Nmap Command in Linux

AIM:

To familiarize working of nmap in linux

DESCRIPTION:

Nmap is Linux command-line tool for network exploration and security auditing. This tool is generally used by hackers and cybersecurity enthusiasts and even by network and system administrators. It is used for the following purposes:

- ✓ Real time information of a network
- ✓ Detailed information of all the IPs activated on your network
- ✓ Number of ports open in a network
- ✓ Provide the list of live hosts
- ✓ Port, OS and Host scanning

PROCEDURE:

Installing Nmap

sudo apt-get install nmap

- To scan a System with Hostname and IP address. First, Scan using Hostname

nmap www.geeksforgeeks.org

Now let's Scan using IP Address ***nmap 172.217.27.174***

The nmap command allows scanning a system in various ways. In this we are performing a scan using the hostname as "geeksforgeeks" and IP address "172.217.27.174", to find all open ports, services, and MAC addresses on the system.

- To scan using "-v" option.

```
nmap -v www.geeksforgeeks.org
```

It is used to get more detailed information about the remote machines.

To scan multiple hosts

nmap 103.76.228.244 157.240.198.35 172.217.27.174

We can scan multiple hosts by writing IP addresses or hostnames with nmap.

- To scan whole subnet ***nmap 103.76.228.****

We can scan a whole subnet or IP range with nmap by providing “*” with it. It will scan a whole subnet and give the information about those hosts which are Up in the Network.

- To scan specific range of IP address ***nmap 192.168.29.1-20***

We can specify the range of IP addresses. This command will scan IP address 192.168.29.1 to 192.168.29.20 .

- To scan to detect firewall settings.

sudo nmap -sA 103.76.228.244

Detecting firewall settings can be useful during penetration testing and vulnerability scans. To detect it we use “-sA” option. This will provide you with information about firewall being active on the host. It uses an ACK scan to receive the information.

To identify Hostnames

sudo nmap -sL 103.76.228.244

We use “sL” option to find hostnames for the given host by completing a DNS query for each one. In addition to this “-n” command can be used to skip DNS resolution, while the “-R” command can be used to always resolve DNS.

To scan from a file

```
nmap -iL input.txt
```

If we have a long list of addresses that we need to scan, we can directly import a file through the command line. It will produce a scan for the given IP addresses.

- To get some help ***nmap -h***

We use the “-h” option if we have any questions about nmap or any of the given commands. It shows the help section for nmap command, including giving information regarding the available flags.

Here -sS flag is used for TCP SYN Scan, which is a stealthy and efficient method of scanning for open ports on a target system.

nmap -sS <Domain Name>

Here "-oG" flag can be used to store the nmap result in to specific file. ***nmap -sS <Domain Name> -oG <file-path>***

The "-sU" flag is used with nmap to perform a UDP scan, which allows the user to discover open UDP ports and services on a target system.

nmap -sU <Domain Name>

The "-sn" flag is used with nmap to perform a ping scan, which sends ICMP requests to a target host or network to determine hosts is up or not.

nmap -sn <Domain Name>

The "-p" flag is used with nmap to perform scan on a specific port or range of ports. (In our case it will scan port 80,443 and 21)

nmap -p 80 443 21 <Domain Name>

We can also specify the range of ports to scan on a network. (In this case it will scan all the ports in the range of 1 to 80)

```
nmap -p 1-80 <Domain Name>
```

Here -A indicates aggressive, it will give us extra information, like OS detection (-O), version detection, script scanning (-sC), and traceroute (–traceroute). It even provides a lot of valuable information about the host.

nmap -A <Domain Name>

Using this command we can discover the target hosting service or identify additional targets according to our needs for quickly tracing the path. nmap --trace out <Domain Name>

The command will just guess the running operating system (OS) on the host.

```
nmap -O <Domain Name>
```

WIRESHARK

AIM:

To provide deeper understanding of Network Protocol Analysis using Wireshark

DESCRIPTION:

Wireshark is an open-source packet analyzer, which is used for education, analysis, software development, communication protocol development, and network troubleshooting.It is used to track the packets so that each one is filtered to meet our specific needs. It is commonly called as a sniffer, network protocol analyzer, and network analyzer. It is also used by network security engineers to examine security problems.

Wireshark is a free to use application which is used to apprehend the data back and forth. It is often called as a free packet sniffer computer application. It puts the network card into an unselective mode, i.e., to accept all the packets which it receives.

Wireshark can be used in the following ways:

- It is used by network security engineers to examine security problems.
- It allows the users to watch all the traffic being passed over the network.
- It is used by network engineers to troubleshoot network issues.
- It also helps to troubleshoot latency issues and malicious activities on your network.
- It can also analyze dropped packets.
- It helps us to know how all the devices like laptop, mobile phones, desktop, switch, routers, etc., communicate in a local network or the rest of the world.

Wireshark is similar to tcpdump in networking. Tcpdump is a common packet analyzer which allows the user to display other packets and TCP/IP packets, being transmitted and received over a network attached to the computer. It has a graphic end and some sorting and filtering functions. Wireshark users can see all the traffic passing through the network.

Wireshark can also monitor the unicast traffic which is not sent to the network's MAC address interface. But, the switch does not pass all the traffic to the port. Hence, the promiscuous mode is not sufficient to see all the traffic. The various network taps or port mirroring is used to extend capture at any point.

Port mirroring is a method to monitor network traffic. When it is enabled, the switch sends the copies of all the network packets present at one port to another port.

Features of Wireshark

- It is multi-platform software, i.e., it can run on Linux, Windows, OS X, FreeBSD, NetBSD, etc.
- It is a standard three-pane packet browser.
- It performs deep inspection of the hundreds of protocols.
- It often involves live analysis, i.e., from the different types of the network like the Ethernet, loopback, etc., we can read live data.
- It has sort and filter options which makes ease to the user to view the data.
- It is also useful in VoIP analysis.
- It can also capture raw USB traffic.
- Various settings, like timers and filters, can be used to filter the output.
- It can only capture packet on the PCAP (an application programming interface used to capture the network) supported networks.

- Wireshark supports a variety of well-documented capture file formats such as the PcapNg and Libpcap. These formats are used for storing the captured data.
- It is the no.1 piece of software for its purpose. It has countless applications ranging from the tracing down, unauthorized traffic, firewall settings, etc.

PROCEDURE

Below are the steps to install the Wireshark software on the computer:

- Open the web browser.
- Search for 'Download Wireshark.'
- Select the Windows installer according to your system configuration, either 32-bt or 64-bit. Save the program and close the browser.
- Now, open the software, and follow the install instruction by accepting the license.

The Wireshark is ready for use.

On the network and Internet settings option, we can check the interface connected to our computer.If you are Linux users, then you will find Wireshark in its package repositories.By selecting the current interface, we can get the traffic traversing through that interface. The version used here is 3.0.3. This version will open as:

The Wireshark software window is shown above, and all the processes on the network are carried within this screen only. The options given on the list are the Interface list options. The number of interface options will be present. Selection of any option will determine all the traffic. For example, from the above fig. select the Wi-Fi option. After this, a new window opens up, which will show all the current traffic on the network. Below is the image which tells us about the live capture of packets and our Wireshark will look like:

The above arrow shows the packet content written in hexadecimal or the ASCII format. And the information above the packet content, are the details of the packet header.

It will continue listening to all the data packets, and you will get much data. If you want to see a particular data, then you can click on the red button. The traffic will be stationary, and you can note the parameters like time, source, destination, the protocol being used, length, and the Info. To view in-depth detail, you can click on that particular address; a lot of the information will be displayed below that.

The screen/interface of the Wireshark is divided into five parts:

- ✓ First part contains a menu bar and the options displayed below it. This part is at the top of the window. File and the capture menus options are commonly used in Wireshark. The capture menu allows to start the capturing process. And the File menu is used to open and save a capture file.
- ✓ The second part is the packet listing window. It determines the packet flow or the captured packets in the traffic. It includes the packet number, time, source, destination, protocol, length, and info. We can sort the packet list by clicking on the column name.
- ✓ Next comes the packet header- detailed window. It contains detailed information about the components of the packets. The protocol info can also be expanded or minimized according to the information required.
- ✓ The bottom window called the packet contents window, which displays the content in ASCII and hexadecimal format.

- ✓ At last, is the filter field which is at the top of the display. The captured packets on the screen can be filtered based on any component according to your requirements. For example, if we want to see only the packets with the HTTP protocol, we can apply filters to that option. All the packets with HTTP as the protocol will only be displayed on the screen, shown below:

  You can also select the connection to which your computer is connected. For example, in this PC, we have chosen the current network, i.e., the ETHERNET.

  After connecting, you can watch the traffic below:

  In view option on the menu bar, we can also change the view of the interface. You can change the number of things in the view menu. You can also enable or disable any option according to the requirements.

  There is a filter block below the menu bar, from where a large amount of data can be filtered. For example, if we apply a filter for HTTP, only the interfaces with the HTTP will be listed.

  If you want to filter according to the source, right-click on the source you want to filter and select 'Apply as Filter' and choose '...and filter.'

  Steps for the permanent colorization are: click on the 'View' option on the menu bar and select 'Coloring Rules.' The table will appear like the image shown below:

  For the network administrator job, advanced knowledge of Wireshark is considered as the requirements. So, it is essential to understand the concepts of the software. It contains these 20 default coloring rules which can be added or removed according to the requirements. Select the option 'View' and then choose 'Colorize Packet List,' which is used to toggle the color on and off. Whenever we type any commands in the filter command box, it turns green if your command is correct. It turns red if it is incorrect or the Wireshark does not recognize your command.

Wireshark Linux

```
  $ sudo apt-get install wireshark
$ sudo dpkg-reconfigure wireshark-common

$ sudo usermod -a -G wireshark $USER

$ newgrp wireshark
```

Once you have completed the above steps, you then log out and log back in, and then start Wireshark:

```
$ wireshark
```

Wireshark tries to help you identify packet types by applying common-sense color coding.

In Wireshark, just go to Statistics >> I/O Graph, and you'll see a graph similar to the one below:

Figure : Viewing the input/output traffic graph in Wireshark

This particular graph is showing typical traffic generated by a home office. The spikes in the graph are bursts of traffic that were caused by generating a Distributed Denial of Service (DDoS) attack using a few Linux systems.

In this case, three major traffic bursts were generated. Many times, cybersecurity pros use Wireshark as a quick and dirty way to identify traffic bursts during attacks.

It turned out that the client didn't know this device was even on the network. Thus, it was removed, helping to make the network a bit more secure. Notice, also, that this network connection is experiencing a lot of traffic to Amazon (administering a server in AWS at the time) and Box.com (using Box for system backup at the time).

In some cases, it is even possible to use Wireshark to identify the geographic location of source and destination traffic. If you click on the Map button at the bottom of the screen , Wireshark will show you a map, providing its best guess of the location of the IP addresses you've identified.

Figure: Viewing geographic estimations in Wireshark

Because IPv4 addresses can be easily spoofed, you can't rely completely on this geographical information. But it can be fairly accurate.

You can apply Wireshark filters in two ways:

1. In the Display Filter window, at the top of the screen
2. By highlighting a packet (or a portion of a packet) and right-clicking on the packet

Valid filter rules are always colored green. If you make a mistake on a filter rule, the box will turn a vivid pink. ip.addr == 18.224.161.65

Figure: Applying a filter to a capture in Wireshark

Alternatively, you can highlight the IP address of a packet and then create a filter for it.
Once you select the IP address, right-click, and then select the Apply As Filter option.

You'll then see a menu of additional options. One of those is called Selected. If you choose Selected, then Wireshark will create a filter that shows only packets with that IP address in it.

You're not limited to just IPv4 addresses. For example, if you want to see if a particular computer is active and using an IPv6 address on your network, you can open up a copy of Wireshark and apply the following rule:

ipv6.dst == 2607:f8b0:400a:15::

STEGOSUITE

AIM: To familiarize stegnography tools in linux

DESCRIPTION

With the rapid growth of network bandwidth and digital-communication techniques, the Internet is available to the majority of the population and it also becomes the common channel for transmitting many documents—for instance, video, image, text, and audio (in digital form). Many practices and methods have been developed to make the transmission of data more secure. The problem is that the focus of the current research is mainly on the better designing of data-hiding techniques used for transmitting secret data where digital images are selected as the cover-media. Unlike the other forms of communication, the process of steganography is defeated when the communication between sender and receiver is detected. Therefore, the first requirement for a good steganographic structure is it is undetectable.

Hiding secret information in Audio and Images: Many methods for hiding information in audio and images exist. These methods may include hiding information in unused space in file headers to hold "extra" information. Embedding techniques can range from the placement of indistinguishable information level, sometimes it includes manipulation of compression algorithms, and to the extent of modification of properties of the carrier. In audio file Steganography, small echoes or slight delays can be added or indistinct signals can be masked by sounds of higher amplitude.

Stegosuite: Stegosuite is a graphical steganography tool (this is the main difference between Stegosuite and Stegohide). It is used to hide secret data or information in image files. Stegosuite provides the facility of embedding text messages and multiple files of any type. To make the process of embedding more secure, the embedded data is encrypted using AES (Advanced Encryption Standard). Currently, the Stegosuite tool supports BMP, GIF, JPG, and PNG file types.

PROCEDURE

Installing Stegosuite: To install the Stegosuite tool in Kali Linux follow the below commands.

sudo apt-get update -y

// Execute above command first then execute

// next command to avoid Archives

// Installation Error sudo apt-get install stegosuite

Now after complete execution of the above commands. To run Stegosuite simply type "stegosuite" in terminal.

stegosuite

Embed data:

1. *stegosuite* command will open the Stegosuite window.
2. Click on the file in the Stegosuite window to select the image file.
3. Type the secret message or select the text files you want to embed in the image.
4. Choose any password and after that click on the Embed button.

Extract Data:

1. Click on the file in Stegosuite window to select the stego file you want to extract hidden information from
2. Type the password and click the extract button
3. The hidden data will be extracted

Type your Password and click on Extract. The hidden message and text files will be visible now.

HTTRACK

AIM:

To familiarize the cyber forensic tool httrack

DESCRIPTION:

HTTrack is an easy-to-use offline browser utility. It allows you to download a World Wide Website from the Internet to a local directory, building recursively all directories, getting html, images, and other files from the server to your computer. HTTrack arranges the original site's relative link-structure. Simply open a page of the "mirrored" website in your browser, and you can browse the site from link to link, as if you were viewing it online. HTTrack can also update an existing mirrored site, and resume interrupted downloads. HTTrack is fully configurable, and has an integrated help system.

WinHTTrack (Windows release of HTTrack) and WebHTTrack (Linux/Unix release of HTTrack) are very similar, but not exactly identical. You may encounter minor differences (in the display, or in various options) between these two releases. The engine behind these two release is identical.

PROCEDURE:

Step 1: Choose a project name and destination folder

- Change the destination folder if necessary

  It is more convenient to organize all mirrors in one directory, for example My Web Sites.If you already have made mirrors using HTTrack, be sure

Select a new project name OR select an existing project for update/retry

Click on the NEXT button
Step 2 : Fill the addresses

Select an action,the default action is Download web sites

- ✓ *Download web site(s)* Will transfert the desired sites with default options

Download web site(s) + questions Will transfert the desired sites with default options, and ask questions if any links are considered as potentially downloadable

Get individual files Will only get the desired files you specify (for example, ZIP files), but will not spider through HTML files

Download all sites in pages (multiple mirror) Will download all sites that appears in the site(s) selected. If you drag&drop your boormark file, this option lets you mirror all your favorite sites

Test links in pages (bookmark test) Will test all links indicated. Useful to check a bookmark file

Continue interrupted download Use this option if a download has been interrupted (user interruption,crash..)

Update existing download Use this option to update an existing project. The engine will recheck the complete structure, checking each downloaded file for any updates on the web site.

Enter the site's addresses :You can click on the *Add a URL* button to add each address, or just type them in the box

You may define options by clicking on the **Set options** button.You can define filters or download parameters in the option panel. You may also add a URL by clicking on the **Add a URL** button.This option lets you define additional parameters (login/password) for the URL, or capture a complex URL from your browser. Click on the NEXT button.

Step 3 : Ready to start

If you want, you may connect immediately or delay the mirrorIf you don't select anything, HTTrack will assume that you are already connected to the Internet and that you want to start the mirror action now .Connect to this provider, you can select here a specific provider to connect to when begining the mirror if you are not already connected to the Internet. Disconnect when finished ,Click on this checkbox to ask httrack to disconnect the network when mirror is finished. Shutdown PC when finished,Click on this checkbox to ask httrack to shutdown your computer when mirror is finished. On Hold,you can enter here the time of the mirror start. You can delay up to 24 hours a mirror using this feature.

Click on the FINISH button

Step 4 : Wait

Wait until the mirror is finishing,You can cancel at any time the mirror, or cancel files currently downloaded for any reasons (file too big, for example)Options can be changed during the mirror: maximum number of connections, limits...

Step 5 : Check the result ,You may check the error log file, which could contain useful information if errors have occurred.

Netstat Command in Linux

Linux netstat command stands for **Network statistics**. It displays information about different interface statistics, including open sockets, routing tables, and connection information. Further, it can be used to displays all the socket connections (including TCP, UDP). Apart from connected sockets, it also displays the sockets that are pending for connections. It is a handy tool for network and system administrators.

In computing, the netstat command is a command-line network utility that shows network connections for TCP (both outgoing and incoming), several network interfaces (softwaredefined network interface or network interface controller), network protocol statistics, and routing tables. It's available on Plan 9, Unix, Inferno, and Unix-like OSes, including BSD, Solaris, Linux, and macOS. Also, it's available on IBM OS/2 and Microsoft Windows NT-based OSes, including Windows 10, Windows 8, Windows 7, Windows Vista, and Windows XP.

In the network, t is used to find problems and determine the traffic as a performance measurement. This program is almost obsolete on Linux, although still added in several distributions.

The netstat command is superseded by ***"s"*** on Linux. The ip route command is the replacement of the netstat -r command, and the ip maddr command is the replacement of the netstat -i command, each of which is suggested instead. Netstat gives statistics for belowࠤ **Proto:** Proto is the protocol name (UDP or TCP). **Local Address:** Local Address specifies the local computer's IP address and port number being utilized. The local computer's name related to the

IP address and the port name is displayed unless the parameter, i.e., -n, is mentioned. An asterisk (*) is displayed for the host when the server is active (listening) in every interface. The port number is displayed as an asterisk if the port isn't established yet.

Foreign Address: The port number and IP address of the remote computer to which a socket is linked. The names that relate to the IP address and all ports are displayed unless the parameter, i.e., -n, is mentioned. The port number is displayed as an asterisk if the port isn't established yet.

State: Represents the TCP connection state. Several possible states are available, including TIME_WAIT, SYN_SEND, SYN_RECEIVED, LISTEN, LAST_ACK, FIN_WAIT_2, FIN_WAIT_1, ESTABLISHED, CLOSED, and CLOSE_WAIT.

Some of the netstat commands and their description are mentioned below:

Command	Description
netstat -a	Represents every socket, both non-listening and listening, and every protocol, like UDP, TCP, etc.
netstat -at	Represents TCP connections only (-au represents UDP connections only).
netstat -ant	Represents every TCP connection without DNS resolution (rather than displays IP addresses).
netstat -al	Displays listening sockets only.
netstat -aep	Displays PID and to which function all sockets belong; e includes extra information like the user. Execute as root to check every PID.

netstat -s > file2.txt	Displays network statistics.
netstat -i	Shows a table of every network interface. Include -e to receive the result, which is the same as ifconfig.
netstat -r	Displays the information on kernel routing. It is a similar result as route -e.
netstat -ct	Shows TCP connections regularly.
netstat -g	Shows the information of multicast group membership for IPv6 and IPv4.
netstat -atnp | grep ESTA	Shows every "established" connection of TCP currently.
netstat -lntu	Shows every service listening UDP and TCP, every free open port over the local device.

Syntax:

The netstat command supports various command-line options. The basic syntax of the netstat command is as follows:

netstat

It supports multiple command-line options to print information about the Linux networking subsystem. The output is controlled by the first argument. Let's see the list of the first arguments:

(none): If no option is specified, it will execute the default command that displays a list of open sockets of all configured address families.

--route, -r: It is used to print the kernel routing tables. The "netstat -r" command and "route e" command will produce the same output.

--groups, -g: It is used to display multicast group membership information different IP versions (Ipv4 and IPV6).

--interfaces, -i: It is used to display all network interfaces.

--masquerade, -M: It displays masqueraded connections.

--statistics, -s: This option displays the summary statistics for each protocol.

--verbose, -v: It is used to display the detailed output. It is a handy tool for displaying the details about unconfigured address families.

--wide, -W: It is used as an output not to reduce the IP address as necessary. It is still optional not to break existing scripts.

--numeric-hosts: It is used to display numerical host addresses; it does not affect the resolution of port or user names.

--numeric-ports: It is used to display numerical port numbers, it does not affect the properties and objects of host or user names.

--numeric-users: It is used to display numeric user Ids, it does not affect the resolution of host or port names.

--numeric, -n: It is used to display numeric addresses alternatively defining symbolic hosts, ports, or usernames.

--protocol=family, -A: It is used to specify the address families for which connections are to be displayed. The address families are a comma (',') separated like Inet, inet6, Unix, ax25, Netrom, Econet, Ipx, DDP, and Bluetooth.

-c, --continuous: It is used to display the selected information continuously for every second.

-e, --extend: It is used for extended output. This option can be used twice for maximum detail.

-o, --timers: It is used to include networking timers related information.

-p, --program: It is used to display the PID and name of the process to the corresponding sockets.

-l, --listening: It is used to display only listening sockets.

-a, --all: It is used to display both sockets (i.e., listening and non-listening). By specifying the '--interfaces' option, we can list the interfaces that are not up.

-F: It is used to display the routing information from the FIB.

-C: It is used to display the routing information from the route cache.

Installation of the netstat command

If the netstat command is not installed on your machine, it will display the traditional Linux installation error message "Command 'netstat' not found."

To install it, execute the below command:

sudo apt install net-tools

○ Display All Connections ○ Display only TCP or UDP connections ○ Disable reverse DNS lookup for faster output ○ Display only listening connections ○ Display Pid and Uid ○ Display Statistics ○ Display kernel routing information ○ Display network interfaces ○ Display netstat output continuously ○ Display multicast group information

Display All Connections

The '-a' option is used to display all the existing connections. Execute the netstat command as follows: netstat- a

Display only TCP or UDP Connections

We can list only the TCP or UDP connections. To display only the TCP connection, execute the command with the 't' option as follows:

```
netstat -at
```

To display only UDP connection, execute it with 'u' option as follows:

```
netstat -au
```

Disable Reverse DNS Lookup for Faster Output

The default behavior of the netstat command finds out the hostname for each IP address by a reverse DNS lookup. It causes the slowdowns in output. If we don't want to know the hostname, then disable the reverse DNS lookup by suppressing the 'n' option with it: Consider the below command:

```
netstat -ant
```

The above command will disable the reverse DNS lookup and display all the TCP connections.

Display only Listening Connections

The listening connections are such connections that are available for connection requests. Any network process keeps an open port for the listening incoming connection requests. These connections can be listed by executing the below command:

```
netstat -tnl
```

The above command will list all the listening connection for TCP connections

Display Pid and Uid

While checking the network statistics, it is sometimes vital to know the Pid and Uid for a particular connection or user. The Pid and Uid can be listed by executing the 'p' option. Execute the below command:

```
sudo netstat -nlpt
```

The above command will list all the Pid for the TCP connections. It is necessary to execute this command with sudo privilege. Otherwise, it will not display the Pid.

Display Statistics netstat command is also a handy tool for displaying the network statistics such as no packets transmitted and received by a protocol. To display the network statistics, execute the command with the '-s' option as follows: `netstat -s` The above command will display the network statistics

Display kernel Routing Information

The 'r' option is used to display the kernel routing information. It will display the same output as route command. To display the routing information, execute the command as follows:

```
netstat -rn
```

The above command will display the routing information. The 'n' option will disable the hostname lookup.

Display Network Interfaces

We can also display information about the network interfaces by using the netstat command. To display the network interfaces, execute the command with 'i' option as follows:

```
netstat -i
```

The above command will list the network interfaces and related information.

Display netstat Output Continuously

To display the netstat output continuously, execute the command with the 'c' option as follows:

```
netstat -ct
```

The above command will display the TCP connections continuously.

Display Multicast Group Information

The 'g' option is used to display the multicast group information. To print the details for Ipv4 and Ipv6, execute the command as follows:

```
netstat -g
```

The above command will display the multicast group information.

Install Net-Tools in Linux

The netstat command is a part of a package called net-tools. We get the net-tools package using the following command in Ubuntu:

```
$ sudo apt install net-tools
```

Check Netstat Version

Upon the installation process, we can check the installed Netstat version using the following command:

```
$ netstat -v
```

Show Routing Table

The netstat command displays the routing table information on the command line. If we want to check the routing table, we can use the -nr option with Netstat; it will display the kernel routing table in a similar form to that route does. We can run the following command:

```
$ netstat -nr
```

The -nr flag permits Netstat to show addresses separated by dots rather than applying symbolic address titles.

Show Network Connection

The netstat command has a variety of options to view passive and active sockets. Active TCP, Unix, RAW, and UDP socket connections are mentioned by the -x, -w, -u, and -t options, respectively.

We can run the following command:

```
$ netstat -ta
```

Show Network Services

We can execute the below command to see a network list, their related ports, and their current states:

```
$ netstat -pnltu
```

Show every listening port of the UDP and TCP connection

We can see every UDP and TCP port by running the following command in the terminal window:

```
$ netstat -a | more
```

Show every listening connection

We can use the netstat command with the -l option to list every active connection:

```
$ netstat -l
```