

NAME: PRERNA SUNIL JADHAV

SAP ID: 60004220127

BATCH: COMPUTER ENGINEERING

COURSE: INFORMATION SECURITY LABORATORY

COURSE CODE: DT19CEL603

### EXPERIMENT 10

AIM: Perform Packet capture & sniffing IP traffic using Wireshark.

THEORY: Packet sniffers intercept packets of data flowing across a computer network in order to view their content in. This act is called packet sniffing.

Webpages and emails are not sent through the Internet as one document rather, the sending side breaks them down into many little data packets. These packets are then addressed to an IP address at the receiving end, which has to send back an acknowledgment of each packet it receives.

These packets are not transferred from the sender to the receiver through a single or direct connection. Instead as each packet traverses, the internet enroute to its destination, it passes through a no. of traffic control devices such as routers and switches. Each time a



a packet passes through one of these traffic control devices, it is susceptible to capture & analysis. Wireshark snort sniff are examples of packet sniffing tools.

CONCLUSION: Thus, we have performed packet capture & sniffed IP traffic using wireshark



Academic Year: 2022-2023

|                 |                                      |
|-----------------|--------------------------------------|
| Name:           | Prerna Sunil Jadhav                  |
| Sap Id:         | 60004220127                          |
| Class:          | T. Y. B. Tech (Computer Engineering) |
| Course:         | Information Security Laboratory      |
| Course Code:    | DJ19CEL603                           |
| Experiment No.: | 10                                   |

**AIM:** Perform Packet Capture and Sniff IP traffic using Wireshark.

### Capturing ICMP Packets:

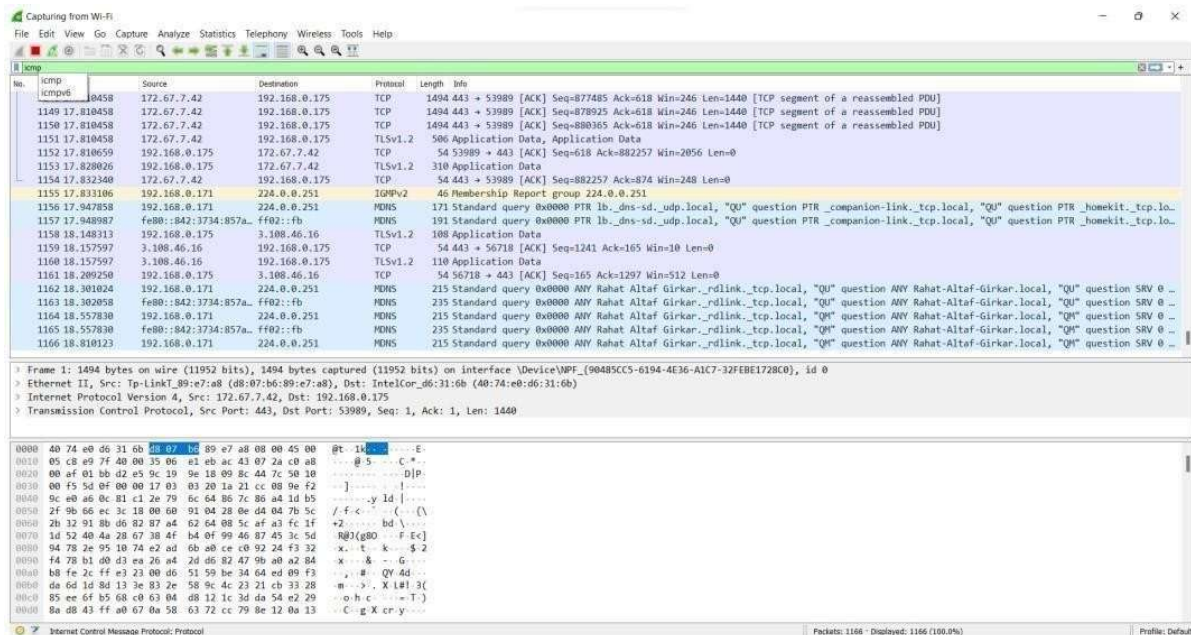
C:\Users\Marwin Shroff>ping 8.8.8.8 Pinging 8.8.8.8 with 32 bytes of data: Reply from 8.8.8.8:  
bytes=32 time=5ms TTL=119

Reply from 8.8.8.8: bytes=32 time=6ms TTL=119 Reply from 8.8.8.8: bytes=32 time=2ms TTL=119

Reply from 8.8.8.8: bytes=32 time=3ms TTL=119 Ping statistics for 8.8.8.8:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds:

Minimum = 2ms, Maximum = 6ms, Average = 4ms







### Capturing TCP Packets:

Wireshark packet capture showing TCP segments. The packet list shows multiple TCP segments from 192.168.0.175 to 192.168.0.175. The packet details show the selected packet (No. 23818) with Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol fields. The packet bytes show the raw data in hexadecimal and ASCII.

### Capturing FTP Packets:

```
C:\Users\Marwin Shroff>ftp ftp.cdc.gov Connected to ftp.cdc.gov. 220 Microsoft FTP Service
200 OPTS UTF8 command successful - UTF8 encoding now ON. User (ftp.cdc.gov:(none)):
anonymous
331 Anonymous access allowed, send identity (e-mail name) as password. Password: 230 User
logged in.
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection.
.change.dir .message pub Readme
Siteinfo w3c welcome.msg 226 Transfer complete. ftp: 67 bytes received in 0.03Seconds
2.03Kbytes/sec.
```



Academic Year: 2022-2023

Wireshark packet capture showing ARP request and response. The capture is on interface \Device\NPF\_{90485CC5-6194-4E36-A1C7-32FEBE1728C0}, id 0. The packet list shows a sequence of packets, with the selected packet being an ARP request from 192.168.0.175 to 192.168.0.175. The packet details pane 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.

## Capturing ARP Packets:

Wireshark packet capture showing ARP request and response. The capture is on interface \Device\NPF\_{90485CC5-6194-4E36-A1C7-32FEBE1728C0}, id 0. The packet list shows a sequence of packets, with the selected packet being an ARP request from 192.168.0.175 to 192.168.0.175. The packet details pane 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.



Academic Year: 2022-2023

B) Tracing Packets based on filters: 1] Filter Results by Port:  
Traces all packets related to Port 80.

Wireshark packet capture showing filter 'top port == 80' applied. The packet list shows various TCP and HTTP packets. The packet details pane shows the selected packet (No. 10205) with details for Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol. The packet bytes pane shows the raw data in hexadecimal and ASCII.

2] Filter by Delta Time :  
Displays tcp packets with delta time of greater than 0.500 sec

Wireshark packet capture showing filter 'top.time\_delta > 0.500' applied. The packet list shows various TCP and Application Data packets. The packet details pane shows the selected packet (No. 82035) with details for Ethernet II, Internet Protocol Version 4, and Transport Layer Security. The packet bytes pane shows the raw data in hexadecimal and ASCII.

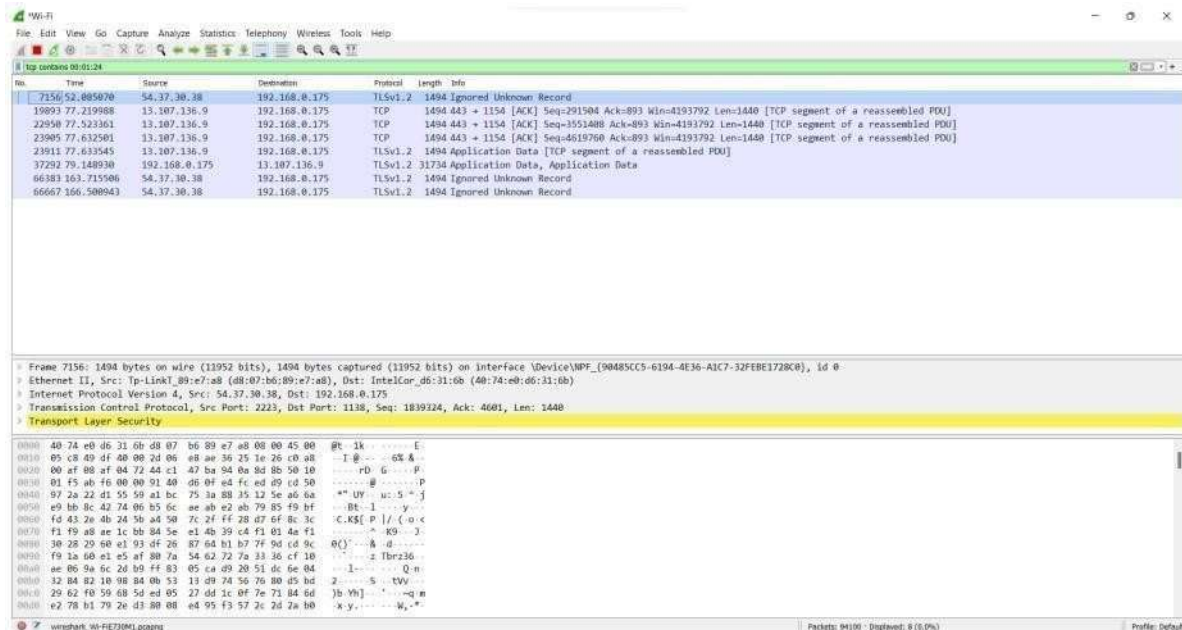




Academic Year: 2022-2023

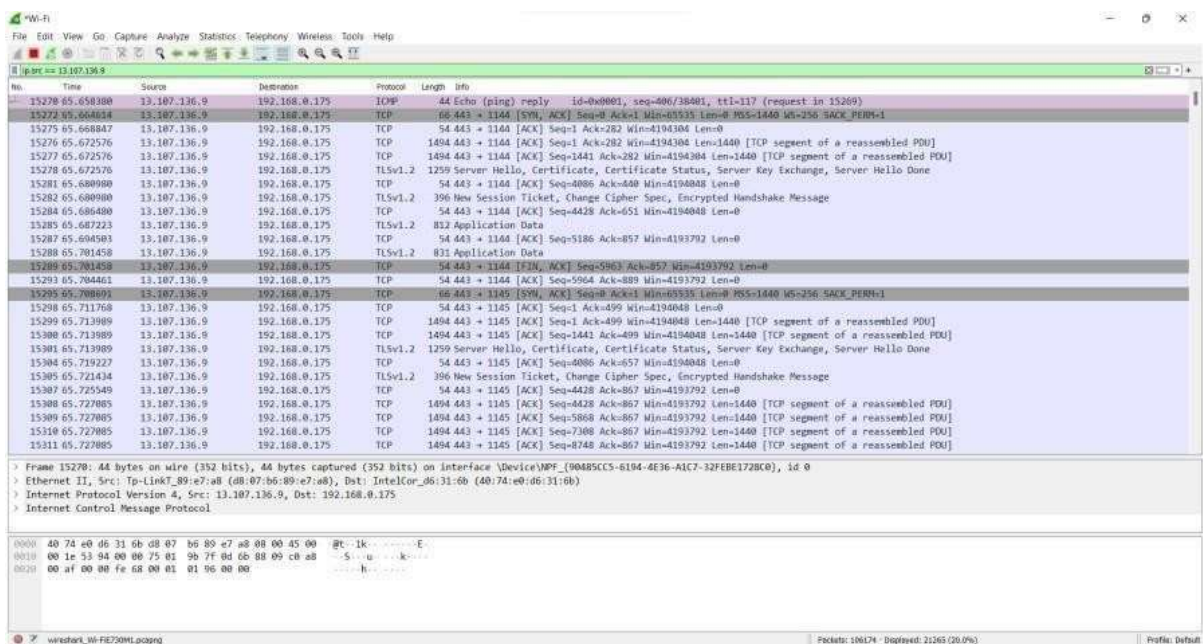
### 3] Filter by Byte Sequence:

Displays packets which contain a particular byte sequence.



### 4] Filter by Source IP Address:

Displays packets which have source IP address same as the one provided in the argument.



**CONCLUSION:** Thus, we have successfully studied packet sniffing tools (Wireshark) and explored how packets can be traced based on different filters