors Network Traffic. All of the benefits of firewall security start with the ability to monitor network traffic.
- Stops Virus Attacks. Nothing can shut your digital operations down faster and harder than a virus attack.
- Prevents Hacking.
- Stops Spyware.
- Promotes Privacy.

4. **Procedure:**

**Device Configuration:**

- select 3 PCs, 1 Hub-PT, 1 Server-PT, & a copper straight through connection cable for the connection's b/w PCs, Hub and Server.
- Connect fastethernet0 port of server with fastethernet0 port of hub.
- Connect fastethernet0 port of PC0 with fastethernet1 port of hub.
- Connect fastethernet0 port of PC1 with fastethernet2 port of hub.
- Connect fastethernet0 port of PC2 with fastethernet3 port of hub.

**Server Configuration:**

- Go to router and then config.
- Select FastEthernet0/0 under interface tab.
- Set IP address and subnet mask for the selected router (i.e., 20.0.0.1).
- Go to services then select HTTP. Turn on both radio buttons.
- Now select DHCP, turn on service beside fastethernet0.
- Save all the changes.

**PC Configuration:**

- Go to Desktop, then to: IP configuration on each PC.
- Select Fastethernet0 from interface dropdown menu.
- Select the DHCP radio button. (It will prompt DHCP request successful).

**Firewall Configuration (for the Server):**

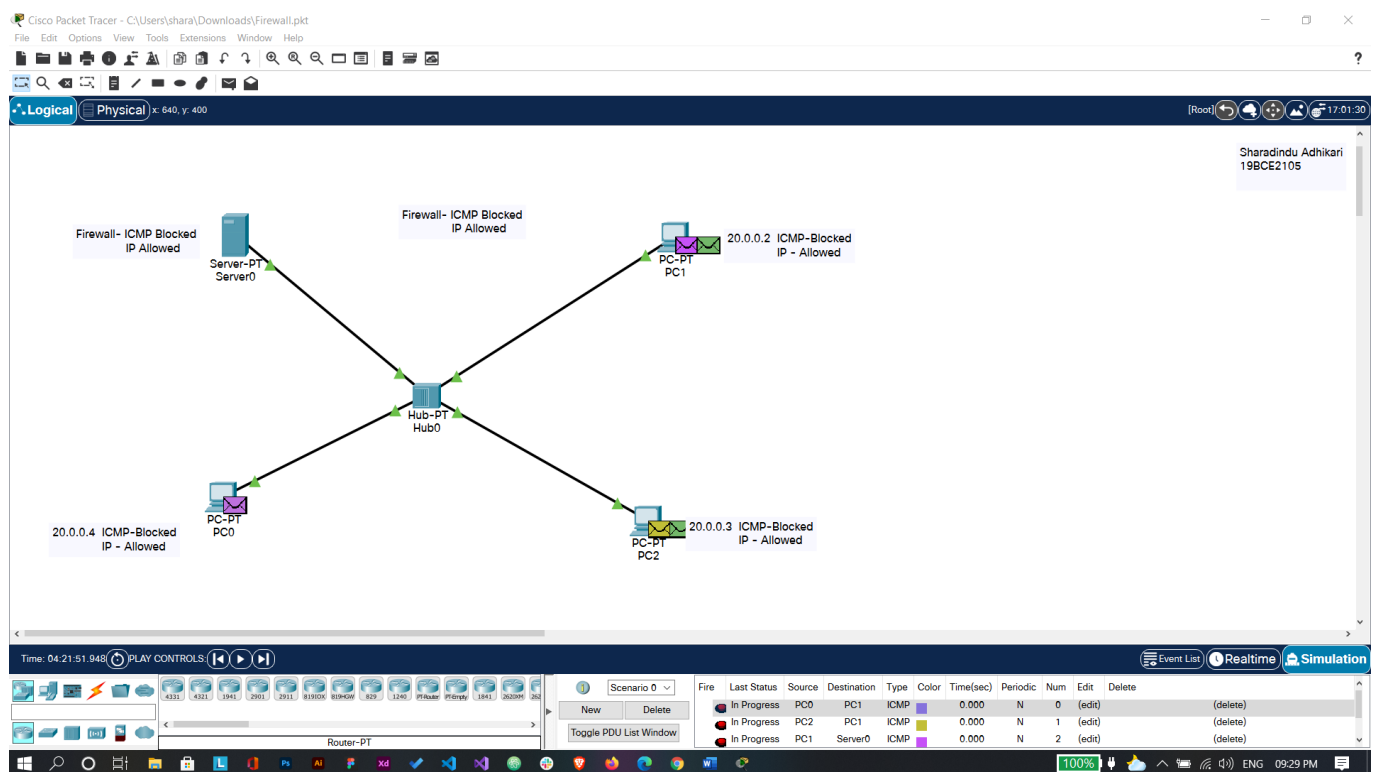
- Go to desktop and select IPV4 firewall.
- Turn the service radio button on.
- Select Action->Deny and Protocol->ICMP.
- Set Remote-IP to 0.0.0.0 and Remote Wildcard Mask to 255.255.255.255.
- Click on Add to append the changes and then Save.
- Repeat the above 2 steps for Action->Allow and Protocol->IP.

## 5. Testing:

- For testing ICMP protocol go to Desktop->Command Prompt for any PC and enter the ping command followed by the IP address of the Server to analyse the working of the firewall.
- For testing IP protocol go to Desktop->Web Browser and enter the IP address of the Server, if it leads to the CISCO Homepage then firewall is set-up successfully.

## 6. Screenshots:

### Device configuration:



### Server configuration:

Server0 Configuration Window - IP Configuration Tab

**IP Configuration**

☐ DHCP ☒ Static

IPv4 Address: 20.0.0.1

Subnet Mask: 255.0.0.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

**IPv6 Configuration**

☐ Automatic ☒ Static

IPv6 Address: /

Link Local Address: FE80::2D0:97FF:FE00:6D19

Default Gateway:

DNS Server:

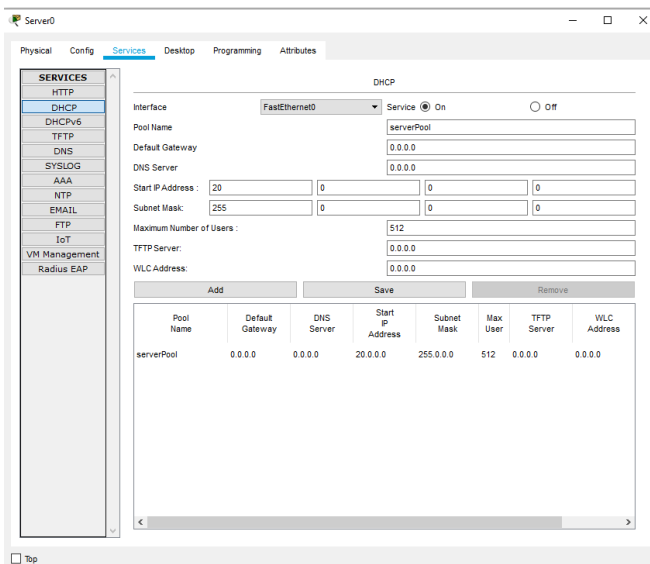
**802.1X**

☐ Use 802.1X Security

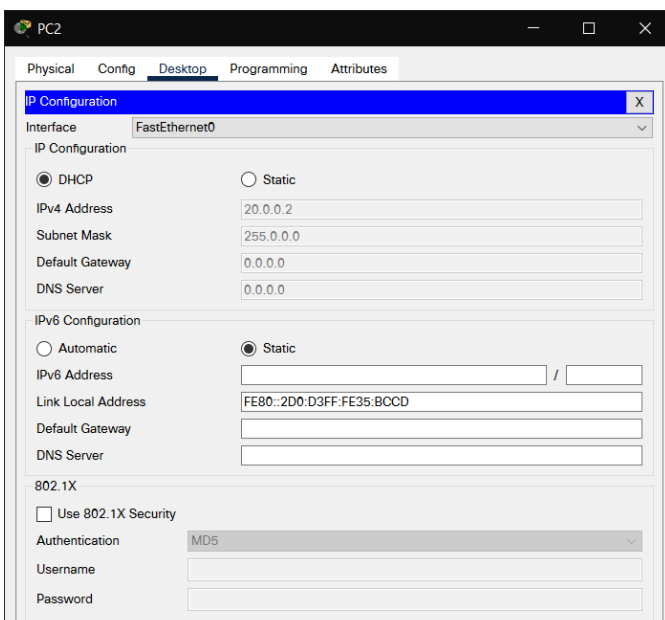
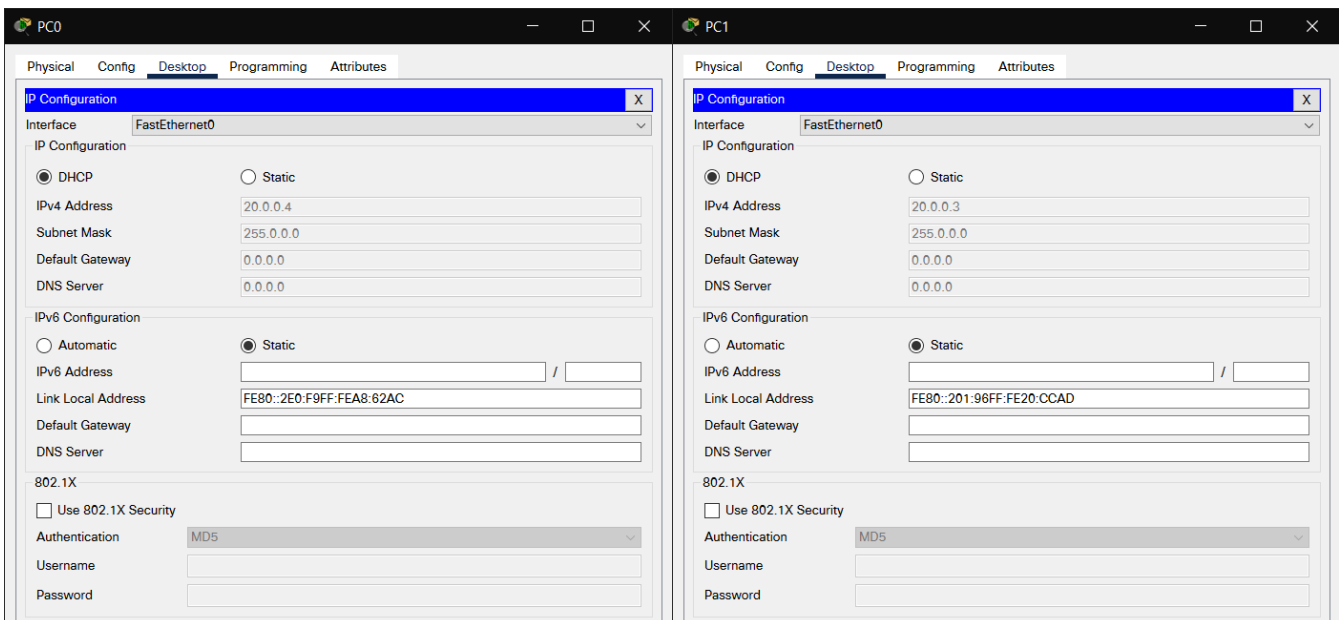
Authentication: MD5

Username:

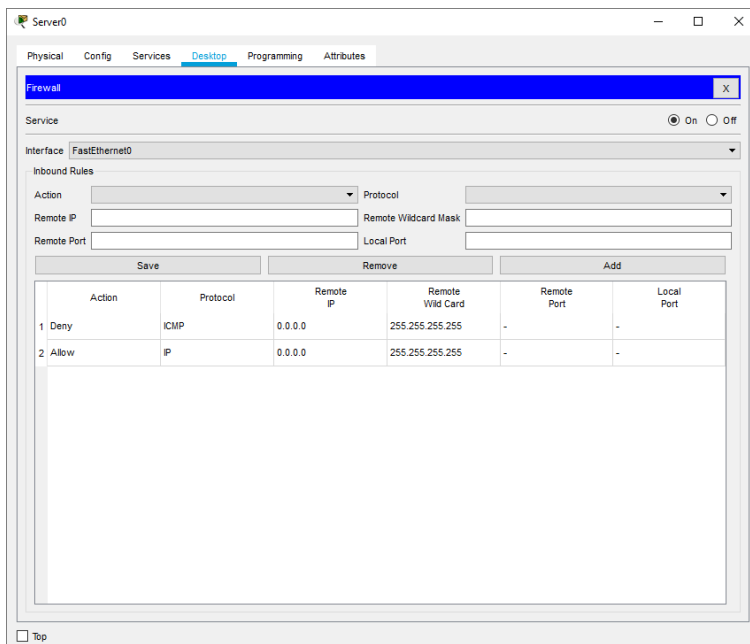
Password:



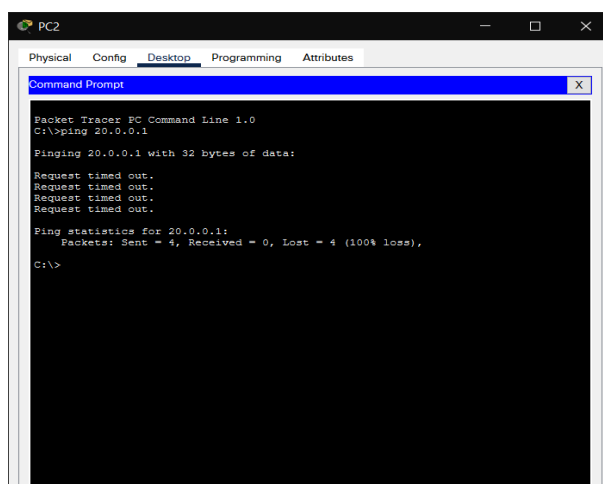
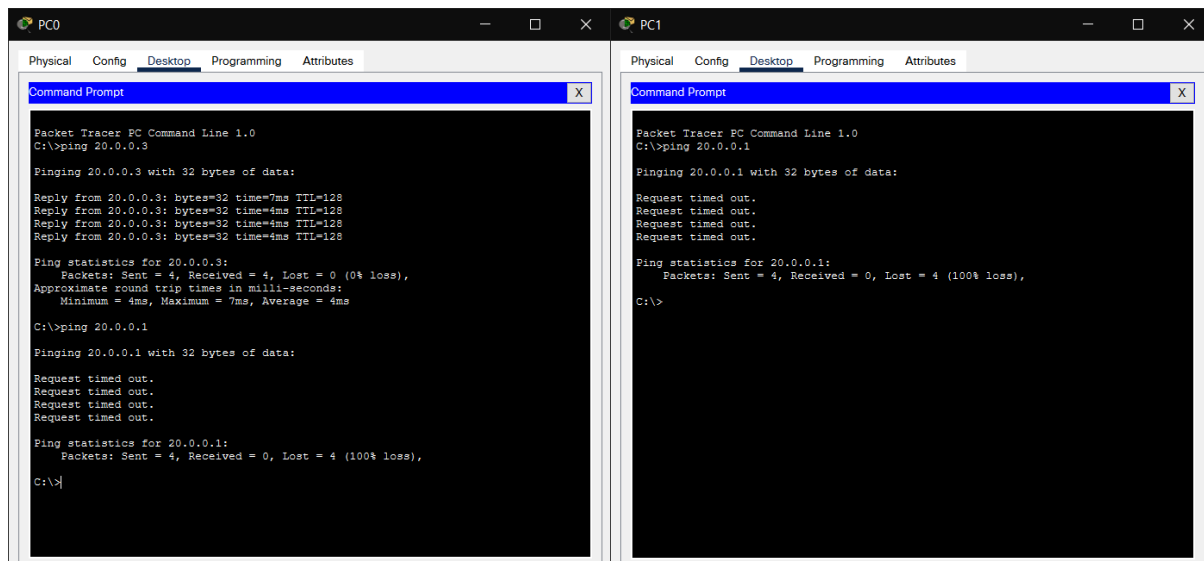
### PC configurations:



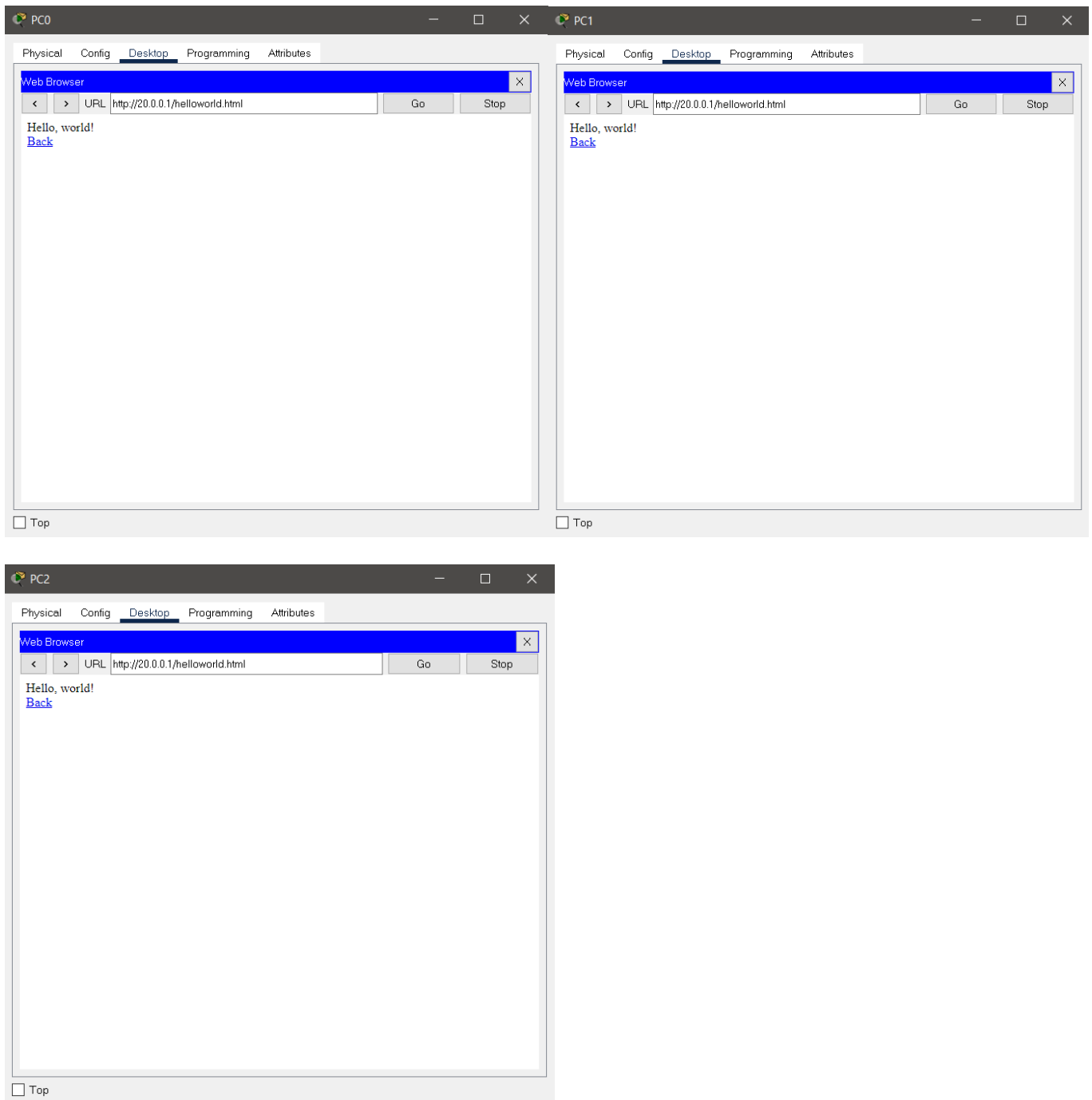
## Firewall configuration:



## Testing ICMP Protocol:



## Testing IP Protocol:



7. **Observation:** It is clear that some form of security for private networks connected to the internet is essential. A firewall is an important and necessary part of that security, but cannot be expected to perform all the required security functions. That being said, our firewall implemented in this system is working well, as has been demonstrated.

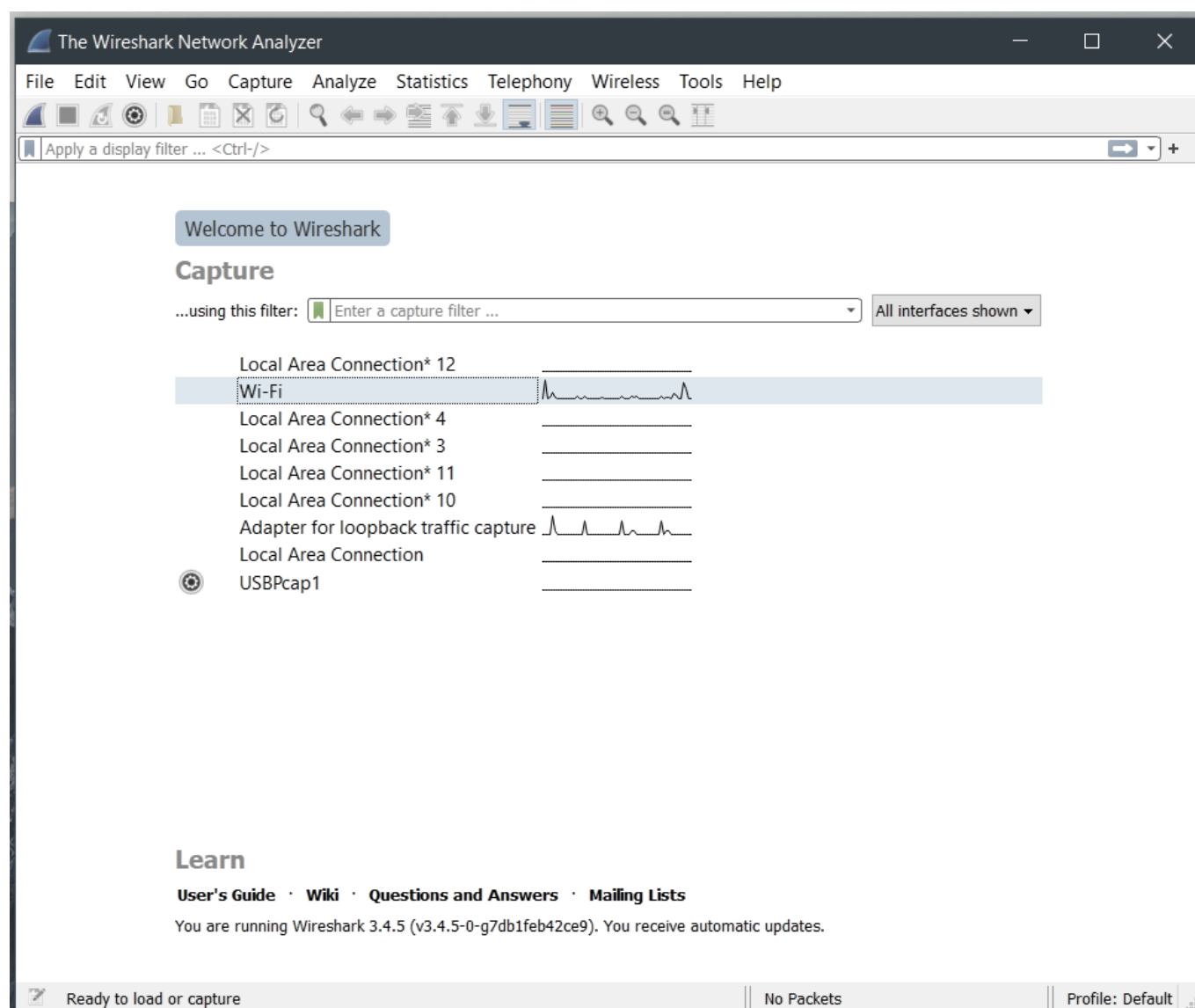
## Experiment 2: Wireshark Filters

**Aim:** To analyse the Wireshark capture filter.

**Introduction:** Wireshark is a network packet analyzer. A network packet analyzer presents captured packet data in as much detail as possible.

We could think of a network packet analyzer as a measuring device for examining what's happening inside a network cable, just like an electrician uses a voltmeter for examining what's happening inside an electric cable (but at a higher level, of course).

In the past, such tools were either very expensive, proprietary, or both. However, with the advent of Wireshark, that has changed. Wireshark is available for free, is open source, and is one of the best packet analyzers available today.



Once we have completed the intro steps and finished the capture process, the Wireshark main window should be alive with data. As a matter of fact, we might get overwhelmed by the amount of data that

appears, but it will all start to make sense very quickly as we break down the main window of Wireshark – one piece at a time.

## Capturing from WiFi:

Wireshark interface showing a packet capture from Wi-Fi. The packet list shows a series of IGMPv2 and TCP segments. The packet details pane shows the structure of a TCP segment. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.7	5.9.70.141	TCP	55	56781 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1 [TCP segment of a reassembled PDU]
2	0.221417	5.9.70.141	192.168.1.7	TCP	66	443 → 56781 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
3	1.001417	192.168.1.1	224.0.0.1	IGMPv2	46	Membership Query, specific for group 239.255.102.18
4	1.149751	192.168.1.7	239.255.102.18	IGMPv2	46	Membership Report group 239.255.102.18
5	3.241236	192.168.1.1	224.0.0.1	IGMPv2	46	Membership Query, general
6	3.635870	192.168.1.7	224.0.0.252	IGMPv2	46	Membership Report group 224.0.0.252
7	3.635965	192.168.1.7	239.255.255.250	IGMPv2	46	Membership Report group 239.255.255.250
8	3.635989	192.168.1.7	224.0.0.251	IGMPv2	46	Membership Report group 224.0.0.251
9	3.636009	192.168.1.7	239.255.102.18	IGMPv2	46	Membership Report group 239.255.102.18
10	4.011018	192.168.1.1	224.0.0.1	IGMPv2	46	Membership Query, specific for group 239.255.102.18
11	4.137302	192.168.1.7	239.255.102.18	IGMPv2	46	Membership Report group 239.255.102.18
12	5.230058	192.168.1.7	5.9.70.141	TCP	55	[TCP Keep-Alive] 56781 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1
13	5.451955	5.9.70.141	192.168.1.7	TCP	66	[TCP Keep-Alive ACK] 443 → 56781 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
14	5.638472	192.168.1.7	117.219.230.204	QUIC	1392	Initial, DCID=2c4c8281c0539ffc, PKN: 1, CRYPTO, PADDING
15	5.638650	192.168.1.7	117.219.230.204	QUIC	117	0-RTT, DCID=2c4c8281c0539ffc
16	5.638790	192.168.1.7	117.219.230.204	QUIC	1059	0-RTT, DCID=2c4c8281c0539ffc
17	5.648062	117.219.230.204	192.168.1.7	QUIC	1392	Protected Payload (KP0)
18	5.648062	117.219.230.204	192.168.1.7	QUIC	572	Protected Payload (KP0)
19	5.648648	192.168.1.7	117.219.230.204	QUIC	120	Handshake, DCID=2c4c8281c0539ffc
20	5.649102	192.168.1.7	117.219.230.204	QUIC	75	Protected Payload (KP0), DCID=2c4c8281c0539ffc
21	5.649814	117.219.230.204	192.168.1.7	QUIC	69	Protected Payload (KP0)
22	5.651259	117.219.230.204	192.168.1.7	QUIC	719	Protected Payload (KP0)
23	5.651259	117.219.230.204	192.168.1.7	QUIC	1388	Protected Payload (KP0)
24	5.651259	117.219.230.204	192.168.1.7	QUIC	1392	Protected Payload (KP0)
25	5.651259	117.219.230.204	192.168.1.7	QUIC	1392	Protected Payload (KP0)

Frame 1: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface \Device\NPF\_{A241E987-CAAE-4BCF-8F47-95855FC2CCE5}, id 0  
 Ethernet II, Src: IntelCor\_a5:0d:16 (0c:dd:24:a5:0d:16), Dst: Syrotech\_0a:65:fb (7c:a9:6b:0a:65:fb)  
 Internet Protocol Version 4, Src: 192.168.1.7, Dst: 5.9.70.141  
 Transmission Control Protocol, Src Port: 56781, Dst Port: 443, Seq: 1, Ack: 1, Len: 1

0000 7c a9 6b 0a 65 fb 0c dd 24 a5 0d 16 08 00 45 00 |.k.e...\$....E-  
 0010 00 29 de 5e 40 00 80 06 00 00 c0 a8 01 07 05 09 |..@.....  
 0020 46 8d dd cd 01 bb 3e c0 9b cb 44 c0 87 56 50 10 |F.....D..VP-  
 0030 02 03 0d 61 00 00 00 |...a....

## Filter TCP:

Wireshark interface showing a packet capture filtered for TCP. The packet list shows a series of TCP segments. The packet details pane shows the structure of a TCP segment. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	52.114.133.28	192.168.1.7	TCP	56	443 → 56216 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1420 WS=256 SACK_PERM=1
2	0.000125	192.168.1.7	52.114.133.28	TCP	54	56216 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=0
3	0.000342	192.168.1.7	52.114.133.28	TLSv1.2	571	Client Hello
4	0.001956	52.114.133.28	192.168.1.7	TCP	1474	443 → 50585 [ACK] Seq=1 Ack=1 Win=2050 Len=1420 [TCP segment of a reassembled PDU]
5	0.001956	52.114.133.28	192.168.1.7	TCP	1474	443 → 50585 [ACK] Seq=1421 Ack=1 Win=2050 Len=1420 [TCP segment of a reassembled PDU]
6	0.001956	52.114.133.28	192.168.1.7	TCP	1474	443 → 50585 [ACK] Seq=2841 Ack=1 Win=2050 Len=1420 [TCP segment of a reassembled PDU]
7	0.001956	52.114.133.28	192.168.1.7	TCP	1474	443 → 50585 [ACK] Seq=4261 Ack=1 Win=2050 Len=1420 [TCP segment of a reassembled PDU]
8	0.001956	52.114.133.28	192.168.1.7	TLSv1.2	234	Server Hello, Certificate, Certificate Status, Server Key Exchange, Server Hello Done
9	0.002003	192.168.1.7	52.114.133.28	TCP	54	50585 → 443 [ACK] Seq=1 Ack=5861 Win=515 Len=0
10	0.005665	192.168.1.7	52.114.133.28	TLSv1.2	212	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
11	0.005856	192.168.1.7	52.114.133.28	TLSv1.2	153	Application Data
12	0.006040	192.168.1.7	52.114.133.28	TLSv1.2	226	Application Data
13	0.006075	192.168.1.7	52.114.133.28	TLSv1.2	373	Application Data
14	0.102923	52.113.194.132	192.168.1.7	TLSv1.2	389	Application Data
15	0.149814	192.168.1.7	52.113.194.132	TCP	54	49613 → 443 [ACK] Seq=1 Ack=336 Win=512 Len=0
16	0.233055	52.114.133.28	192.168.1.7	TCP	1474	443 → 56216 [ACK] Seq=1 Ack=518 Win=524800 Len=1420 [TCP segment of a reassembled PDU]
17	0.233055	52.114.133.28	192.168.1.7	TCP	1474	443 → 56216 [ACK] Seq=1421 Ack=518 Win=524800 Len=1420 [TCP segment of a reassembled PDU]
18	0.233055	52.114.133.28	192.168.1.7	TCP	1474	443 → 56216 [ACK] Seq=2841 Ack=518 Win=524800 Len=1420 [TCP segment of a reassembled PDU]
19	0.233055	52.114.133.28	192.168.1.7	TCP	1474	443 → 56216 [ACK] Seq=4261 Ack=518 Win=524800 Len=1420 [TCP segment of a reassembled PDU]
20	0.233055	52.114.133.28	192.168.1.7	TCP	1474	443 → 56216 [ACK] Seq=5861 Ack=518 Win=524800 Len=1420 [TCP segment of a reassembled PDU]
21	0.233055	52.114.133.28	192.168.1.7	TLSv1.2	234	Server Hello, Certificate, Certificate Status, Server Key Exchange, Server Hello Done
22	0.233104	192.168.1.7	52.114.133.28	TCP	54	56216 → 443 [ACK] Seq=518 Ack=5861 Win=515 Len=0
23	0.234284	192.168.1.7	52.114.133.28	TLSv1.2	212	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
24	0.237984	52.114.133.28	192.168.1.7	TCP	54	443 → 50585 [ACK] Seq=5861 Ack=258 Win=2049 Len=0
25	0.239048	52.114.133.28	192.168.1.7	TCP	54	443 → 50585 [ACK] Seq=5861 Ack=2430 Win=2052 Len=0

Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF\_{A241E987-CAAE-4BCF-8F47-95855FC2CCE5}, id 0  
 Ethernet II, Src: Syrotech\_0a:65:fb (7c:a9:6b:0a:65:fb), Dst: IntelCor\_a5:0d:16 (0c:dd:24:a5:0d:16)  
 Internet Protocol Version 4, Src: 52.114.133.28, Dst: 192.168.1.7  
 Transmission Control Protocol, Src Port: 443, Dst Port: 56216, Seq: 0, Ack: 1, Len: 0

0000 0c dd 24 a5 0d 16 7c a9 6b 0a 65 fb 08 00 45 00 |..\$...|.k.e...E-  
 0010 00 34 82 ea 40 00 6b 06 11 9c 34 72 85 1c c0 a8 |..@.k...4r...  
 0020 01 07 01 bb db 9b 36 4c 52 62 a1 48 28 aa 80 12 |...@..6L.Rb.H(...  
 0030 ff ff c3 f5 00 00 02 04 05 bc 01 03 03 08 01 01 |.....  
 0040 04 02 |..



## Filter UDP:

The screenshot shows a Wireshark capture of network traffic. The filter bar at the top is set to 'udp'. The packet list pane displays a series of UDP packets, all originating from 192.168.1.7 and destined for 50004. The packet details pane for the selected packet (No. 243) shows the Ethernet II, Internet Protocol Version 4, and User Datagram Protocol layers. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
302	4.985387	213.179.210.229	192.168.1.7	UDP	248	50004 → 55515 Len=206
303	4.992276	213.179.210.229	192.168.1.7	UDP	250	50004 → 55515 Len=208
304	5.027855	213.179.210.229	192.168.1.7	UDP	254	50004 → 55515 Len=212
305	5.043107	213.179.210.229	192.168.1.7	UDP	257	50004 → 55515 Len=215
306	5.083247	213.179.210.229	192.168.1.7	UDP	258	50004 → 55515 Len=216
307	5.083247	213.179.210.229	192.168.1.7	UDP	258	50004 → 55515 Len=216
308	5.104800	213.179.210.229	192.168.1.7	UDP	247	50004 → 55515 Len=205
309	5.114914	213.179.210.229	192.168.1.7	UDP	248	50004 → 55515 Len=206
310	5.123179	213.179.210.229	192.168.1.7	UDP	245	50004 → 55515 Len=203
311	5.147215	213.179.210.229	192.168.1.7	UDP	247	50004 → 55515 Len=205
312	5.176655	213.179.210.229	192.168.1.7	UDP	243	50004 → 55515 Len=201
313	5.187755	213.179.210.229	192.168.1.7	UDP	242	50004 → 55515 Len=200
314	5.217555	213.179.210.229	192.168.1.7	UDP	239	50004 → 55515 Len=197
315	5.227649	213.179.210.229	192.168.1.7	UDP	234	50004 → 55515 Len=192
316	5.248861	213.179.210.229	192.168.1.7	UDP	242	50004 → 55515 Len=200
317	5.276592	213.179.210.229	192.168.1.7	UDP	235	50004 → 55515 Len=193
318	5.287471	213.179.210.229	192.168.1.7	UDP	243	50004 → 55515 Len=201
319	5.316566	213.179.210.229	192.168.1.7	UDP	239	50004 → 55515 Len=197
320	5.326490	213.179.210.229	192.168.1.7	UDP	242	50004 → 55515 Len=200
321	5.380807	213.179.210.229	192.168.1.7	UDP	249	50004 → 55515 Len=207
322	5.380807	213.179.210.229	192.168.1.7	UDP	256	50004 → 55515 Len=214
323	5.380807	213.179.210.229	192.168.1.7	UDP	251	50004 → 55515 Len=209
324	5.417737	213.179.210.229	192.168.1.7	UDP	240	50004 → 55515 Len=198
325	5.426405	213.179.210.229	192.168.1.7	UDP	232	50004 → 55515 Len=190

Frame 1: 243 bytes on wire (1944 bits), 243 bytes captured (1944 bits) on interface \Device\NPF\_{A241E987-CAAE-4BCF-8F47-95855FC2CE5}, id 0  
> Ethernet II, Src: IntelCor\_a5:0d:16 (0c:dd:24:a5:0d:16), Dst: Syrotech\_0a:65:fb (7c:a9:6b:0a:65:fb)  
> Internet Protocol Version 4, Src: 192.168.1.7, Dst: 213.179.210.229  
> User Datagram Protocol, Src Port: 60662, Dst Port: 50004  
> Data (201 bytes)

0000 7c a9 6b 0a 65 fb 0c dd 24 a5 0d 16 08 00 45 00 |.k.e...\$.....E  
0010 00 e5 ce cd 00 00 80 11 00 00 c0 a8 01 07 d5 b3 |.....  
0020 d2 e5 ec f6 c3 54 00 d1 6b 2b 90 78 08 91 12 2a |.....T...k+.x...\*  
0030 86 09 00 04 16 ef be de 00 01 c9 ae cc 08 e6 aa |.....  
0040 f9 2a ef 21 2d e3 84 ab 3e 79 4f 71 8b 7f 6b 01 |\*!.....>yOq.wk.  
0050 b0 40 69 70 60 5f d6 44 ea ad 3a a0 cb 95 85 57 |@lp...\_D.....W

## Filter TCP Port 80, UDP Port 80:

The screenshot shows a Wireshark capture of network traffic. The filter bar at the top is set to 'tcp.port == 80 || udp.port == 80'. The packet list pane displays a series of TCP and UDP packets, all originating from 192.168.1.7 and destined for 50004. The packet details pane for the selected packet (No. 3) shows the Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol layers. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
3	1.898459	192.168.1.7	5.9.70.141	TCP	55	56781 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1 [TCP segment of a reassembled PDU]
4	2.178489	5.9.70.141	192.168.1.7	TCP	66	443 → 56781 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
5	2.372483	192.168.1.7	13.126.138.201	TCP	54	59680 → 443 [FIN, ACK] Seq=1 Ack=1 Win=507 Len=0
6	2.401349	13.126.138.201	192.168.1.7	TLSv1.2	78	Application Data
7	2.401349	13.126.138.201	192.168.1.7	TCP	54	443 → 59680 [FIN, ACK] Seq=25 Ack=2 Win=8 Len=0
8	2.401349	192.168.1.7	13.126.138.201	TCP	54	59680 → 443 [RST, ACK] Seq=2 Ack=25 Win=0 Len=0
11	3.305299	192.168.1.7	3.108.107.38	TLSv1.2	108	Application Data
12	3.420805	3.108.107.38	192.168.1.7	TCP	54	443 → 53003 [ACK] Seq=1 Ack=55 Win=8 Len=0
13	3.420805	3.108.107.38	192.168.1.7	TLSv1.2	110	Application Data
14	3.464224	192.168.1.7	3.108.107.38	TCP	54	53003 → 443 [ACK] Seq=55 Ack=57 Win=511 Len=0
20	7.193294	192.168.1.7	5.9.70.141	TCP	55	[TCP Keep-Alive] 56781 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1
21	7.440137	5.9.70.141	192.168.1.7	TCP	66	[TCP Keep-Alive] 443 → 56781 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
27	10.798600	192.168.1.7	142.250.194.206	TCP	66	59681 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
28	10.847125	142.250.194.206	192.168.1.7	TCP	66	443 → 59681 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1420 SACK_PERM=1 WS=256
29	10.847176	192.168.1.7	142.250.194.206	TCP	54	59681 → 443 [ACK] Seq=1 Ack=1 Win=131840 Len=0
30	10.847177	192.168.1.7	142.250.194.206	TLSv1	408	Client Hello
31	10.896301	142.250.194.206	192.168.1.7	TCP	54	443 → 59681 [ACK] Seq=1 Ack=355 Win=66816 Len=0
32	10.966219	142.250.194.206	192.168.1.7	TLSv1	177	Server Hello, Change Cipher Spec, Encrypted Handshake Message
33	10.967424	192.168.1.7	142.250.194.206	TLSv1	363	Change Cipher Spec, Encrypted Handshake Message, Application Data, Application Data
34	11.016309	142.250.194.206	192.168.1.7	TCP	54	443 → 59681 [ACK] Seq=124 Ack=664 Win=67840 Len=0
35	11.332768	192.168.1.7	142.250.194.206	TLSv1	368	Application Data, Application Data
36	11.386555	142.250.194.206	192.168.1.7	TCP	54	443 → 59681 [ACK] Seq=124 Ack=978 Win=68864 Len=0
37	11.451678	142.250.194.206	192.168.1.7	TLSv1	699	Application Data
38	11.509101	192.168.1.7	142.250.194.206	TCP	54	59681 → 443 [ACK] Seq=978 Ack=769 Win=131072 Len=0

Frame 3: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface \Device\NPF\_{A241E987-CAAE-4BCF-8F47-95855FC2CE5}, id 0  
> Ethernet II, Src: IntelCor\_a5:0d:16 (0c:dd:24:a5:0d:16), Dst: Syrotech\_0a:65:fb (7c:a9:6b:0a:65:fb)  
> Internet Protocol Version 4, Src: 192.168.1.7, Dst: 5.9.70.141  
> Transmission Control Protocol, Src Port: 56781, Dst Port: 443, Seq: 1, Ack: 1, Len: 1

0000 7c a9 6b 0a 65 fb 0c dd 24 a5 0d 16 08 00 45 00 |.k.e...\$.....E  
0010 00 29 df 27 40 00 80 06 00 00 c0 a8 01 07 05 09 |.)'.@.....  
0020 46 bd dd cd 01 bb 3e c0 9b cb 44 c0 87 56 50 10 |F.....D..VP.  
0030 02 03 00 61 00 00 00 |....a...

## IP address 192.168.1.7

(Displays all traffic for the entered subnet, this will match on source or destination)

The image shows a Wireshark capture of network traffic on the interface \Device\NPF\_{A241E987-CAAE-4BCF-8F47-95855FC2CCE5}, id 0. The capture is filtered for IP address 192.0.2.1. The packet list shows various DNS and TCP packets. The packet details pane shows the structure of a TCP segment (Frame 1276) with a length of 571 bytes. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
1252	10.236330	192.168.1.7	218.248.114.1	DNS	77	Standard query 0xc5e6 A www.wikipedia.org
1253	10.245914	204.79.197.200	192.168.1.7	TLSv1.2	176	Application Data
1254	10.259200	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=24707 Win=525312 Len=0
1255	10.259200	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=27547 Win=525312 Len=0
1256	10.259253	192.168.1.7	52.182.141.63	TLSv1.2	7933	Application Data, Application Data
1257	10.262784	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=31807 Win=525312 Len=0
1258	10.262784	218.248.114.1	192.168.1.7	DNS	122	Standard query response 0xc5e6 A www.wikipedia.org CNAME dyna.wikimedia.org A 103.102.166.224
1259	10.262784	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=33227 Win=525312 Len=0
1260	10.262784	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=36067 Win=525312 Len=0
1261	10.262784	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=38907 Win=525312 Len=0
1262	10.263320	192.168.1.7	103.102.166.224	TCP	66	49460 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
1263	10.263730	192.168.1.7	103.102.166.224	TCP	66	52636 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
1264	10.263933	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=41747 Win=525312 Len=0
1265	10.264505	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=44587 Win=525312 Len=0
1266	10.295034	192.168.1.7	204.79.197.200	TCP	54	54218 → 443 [ACK] Seq=226395 Ack=285840 Win=262912 Len=0
1267	10.503380	192.168.1.7	103.102.166.224	TCP	66	51839 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
1268	10.516036	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=48847 Win=525312 Len=0
1269	10.516785	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=50267 Win=525312 Len=0
1270	10.516785	52.182.141.63	192.168.1.7	TCP	54	443 → 50848 [ACK] Seq=6227 Ack=52466 Win=525312 Len=0
1271	10.520632	52.182.141.63	192.168.1.7	TLSv1.2	909	Application Data
1272	10.521148	192.168.1.7	52.182.141.63	TCP	54	50848 → 443 [FIN, ACK] Seq=52466 Ack=7082 Win=131072 Len=0
1273	10.540843	192.168.1.7	218.248.114.1	DNS	91	Standard query 0x14d6 A browser.pipe.aria.microsoft.com
1274	10.550333	103.102.166.224	192.168.1.7	TCP	66	443 → 49460 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1420 SACK_PERM=1 WS=512
1275	10.550401	192.168.1.7	103.102.166.224	TCP	54	49460 → 443 [ACK] Seq=1 Ack=1 Win=131840 Len=0

Frame 1276: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits) on interface \Device\NPF\_{A241E987-CAAE-4BCF-8F47-95855FC2CCE5}, id 0  
 Ethernet II, Src: IntelCor\_a5:0d:16 (0c:dd:24:a5:0d:16), Dst: Syrotech\_0a:65:fb (7c:a9:6b:0a:65:fb)  
 Internet Protocol Version 4, Src: 192.168.1.7, Dst: 103.102.166.224  
 Transmission Control Protocol, Src Port: 49460, Dst Port: 443, Seq: 1, Ack: 1, Len: 517  
 Transport Layer Security

0000 7c a9 6b 0a 65 fb 0c dd 24 a5 0d 16 08 00 45 00 |.k.e...\$....E.  
 0010 02 2d fe 13 40 00 80 06 00 00 c0 a8 01 07 67 66 |...@.....gf  
 0020 a6 e0 c1 34 01 bb ae 7e 13 1b 52 61 75 3d 50 18 |...4....Rau=P  
 0030 02 03 d2 15 00 00 16 03 01 02 00 01 00 01 fc 03 |.....  
 0040 03 4b d7 cb aa 0b c2 7c cf 0f 70 91 df b9 d7 b8 |.K..k|..p....  
 0050 8d 09 74 39 7b ba 02 1a 08 26 87 0f 53 78 c4 d5 |..t9{...&..Sx..

Internet Protocol Version 4: Protocol Packets: 1717 - Displayed: 1705 (99.3%) - Dropped: 0 (0.0%) Profile: Default

**Conclusion:** By usage of Wireshark, we can track and see all the possible packets, and the main focus of this tool is observing the data traffic within a network, as has been demonstrated here. Such a tool thus allows the user to examine his/her own computer for protocol errors and problems within the network architecture.