Experiment No. 9

Aim

Analyzing network packet stream using tcpdump and wireshark. Perform basic service tests using nc.

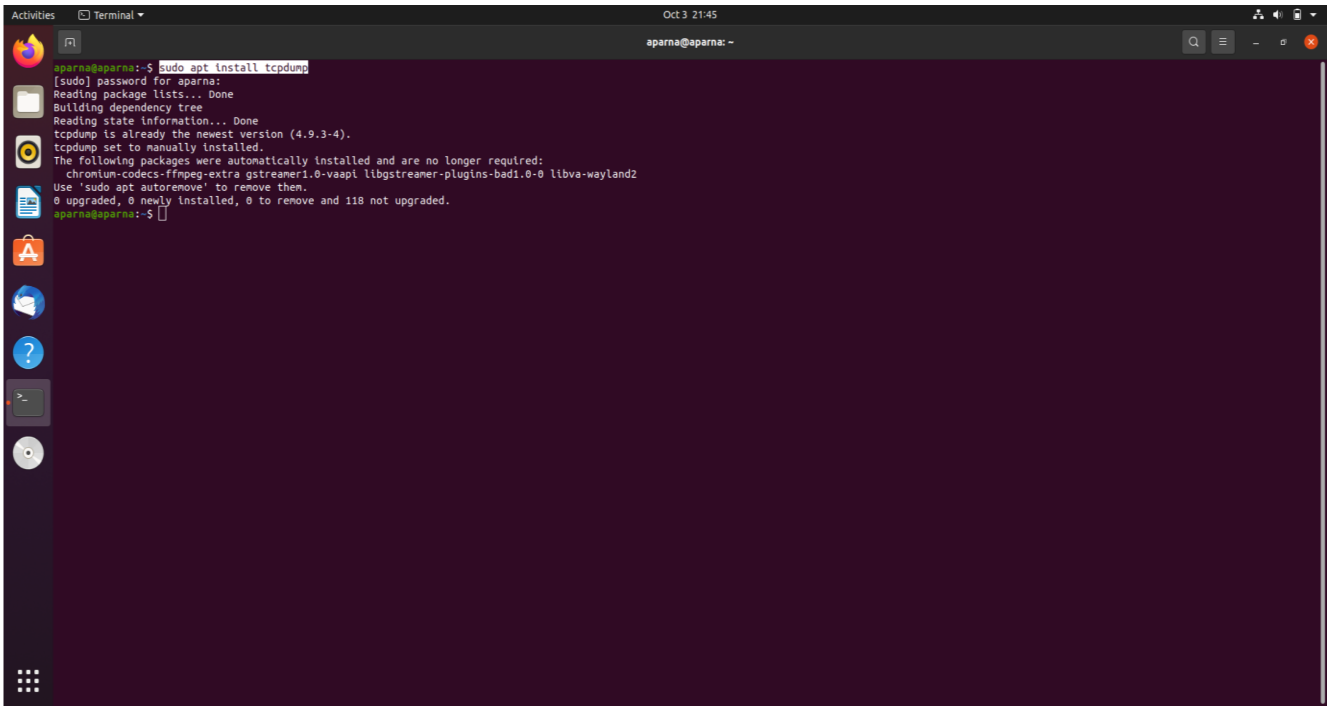
Result

<u>Tcpdump</u>

tcpdump is a packet sniffing and packet analyzing tool for a System Administrator to troubleshoot connectivity issues in Linux. It is used to capture, filter, and analyze network traffic such as TCP/IP packets going through your system. It is many times used as a security tool as well. It saves the captured information in a pcap file, these pcap files can then be opened through Wireshark or through the command tool itself.

<u>Installing tcpdump tool in Linux</u>

`Sudo apt install tcpdump`



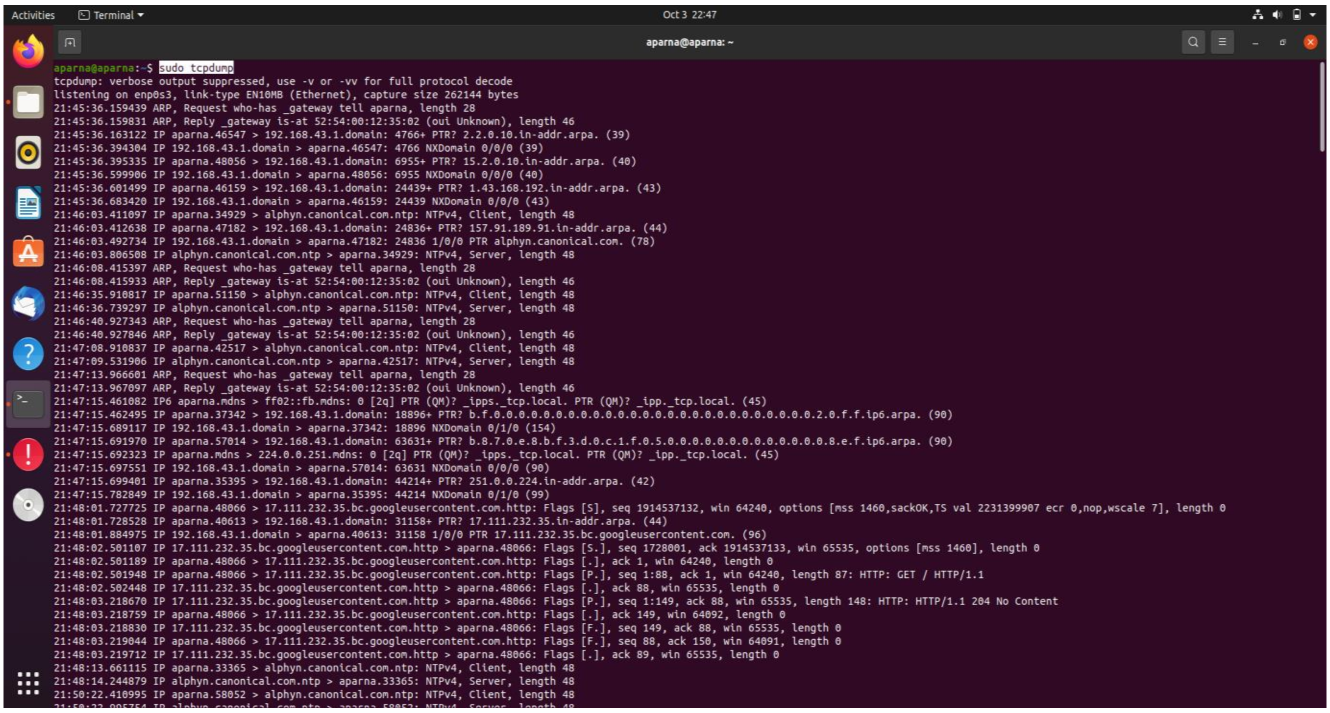
```
aparna@aparna:~$ sudo apt install tcpdump
[sudo] password for aparna:
Reading package lists... Done
Building dependency tree
Reading state information... Done
tcpdump is already the newest version (4.9.3-4).
tcpdump set to manually installed.
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libgstreamer-plugins-bad1.0-0 libva-wayland2
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 118 not upgraded.
aparna@aparna:~$ 
```

<u>**Working with tcpdump command**</u>

1. To capture the packets of current network interface

```
sudo tcpdump
```

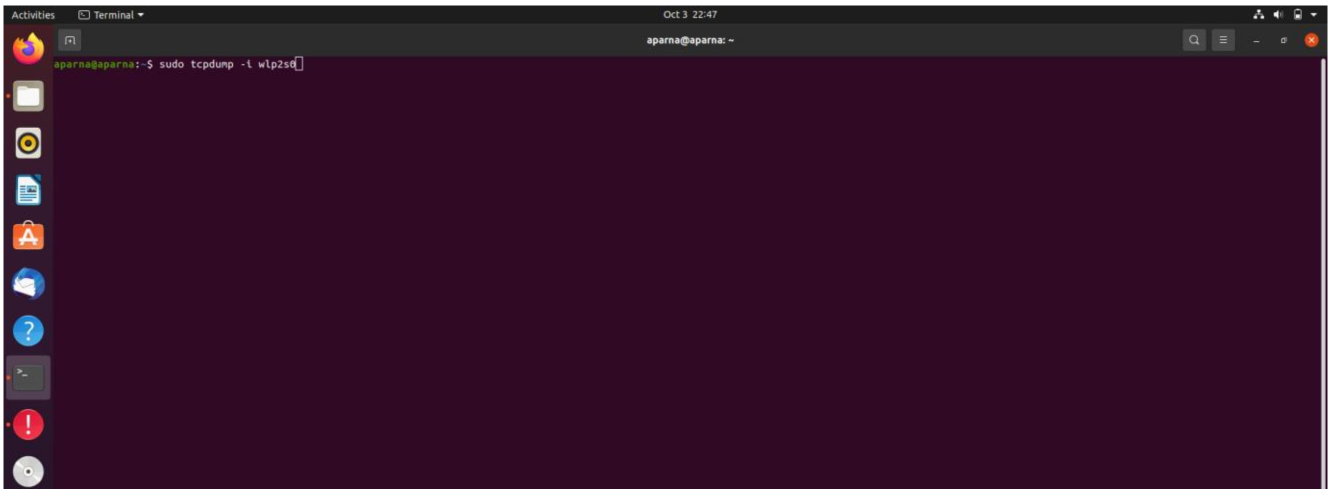
This will capture the packets from the current interface of the network through which the system is connected to the internet.



```
aparna@aparna:~$ sudo tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
21:45:36.159439 ARP, Request who-has _gateway tell aparna, length 28
21:45:36.159831 ARP, Reply _gateway is-at 52:54:00:12:35:02 (oui Unknown), length 46
21:45:36.163122 IP aparna.46547 > 192.168.43.1.domain: 4766+ PTR? 2.2.0.10.in-addr.arpa. (39)
21:45:36.394304 IP 192.168.43.1.domain > aparna.46547: 4766 NXDomain 0/0/0 (39)
21:45:36.395335 IP aparna.48056 > 192.168.43.1.domain: 6955+ PTR? 15.2.0.10.in-addr.arpa. (40)
21:45:36.599906 IP 192.168.43.1.domain > aparna.48056: 6955 NXDomain 0/0/0 (40)
21:45:36.601499 IP aparna.46159 > 192.168.43.1.domain: 24439+ PTR? 1.43.168.192.in-addr.arpa. (43)
21:45:36.683420 IP 192.168.43.1.domain > aparna.46159: 24439 NXDomain 0/0/0 (43)
21:46:03.411097 IP aparna.34929 > alphyn.canonical.com.ntp: NTPv4, Client, length 48
21:46:03.412638 IP aparna.47182 > 192.168.43.1.domain: 24836+ PTR? 157.91.189.91.in-addr.arpa. (44)
21:46:03.492734 IP 192.168.43.1.domain > aparna.47182: 24836 1/0/0 PTR alphyn.canonical.com. (78)
21:46:03.806508 IP alphyn.canonical.com.ntp > aparna.34929: NTPv4, Server, length 48
21:46:08.415397 ARP, Request who-has _gateway tell aparna, length 28
21:46:08.415933 ARP, Reply _gateway is-at 52:54:00:12:35:02 (oui Unknown), length 46
21:46:35.910817 IP aparna.51150 > alphyn.canonical.com.ntp: NTPv4, Client, length 48
21:46:36.739297 IP alphyn.canonical.com.ntp > aparna.51150: NTPv4, Server, length 48
21:46:40.927343 ARP, Request who-has _gateway tell aparna, length 28
21:46:40.927846 ARP, Reply _gateway is-at 52:54:00:12:35:02 (oui Unknown), length 46
21:47:08.910837 IP aparna.42517 > alphyn.canonical.com.ntp: NTPv4, Client, length 48
21:47:09.531906 IP alphyn.canonical.com.ntp > aparna.42517: NTPv4, Server, length 48
21:47:13.966601 ARP, Request who-has _gateway tell aparna, length 28
21:47:13.967097 ARP, Reply _gateway is-at 52:54:00:12:35:02 (oui Unknown), length 46
21:47:15.461082 IP6 aparna.mdns > ff02::fb.mdns: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
21:47:15.462495 IP aparna.37342 > 192.168.43.1.domain: 18896+ PTR? b.f.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.ip6.arpa. (90)
21:47:15.689117 IP 192.168.43.1.domain > aparna.37342: 18896 NXDomain 0/1/0 (154)
21:47:15.691970 IP aparna.57014 > 192.168.43.1.domain: 63631+ PTR? b.8.7.0.e.8.b.f.3.d.0.c.1.f.0.5.0.0.0.0.0.0.0.0.0.0.0.0.0.8.e.f.ip6.arpa. (90)
21:47:15.692323 IP aparna.mdns > 224.0.0.251.mdns: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
21:47:15.697551 IP 192.168.43.1.domain > aparna.57014: 63631 NXDomain 0/0/0 (90)
21:47:15.699401 IP aparna.35395 > 192.168.43.1.domain: 44214+ PTR? 251.0.0.224.in-addr.arpa. (42)
21:47:15.782849 IP 192.168.43.1.domain > aparna.35395: 44214 NXDomain 0/1/0 (99)
21:48:01.727725 IP aparna.48066 > 17.111.232.35.bc.googleusercontent.com.http: Flags [S], seq 1914537132, win 64240, options [mss 1460,sackOK,TS val 2231399907 ecr 0,nop,wscale 7], length 0
21:48:01.728528 IP aparna.40613 > 192.168.43.1.domain: 31158+ PTR? 17.111.232.35.in-addr.arpa. (44)
21:48:01.884975 IP 192.168.43.1.domain > aparna.40613: 31158 1/0/0 PTR 17.111.232.35.bc.googleusercontent.com. (96)
21:48:02.501107 IP 17.111.232.35.bc.googleusercontent.com.http > aparna.48066: Flags [S.], seq 1728001, ack 1914537133, win 65535, options [mss 1460], length 0
21:48:02.501189 IP aparna.48066 > 17.111.232.35.bc.googleusercontent.com.http: Flags [.], ack 1, win 64240, length 0
21:48:02.501948 IP aparna.48066 > 17.111.232.35.bc.googleusercontent.com.http: Flags [P.], seq 1:88, ack 1, win 64240, length 87: HTTP: GET / HTTP/1.1
21:48:02.502448 IP 17.111.232.35.bc.googleusercontent.com.http > aparna.48066: Flags [.], ack 88, win 65535, length 0
21:48:03.218670 IP 17.111.232.35.bc.googleusercontent.com.http > aparna.48066: Flags [P.], seq 1:149, ack 88, win 65535, length 148: HTTP: HTTP/1.1 204 No Content
21:48:03.218759 IP aparna.48066 > 17.111.232.35.bc.googleusercontent.com.http: Flags [.], ack 149, win 64092, length 0
21:48:03.218830 IP 17.111.232.35.bc.googleusercontent.com.http > aparna.48066: Flags [F.], seq 149, ack 88, win 65535, length 0
21:48:03.219044 IP aparna.48066 > 17.111.232.35.bc.googleusercontent.com.http: Flags [F.], seq 88, ack 150, win 64091, length 0
21:48:03.219712 IP 17.111.232.35.bc.googleusercontent.com.http > aparna.48066: Flags [.], ack 89, win 65535, length 0
21:48:13.661115 IP aparna.33365 > alphyn.canonical.com.ntp: NTPv4, Client, length 48
21:48:14.244879 IP alphyn.canonical.com.ntp > aparna.33365: NTPv4, Server, length 48
21:50:22.410995 IP aparna.58052 > alphyn.canonical.com.ntp: NTPv4, Client, length 48
```

1. To capture packets from a specific network interface

```
sudo tcpdump -i wlp2s0
```

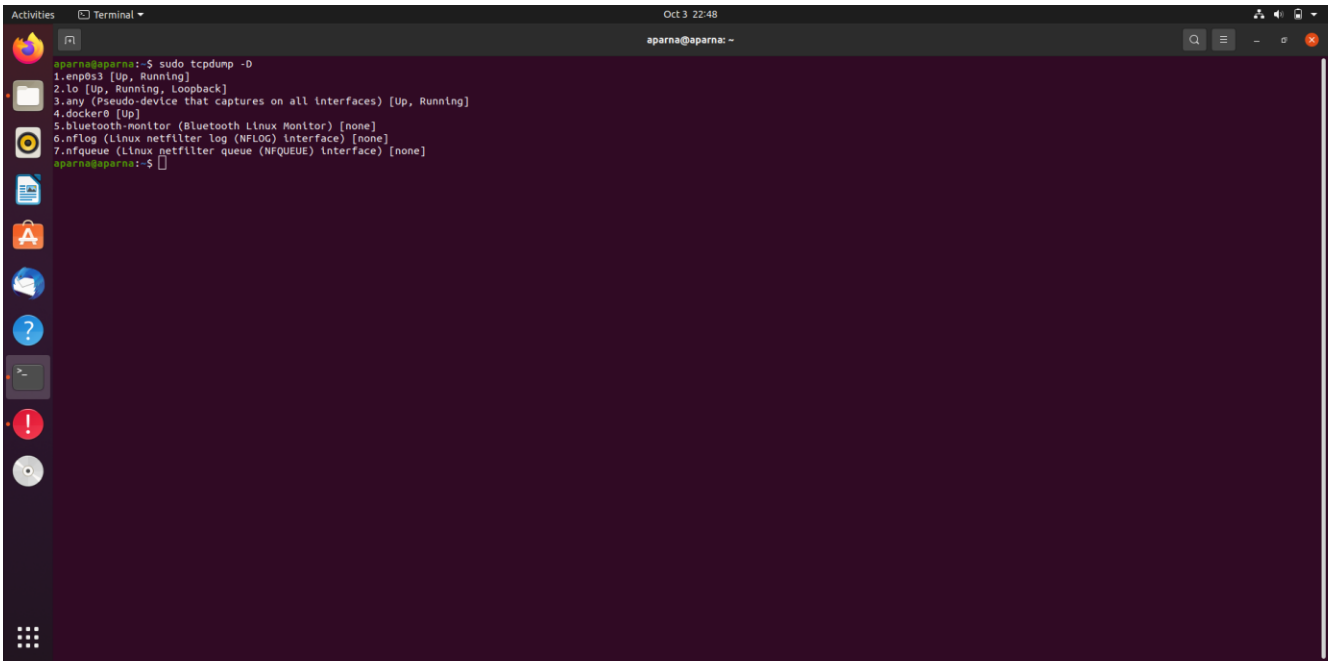


```
aparna@aparna:~$ sudo tcpdump -i wlp2s0
```

This command will now capture the packets from wlp2s0 network interface.

3) To display all available interfaces

```
sudo tcpdump -D
```



```
aparna@aparna:~$ sudo tcpdump -D
1.enp0s3 [Up, Running]
2.lo [Up, Running, Loopback]
3.any (Pseudo-device that captures on all interfaces) [Up, Running]
4.docker0 [Up]
5.bluetooth-monitor (Bluetooth Linux Monitor) [none]
6.nflog (Linux netfilter log (NFLOG) interface) [none]
7.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
aparna@aparna:~$ 
```

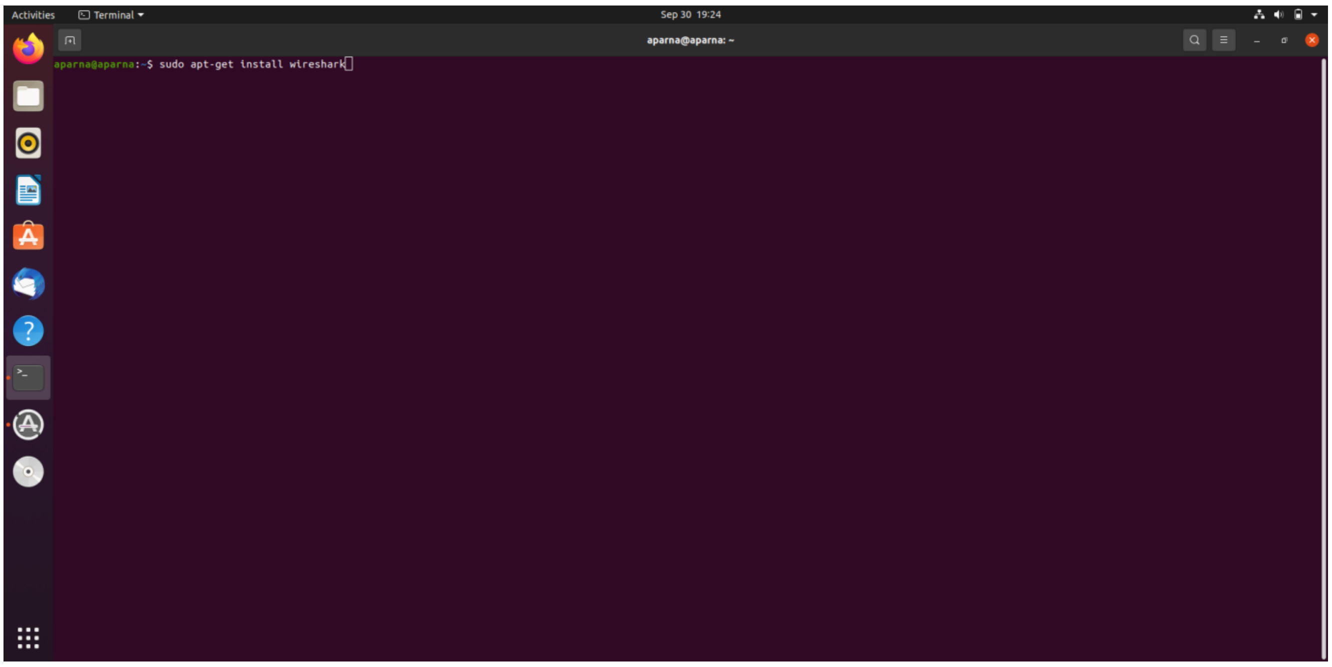
Wireshark

Wireshark is a software tool used to monitor the network traffic through a network interface. It is the most widely used network monitoring tool today.
Wireshark is loved equally by system administrators, network engineers, network enthusiasts, network security professionals and black hat hackers. The extent of its popularity is such, that experience with Wireshark is considered as a valuable/essential trait in a computer networking related professional.
There are many reasons why Wireshark is so popular :

- It has a great GUI as well as a conventional CLI(T Shark).
- It offers network monitoring on almost all types of network standards (ethernet, wlan, Bluetooth etc)
- It is open source with a large community of backers and developers.
- All the necessary components for monitoring, analysing and documenting the network traffic are present. It is free to use.

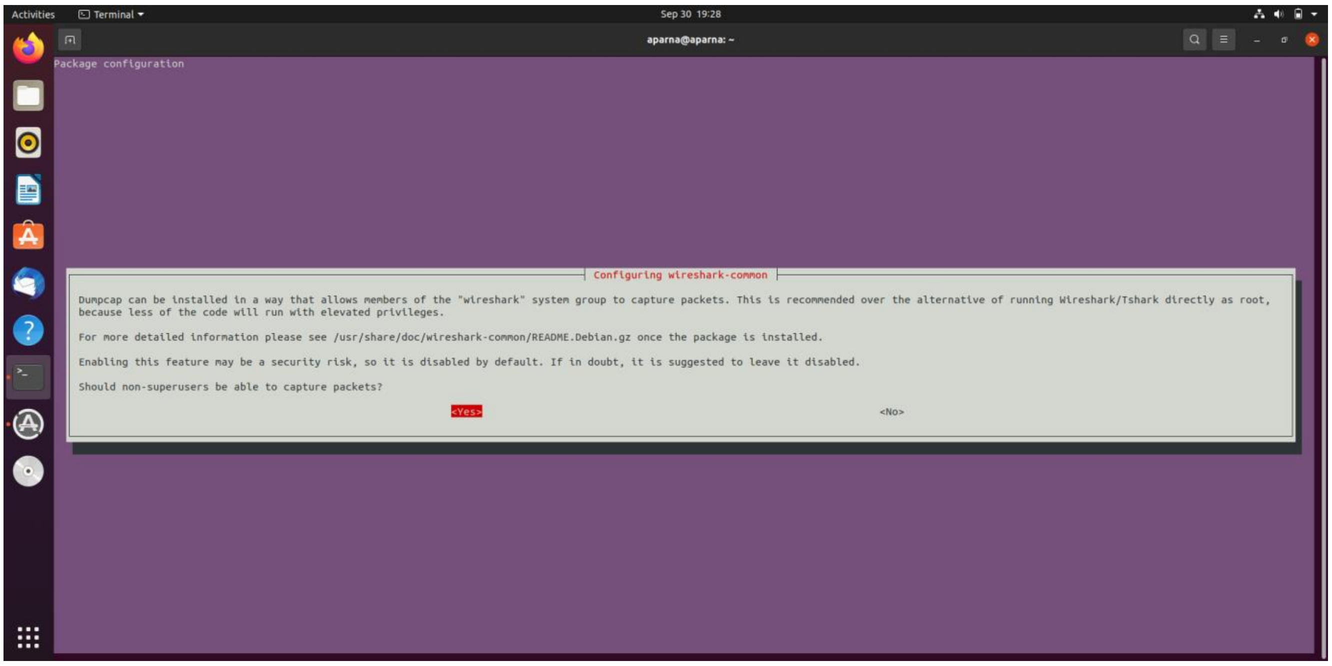
Wireshark installation

`sudo apt install wireshark` command is used to install wireshark in linux



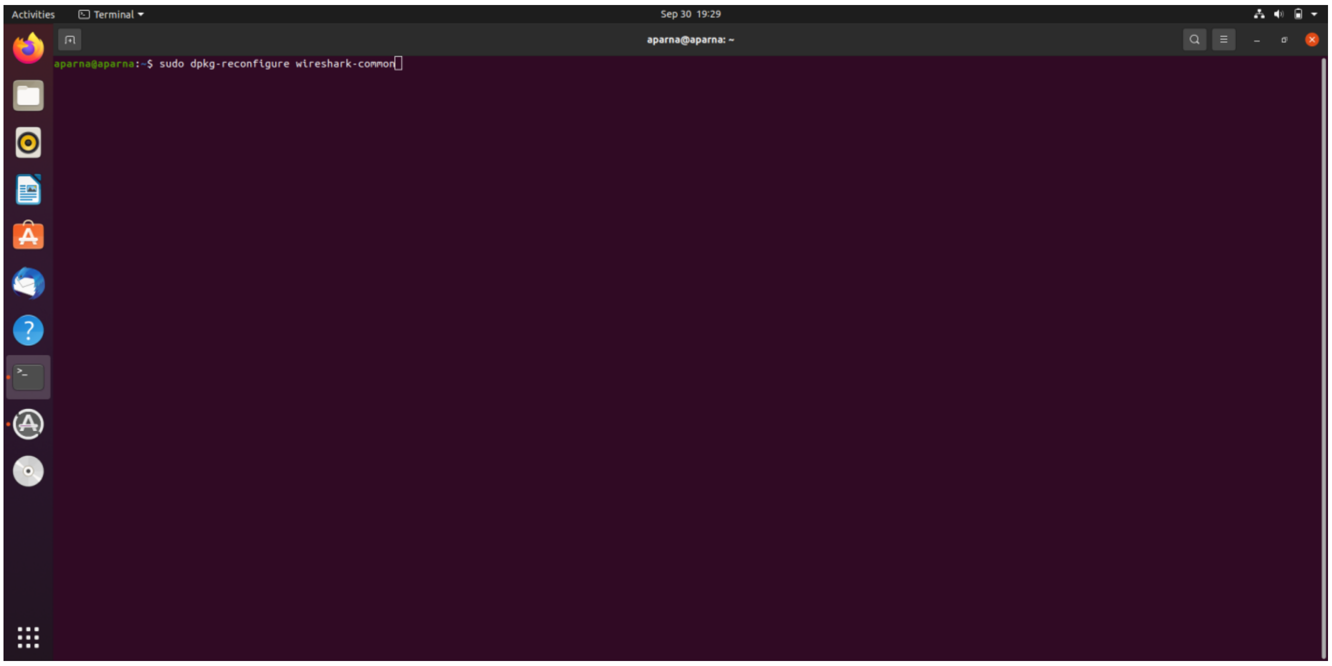
```
aparna@aparna:~$ sudo apt-get install wireshark
```

Enter Y to continue



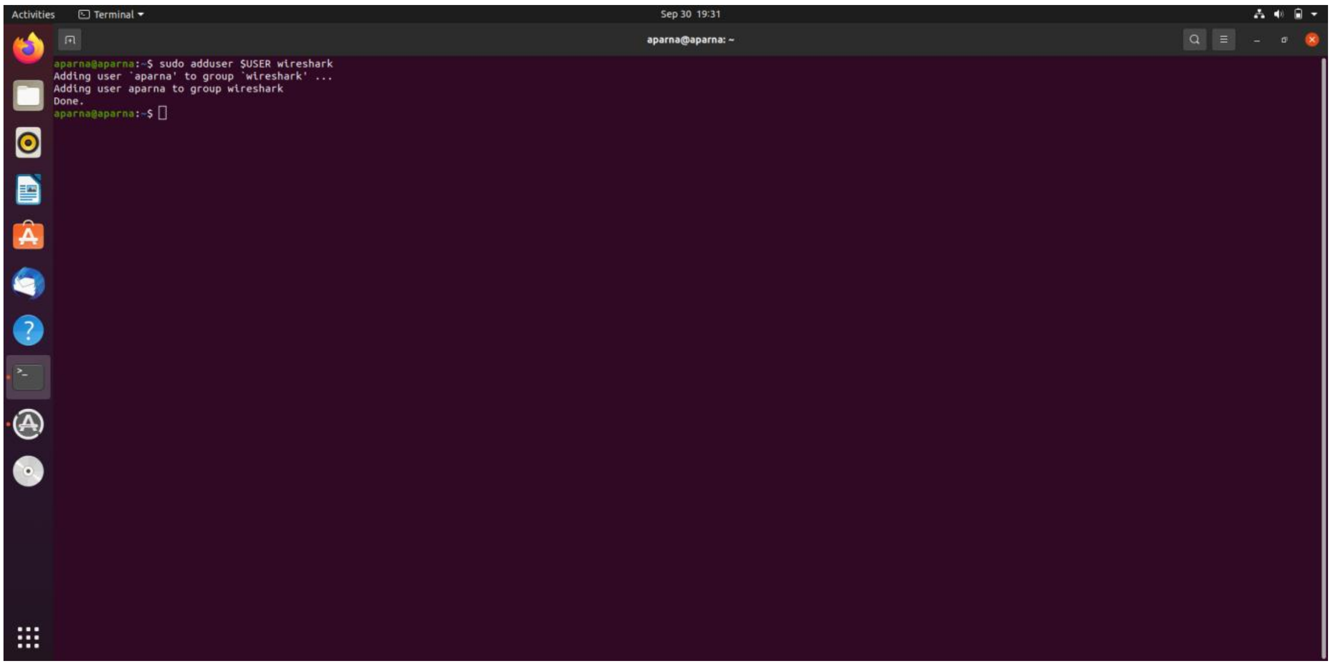
```
Package configuration

Configuring wireshark-common

Dumpcap can be installed in a way that allows members of the "wireshark" system group to capture packets. This is recommended over the alternative of running Wireshark/Tshark directly as root,
because less of the code will run with elevated privileges.

For more detailed information please see /usr/share/doc/wireshark-common/README.Debian.gz once the package is installed.

Enabling this feature may be a security risk, so it is disabled by default. If in doubt, it is suggested to leave it disabled.

Should non-superusers be able to capture packets?

                    <Yes>                                              <No>
```

Next give the command `sudo dpkg-reconfigure wireshark-common`



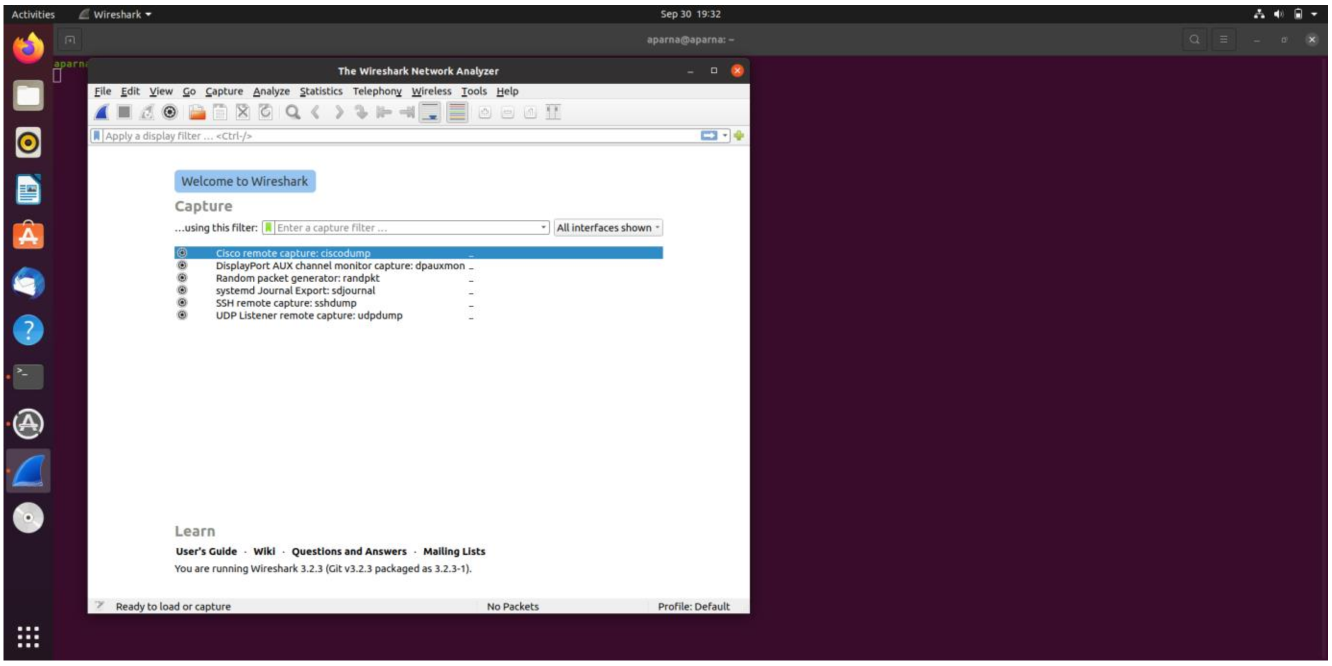
```
aparna@aparna:~$ sudo dpkg-reconfigure wireshark-common
```

Next `sudo adduser $USER wireshark`



```
aparna@aparna:~$ sudo adduser $USER wireshark
Adding user `aparna' to group `wireshark' ...
Adding user aparna to group wireshark
Done.
aparna@aparna:~$
```

You can give appropriate option according to your need. After this the wireshark will be downloaded to the system
On launching Wireshark, you will see a screen like this:


Activities
Wireshark
Sep 30 19:32
aparna@aparna: ~
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Welcome to Wireshark
Capture
…using this filter:
Enter a capture filter …
All interfaces shown
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists

```
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
```

You are running Wireshark 3.2.3 (Git v3.2.3 packaged as 3.2.3-1).

Ready to load or capture   No Packets   Profile: Default

The basic features of Wireshark are:

<u>1) Packet Monitor:</u> This segment visually shows the packets flowing inside the network. There are colour codes for each type of packets. The packets are shown with following information :

1. Source address
2. Destination address
3. Packet type
4. Hex dump of the packet
5. Contents of the packet in text
6. Source port(if applicable)
7. Destination port(if applicable)

<u>2) Import from a capture file:</u>

This feature lets you import packets dump from a capture file to analyse further. There are many formats supported by Wireshark, some of them are:

- pcapng
- libpcap
- Oracle snoop and atmsnoop
- Finisar (previously Shomiti) Surveyor captures
- Microsoft Network Monitor captures
- Novell LANalyzer captures
- AIX iptrace captures
- Cinco Networks NetXray captures
- Network Associates Windows-based Sniffer and Sniffer Pro captures
- Network General/Network Associates DOS-based Sniffer (compressed or uncompressed) captures
- AG Group/WildPackets/Savvius EtherPeek/TokenPeek/AiroPeek/EtherHelp/PacketGrabber captures
- RADCOM's WAN/LAN Analyzer captures
- Network Instruments Observer version 9 captures
- Lucent/Ascend router debug output
- HP-UX's nettl

•Toshiba’s ISDN routers dump output

•ISDN4BSD i4btrace utility

•Traces from the EyeSDN USB S0

•IPLog format from the Cisco Secure Intrusion Detection System pppd logs (pppdump format)

•the output from VMS’s TCPIPtrace/TCPtrace/UCX$TRACE utilities

•the text output from the DBS Etherwatch VMS utility

•Visual Networks’ Visual UpTime traffic capture

•the output from CoSine L2 debug

•the output from Accellent’s 5Views LAN agents

•Endace Measurement Systems’ ERF format captures

•Linux Bluez Bluetooth stack hcidump -w traces

•Catapult DCT2000 .out files

•Gammu generated text output from Nokia DCT3 phones in Netmonitor mode

•IBM Series (OS/400) Comm traces (ASCII & UNICODE)

•Juniper Netscreen snoop captures

•Symbian OS btsnoop captures

•Tamosoft CommView captures

•Textronix K12xx 32bit .rf5 format captures

•Textronix K12 text file format captures

•Apple PacketLogger captures

•Captures from Aethra Telecommunications’PC108 software

3) Export to a capture file: Wireshark lets you save the results as a capture file to continue working on them at later point of time. The supported formats are:

•pcapng (*.pcapng)

•libpcap, tcpdump and various other tools using tcpdump’s capture format (*.pcap, *.cap, *.dmp)

•Accellent 5Views (*.5vw)

•HP-UX’s nettl (*.TRC0, *.TRC1)

•Microsoft Network Monitor – NetMon (*.cap)

•Network Associates Sniffer – DOS (*.cap, *.enc, *.trc, *fdc, *.syc)

•Network Associates Sniffer – Windows (*.cap)

•Network Instruments Observer version 9 (*.bfr)

•Novell LANalyzer (*.tr1)

•Oracle (previously Sun) snoop (*.snoop, *.cap)

•Visual Networks Visual UpTime traffic (*.*).

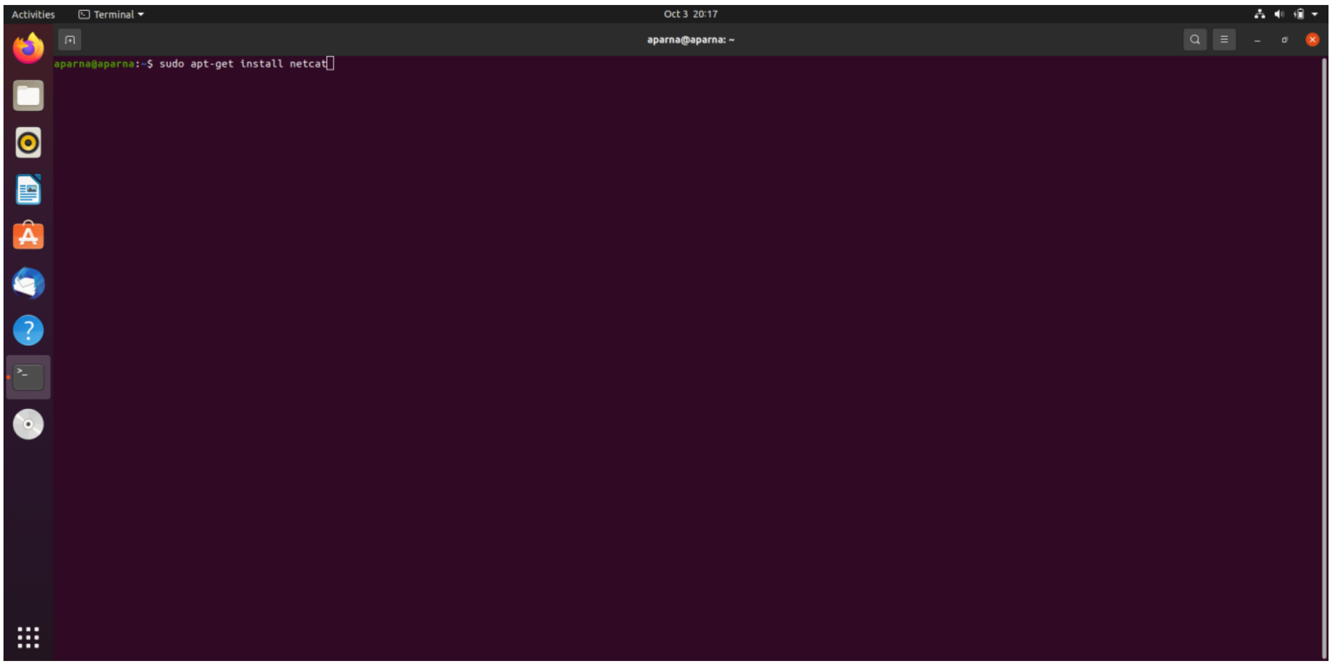
<u>**Netcat:**</u>

Netcat (or nc in short) is a simple yet powerful networking command-line tool used for performing any operation in Linux related to TCP, UDP, or UNIX-domain sockets.

Netcat can be used for port scanning, port redirection, as a port listener (for incoming connections); it can also be used to open remote connections and so many other things. Besides, you can use it as a backdoor to gain access to a target server.

<u>Installing netcat on linux:</u>

```
sudo apt-get install netcat
```



```
aparna@aparna:~$ sudo apt-get install netcat
```

<u>Port scanning:</u>

Netcat can be used for port scanning: to know which ports are open and running services on a target machine. It can scan a single or multiple or a range of open ports.

The -z option sets nc to simply scan for listening daemons, without actually sending any data to them. The -v option enables verbose mode and -w specifies a timeout for connection that can not be established.

Syntax:

```
nc -vz IP_address port
```

Connection timed out:

A *connection timed out* response indicates that your connection is not working, which could mean your firewall is blocking the port. Test the connection status by adding a rule that accepts connections on the required port.

Connection succeeded

If the initial connection succeeds, Netcat can connect to the service. Look at the connection in more detail.

Syntax:

```
nc -vt IP Address Port
```

Closing the connection

You can terminate the connection by either pressing Ctrl-C or type the service-specific quit command.