

EXPIRIMENT 9:

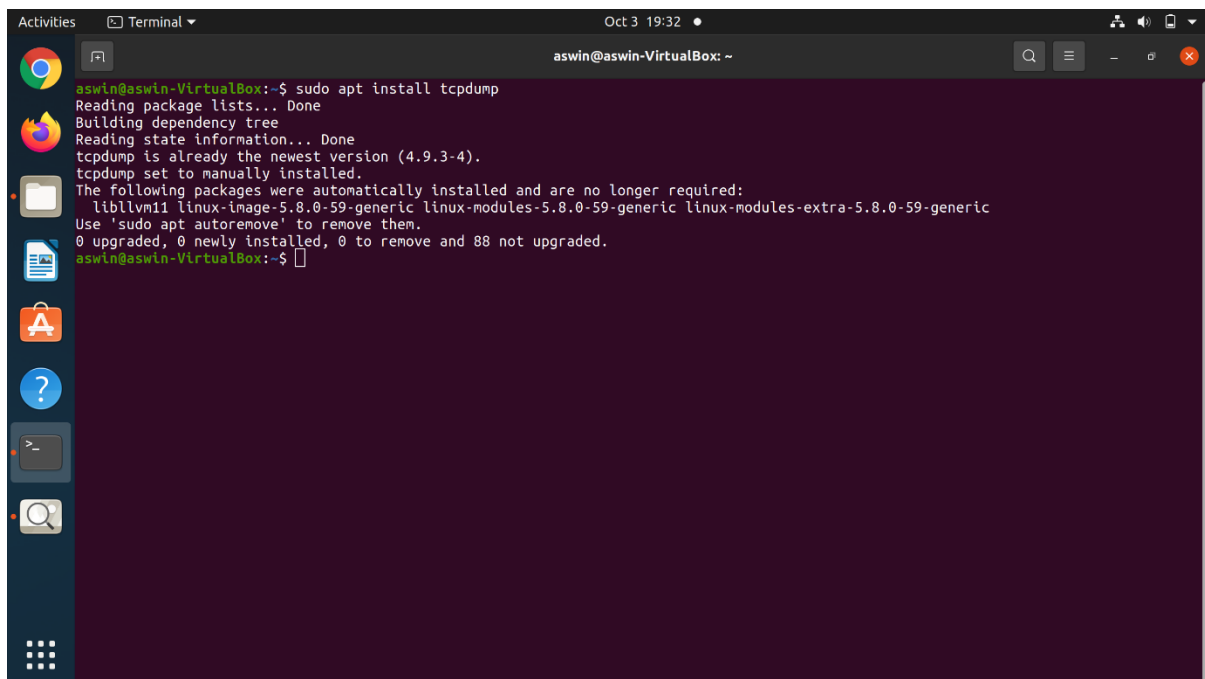
Aim: Analysing network packet stream using tcpdump and wireshark. Perform basic network service tests using nc.

Solution :-

tcpdump:

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.

Installing tcpdump tool in Linux

A screenshot of a Linux terminal window titled 'aswin@aswin-VirtualBox: ~'. The terminal shows the command 'sudo apt install tcpdump' being executed. The output indicates that tcpdump is already installed at version 4.9.3-4 and is set to manually installed. It also lists several packages that were automatically installed and are no longer required, including libllvm11, linux-image-5.8.0-59-generic, linux-modules-5.8.0-59-generic, and linux-modules-extra-5.8.0-59-generic. The terminal shows the command 'sudo apt autoremove' being used to remove these packages. The final output shows '0 upgraded, 0 newly installed, 0 to remove and 88 not upgraded.' The terminal window has a dark background and a light-colored text. The window title bar shows 'Oct 3 19:32' and some system icons on the right. On the left side of the terminal window, there is a vertical sidebar with various application icons like a web browser, file manager, and terminal.

Working with tcpdump command

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.

```
Activities Terminal Oct 3 19:38 aswin@aswin-VirtualBox: ~
aswin@aswin-VirtualBox:~$ 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
19:37:47.719939 IP aswin-VirtualBox.37330 > reliance.reliance.domain: 3152+ A? clientservices.googleapis.com. (47)
19:37:47.721684 IP aswin-VirtualBox.37503 > reliance.reliance.domain: 7886+ PTR? 1.29.168.192.in-addr.arpa. (43)
19:37:47.731542 IP reliance.reliance.domain > aswin-VirtualBox.37503: 7886* 1/0/0 PTR reliance.reliance. (74)
19:37:47.733147 IP aswin-VirtualBox.33829 > reliance.reliance.domain: 65016+ PTR? 15.2.0.10.in-addr.arpa. (40)
19:37:47.746457 IP reliance.reliance.domain > aswin-VirtualBox.37330: 3152 1/0/0 A 142.250.182.3 (63)
19:37:47.753309 IP reliance.reliance.domain > aswin-VirtualBox.33829: 65016 NXDomain* 0/1/0 (99)
19:37:47.840562 IP aswin-VirtualBox.45370 > maa05s18-in-f3.1e100.net.https: Flags [S], seq 1715217196, win 64240, options [mss 1460, sackOK, TS val 1874672962 ecr 0, nop, wscale 7], length 0
19:37:47.841215 IP aswin-VirtualBox.56827 > reliance.reliance.domain: 30272+ PTR? 3.182.250.142.in-addr.arpa. (44)
19:37:47.860895 IP reliance.reliance.domain > aswin-VirtualBox.56827: 30272 1/0/0 PTR maa05s18-in-f3.1e100.net. (82)
19:37:47.862701 IP maa05s18-in-f3.1e100.net.https > aswin-VirtualBox.45370: Flags [S], seq 11008001, ack 1715217197, win 65535, options [mss 1460], length 0
19:37:47.862774 IP aswin-VirtualBox.45370 > maa05s18-in-f3.1e100.net.https: Flags [.] , ack 1, win 64240, length 0
19:37:47.893171 IP aswin-VirtualBox.45370 > maa05s18-in-f3.1e100.net.https: Flags [P.] , seq 1:518, ack 1, win 64240, length 517
19:37:47.894055 IP maa05s18-in-f3.1e100.net.https > aswin-VirtualBox.45370: Flags [.] , ack 518, win 65535, length 0
19:37:47.947248 IP maa05s18-in-f3.1e100.net.https > aswin-VirtualBox.45370: Flags [P.] , seq 1:1431, ack 518, win 65535, length 1430
19:37:47.947319 IP aswin-VirtualBox.45370 > maa05s18-in-f3.1e100.net.https: Flags [.] , ack 1431, win 62920, length 0
19:37:47.947359 IP maa05s18-in-f3.1e100.net.https > aswin-VirtualBox.45370: Flags [P.] , seq 1431:2861, ack 518, win 65535, length 1430
19:37:47.947415 IP aswin-VirtualBox.45370 > maa05s18-in-f3.1e100.net.https: Flags [.] , ack 2861, win 62920, length 0
19:37:47.947788 IP maa05s18-in-f3.1e100.net.https > aswin-VirtualBox.45370: Flags [P.] , seq 2861:4684, ack 518, win 65535, length 1823
19:37:47.947825 IP aswin-VirtualBox.45370 > maa05s18-in-f3.1e100.net.https: Flags [.] , ack 4684, win 61097, length 0
19:37:48.929669 IP aswin-VirtualBox.41307 > reliance.reliance.domain: 25+ A? accounts.google.com. (37)
19:37:48.957103 IP reliance.reliance.domain > aswin-VirtualBox.41307: 25 1/0/0 A 142.250.183.13 (53)
19:37:48.959195 IP aswin-VirtualBox.54816 > bom07s30-in-f13.1e100.net.443: UDP, length 1350
19:37:48.960349 IP aswin-VirtualBox.43769 > reliance.reliance.domain: 6839+ PTR? 13.183.250.142.in-addr.arpa. (45)
19:37:48.981390 IP reliance.reliance.domain > aswin-VirtualBox.43769: 6839 1/0/0 PTR bom07s30-in-f13.1e100.net. (84)
19:37:49.058272 IP aswin-VirtualBox.50570 > bom07s30-in-f13.1e100.net.https: Flags [S], seq 1592066534, win 64240, options [mss 1460, sackOK, TS val 1236385742 ecr 0, nop, wscale 7], length 0
19:37:49.149516 IP aswin-VirtualBox.54816 > bom07s30-in-f13.1e100.net.443: UDP, length 1350
19:37:49.308992 IP aswin-VirtualBox.50572 > bom07s30-in-f13.1e100.net.https: Flags [S], seq 3426555662, win 64240, options [mss 1460, sackOK, TS val 1236385993 ecr 0, nop, wscale 7], length 0
```

2. To capture packets from a specific network interface

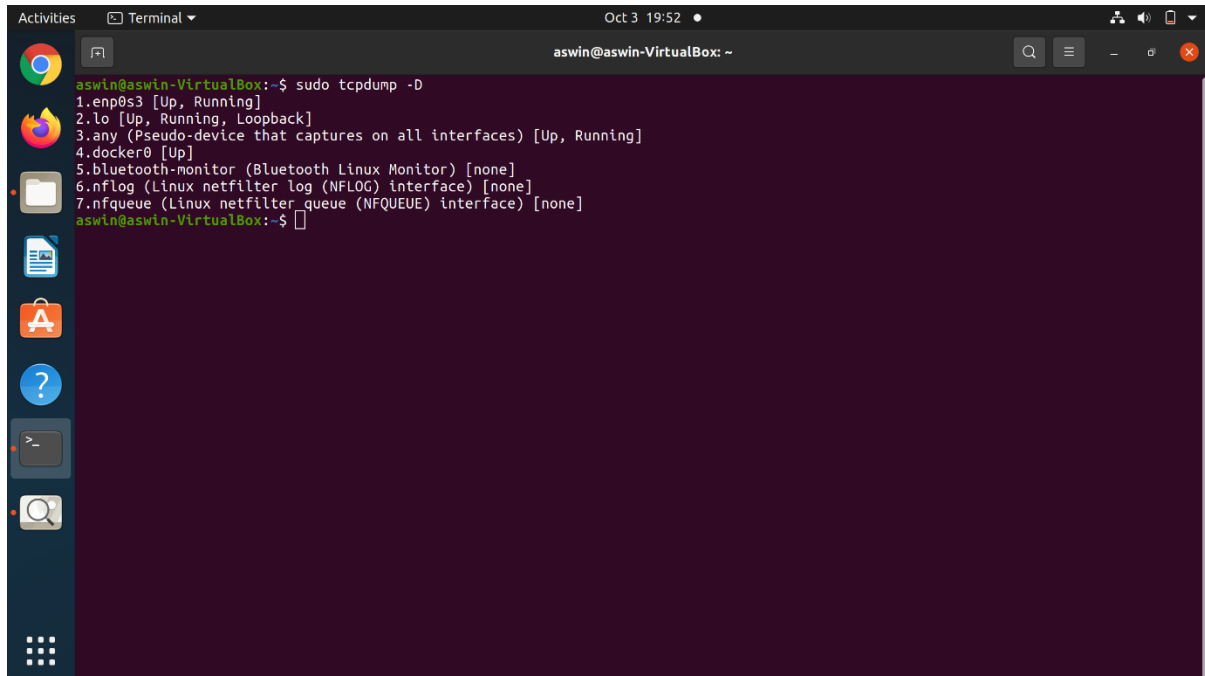
`sudo tcpdump -i enp0s3`

```
Activities Terminal Oct 3 20:03 aswin@aswin-VirtualBox: ~
aswin@aswin-VirtualBox:~$ sudo tcpdump -i enp0s3
[sudo] password for aswin:
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
20:02:36.400174 IP aswin-VirtualBox.51096 > maa05s05-in-f16.1e100.net.http: Flags [.] , ack 29509436, win 62780, length 0
20:02:36.400561 IP maa05s05-in-f16.1e100.net.http > aswin-VirtualBox.51096: Flags [.] , ack 1, win 65535, length 0
20:02:36.405088 IP aswin-VirtualBox.42334 > reliance.reliance.domain: 62269+ PTR? 176.163.217.172.in-addr.arpa. (46)
20:02:36.428800 IP reliance.reliance.domain > aswin-VirtualBox.42334: 62269 1/0/0 PTR maa05s05-in-f16.1e100.net. (85)
20:02:36.430060 IP aswin-VirtualBox.58457 > reliance.reliance.domain: 8192+ PTR? 15.2.0.10.in-addr.arpa. (40)
20:02:36.450582 IP reliance.reliance.domain > aswin-VirtualBox.58457: 8192 NXDomain* 0/1/0 (99)
20:02:36.451983 IP aswin-VirtualBox.37891 > reliance.reliance.domain: 27867+ PTR? 1.29.168.192.in-addr.arpa. (43)
20:02:36.456826 IP reliance.reliance.domain > aswin-VirtualBox.37891: 27867* 1/0/0 PTR reliance.reliance. (74)
20:02:38.723594 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.https > aswin-VirtualBox.55118: Flags [P.] , seq 79209179:79209418, ack 1392979933, win 65535, length 239
20:02:38.723658 IP aswin-VirtualBox.55118 > whatsapp-cdn-shv-01-maa2.fbcdn.net.https: Flags [.] , ack 239, win 65535, length 0
20:02:38.724854 IP aswin-VirtualBox.51275 > reliance.reliance.domain: 48278+ PTR? 53.23.240.157.in-addr.arpa. (44)
20:02:38.755507 IP reliance.reliance.domain > aswin-VirtualBox.51275: 48278 1/0/0 PTR whatsapp-cdn-shv-01-maa2.fbcdn.net. (92)
20:02:44.775746 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.https > aswin-VirtualBox.55118: Flags [P.] , seq 239:447, ack 1, win 65535, length 208
20:02:44.775891 IP aswin-VirtualBox.55118 > whatsapp-cdn-shv-01-maa2.fbcdn.net.https: Flags [.] , ack 447, win 65535, length 0
20:02:44.861367 IP aswin-VirtualBox.57351 > reliance.reliance.domain: 50267+ A? mmx-ds.cdn.whatsapp.net. (41)
20:02:44.867673 IP reliance.reliance.domain > aswin-VirtualBox.57351: 50267 1/0/0 A 157.240.23.53 (57)
20:02:44.868996 IP aswin-VirtualBox.53161 > whatsapp-cdn-shv-01-maa2.fbcdn.net.443: UDP, length 1350
20:02:44.869443 IP aswin-VirtualBox.53161 > whatsapp-cdn-shv-01-maa2.fbcdn.net.443: UDP, length 75
20:02:44.869974 IP aswin-VirtualBox.53161 > whatsapp-cdn-shv-01-maa2.fbcdn.net.443: UDP, length 334
20:02:44.892175 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 1232
20:02:44.892544 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 1232
20:02:44.892559 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 187
20:02:44.893195 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 27
20:02:44.893207 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 56
20:02:44.893209 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 26
20:02:44.893225 IP aswin-VirtualBox.53161 > whatsapp-cdn-shv-01-maa2.fbcdn.net.443: UDP, length 81
20:02:44.893469 IP aswin-VirtualBox.53161 > whatsapp-cdn-shv-01-maa2.fbcdn.net.443: UDP, length 34
20:02:44.893884 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 1232
20:02:44.893901 IP whatsapp-cdn-shv-01-maa2.fbcdn.net.443 > aswin-VirtualBox.53161: UDP, length 1232
20:02:44.894091 IP aswin-VirtualBox.53161 > whatsapp-cdn-shv-01-maa2.fbcdn.net.443: UDP, length 38
```

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

3) To display all available interfaces

`sudo tcpdump -D`



```
aswin@aswin-VirtualBox:~$ 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]
aswin@aswin-VirtualBox:~$
```

wireshark

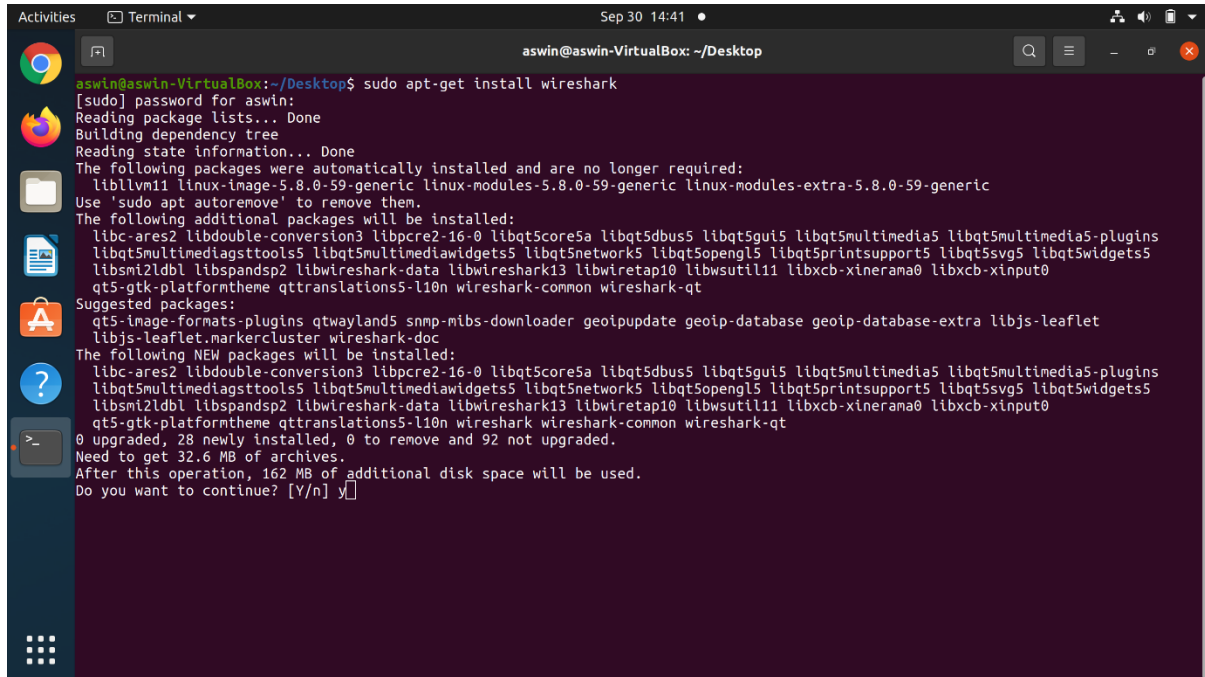
Wireshark is a software tool used to monitor the network traffic through a network interface. It is the most widely used network monitoring tool today. Wireshark is loved equally by system administrators, network engineers, network enthusiasts, network security professionals and black hat hackers. The 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.

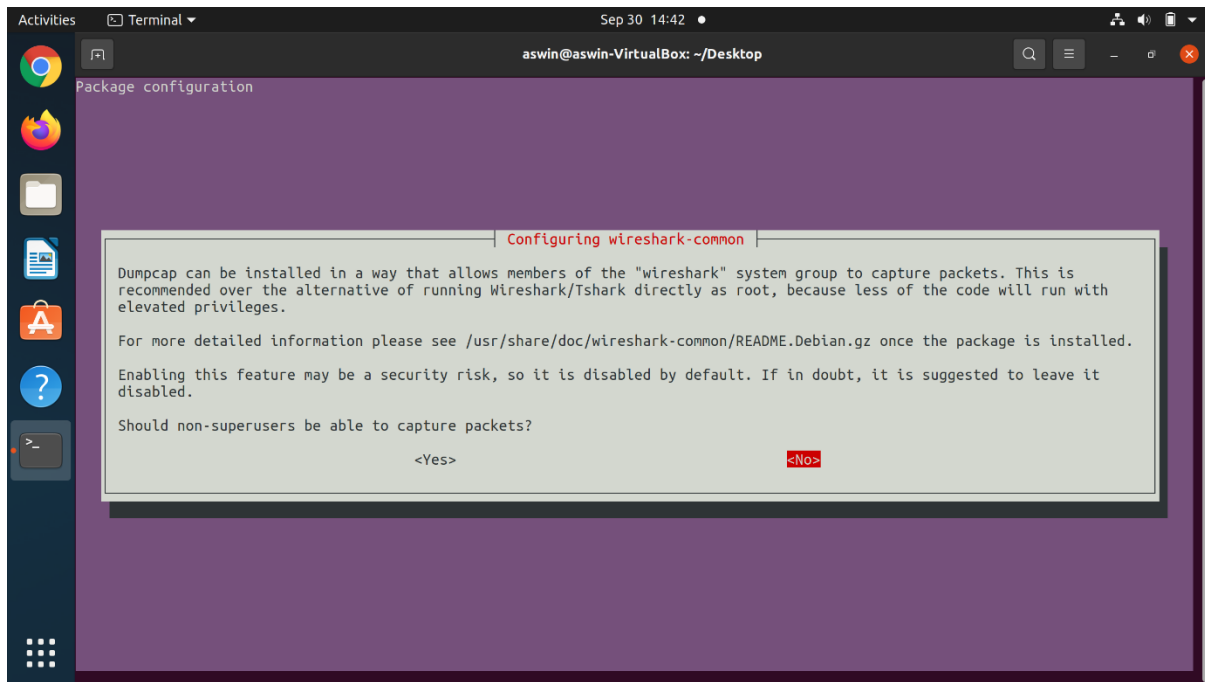
Wireshark installation:

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

A terminal window titled 'aswin@aswin-VirtualBox: ~/Desktop' showing the command 'sudo apt-get install wireshark' and its output. The output includes package lists, dependency tree building, state information reading, and a list of packages to be installed. It also shows suggested packages and a confirmation prompt 'Do you want to continue? [Y/n] y'.

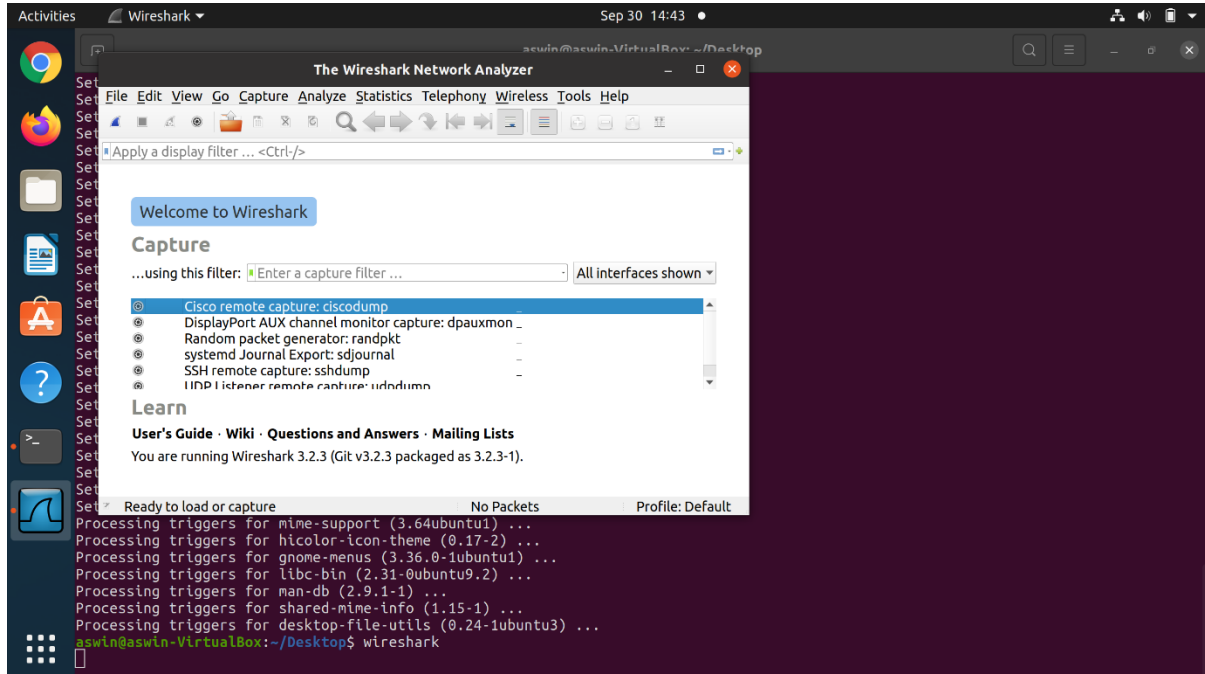
```
aswin@aswin-VirtualBox:~/Desktop$ sudo apt-get install wireshark
[sudo] password for aswin:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libllvm11 linux-image-5.8.0-59-generic linux-modules-5.8.0-59-generic linux-modules-extra-5.8.0-59-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libc-ares2 libdouble-conversion3 libpcre2-16-0 libqt5core5a libqt5dbus5 libqt5gui5 libqt5multimedia5 libqt5multimedia5-plugins
  libqt5multimediagsttools5 libqt5multimediawidgets5 libqt5network5 libqt5opengl5 libqt5sprintsupport5 libqt5svg5 libqt5widgets5
  libsmi2ldbl libspandsp2 libwireshark-data libwireshark13 libwiretap10 libwsutil11 libxcb-xinerama0 libxcb-xinput0
  qt5-gtk-platformtheme qttranslations5-l10n wireshark-common wireshark-qt
Suggested packages:
  qt5-image-formats-plugins qtwayland5 snmp-mibs-downloader geopipupdate geopip-database geopip-database-extra libjs-leaflet
  libjs-leaflet.markercluster wireshark-doc
The following NEW packages will be installed:
  libc-ares2 libdouble-conversion3 libpcre2-16-0 libqt5core5a libqt5dbus5 libqt5gui5 libqt5multimedia5 libqt5multimedia5-plugins
  libqt5multimediagsttools5 libqt5multimediawidgets5 libqt5network5 libqt5opengl5 libqt5sprintsupport5 libqt5svg5 libqt5widgets5
  libsmi2ldbl libspandsp2 libwireshark-data libwireshark13 libwiretap10 libwsutil11 libxcb-xinerama0 libxcb-xinput0
  qt5-gtk-platformtheme qttranslations5-l10n wireshark-common wireshark-qt
0 upgraded, 28 newly installed, 0 to remove and 92 not upgraded.
Need to get 32.6 MB of archives.
After this operation, 162 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Enter Y to continue



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:



The basic features of Wireshark are:

- 1) **Packet Monitor:** 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)

2) Import from a capture file:

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/PackageGrabber 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, *.sync)
- 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 (*.*)

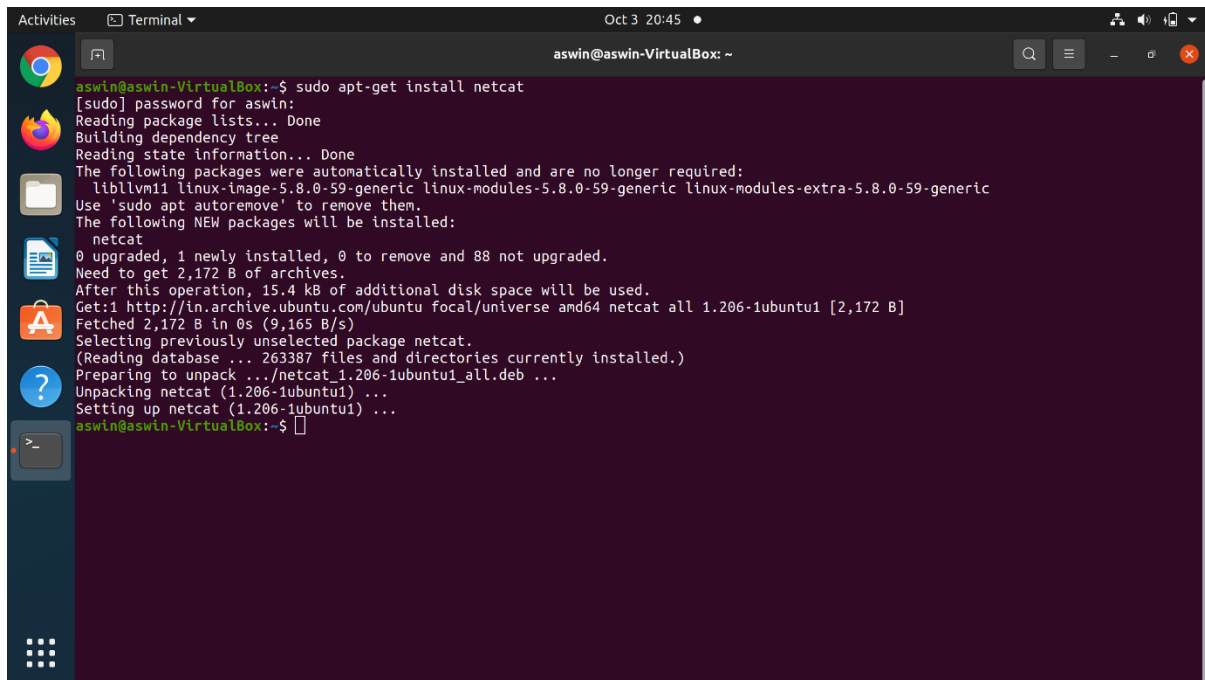
Netcat:

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.

Installing netcat on linux:

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


A screenshot of a terminal window titled "aswin@aswin-VirtualBox: ~". The terminal shows the command "sudo apt-get install netcat" being executed. The output includes the password prompt, package list reading, dependency tree building, and state information reading. It lists packages to be removed (libllvm11, linux-image-5.8.0-59-generic, linux-modules-5.8.0-59-generic, linux-modules-extra-5.8.0-59-generic) and the new package to be installed (netcat). It shows the disk space requirements and the download of netcat from the Ubuntu archive. The installation process is shown as successful, with netcat being unpacked and set up. The terminal ends with the prompt "aswin@aswin-VirtualBox:~\$".

```
aswin@aswin-VirtualBox:~$ sudo apt-get install netcat
[sudo] password for aswin:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libllvm11 linux-image-5.8.0-59-generic linux-modules-5.8.0-59-generic linux-modules-extra-5.8.0-59-generic
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  netcat
0 upgraded, 1 newly installed, 0 to remove and 88 not upgraded.
Need to get 2,172 B of archives.
After this operation, 15.4 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 netcat all 1.206-1ubuntu1 [2,172 B]
Fetched 2,172 B in 0s (9,165 B/s)
Selecting previously unselected package netcat.
(Reading database ... 263387 files and directories currently installed.)
Preparing to unpack .../netcat_1.206-1ubuntu1_all.deb ...
Unpacking netcat (1.206-1ubuntu1) ...
Setting up netcat (1.206-1ubuntu1) ...
aswin@aswin-VirtualBox:~$
```

Port scanning:

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 cannot 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.