Mawlana Bhashani Science and Technology University

Assignment No : 02

Assignment Title: Linux assignment-02

Course Title: Computer Network

Submitted by

Name: Moniruzzaman & Mst.Zakia Sultana

ID:IT-18007 & IT-18027

3rd year 2nd semester

Session: 2017-2018

Dept. of ICT

MBSTU.

Submitted To

Nazrul Islam

Assistant Professor

Dept. of ICT

MBSTU.

1) Introduction

If you have a network that ranges from 192.168.1.0 to 192.168.1.255 explain why individual devices in the network can only be assigned IP addresses in the range of 192.168.1.1 to 192.168.1.254. **(Write down the answers in your written report).**

Answer:

If I have network address from 192.168.1.1 to 192.168.1.255, then we can say that:

The network identifier would be 192.168.1.0-

- An address like 192.168.0.0 becomes unusable for any other purpose after it's established as a network number. If an administrator assigns 192.168.0.0 to any device on the network as a static IP address, the network stops functioning until that device is taken offline. As a network number, this address is used in routing tables and by routers to share network information with each other.

The broadcast address would be 192.168.1.255-

- A broadcast address is a network address used to transmit to all devices connected to a multiple-access communications network. A message sent to a broadcast address may be received by all network-attached hosts. A data packet is transmitted from one point to all users of a messaging network in this way. This occurs with the use of the broadcast address.

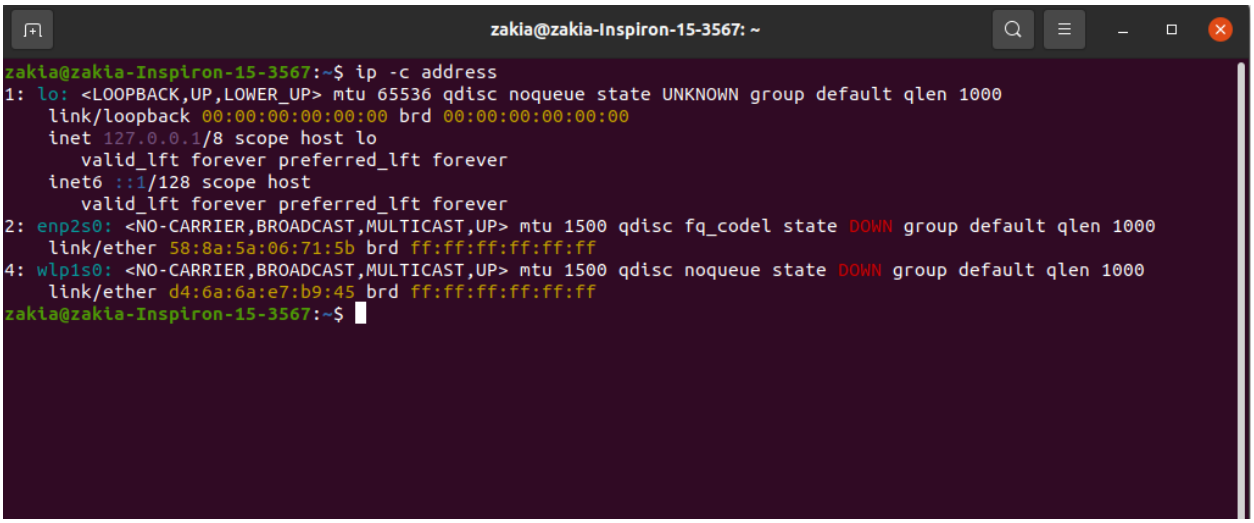
So, valid IP addresses would be 192.168.1.1 to 192.168.1.254.

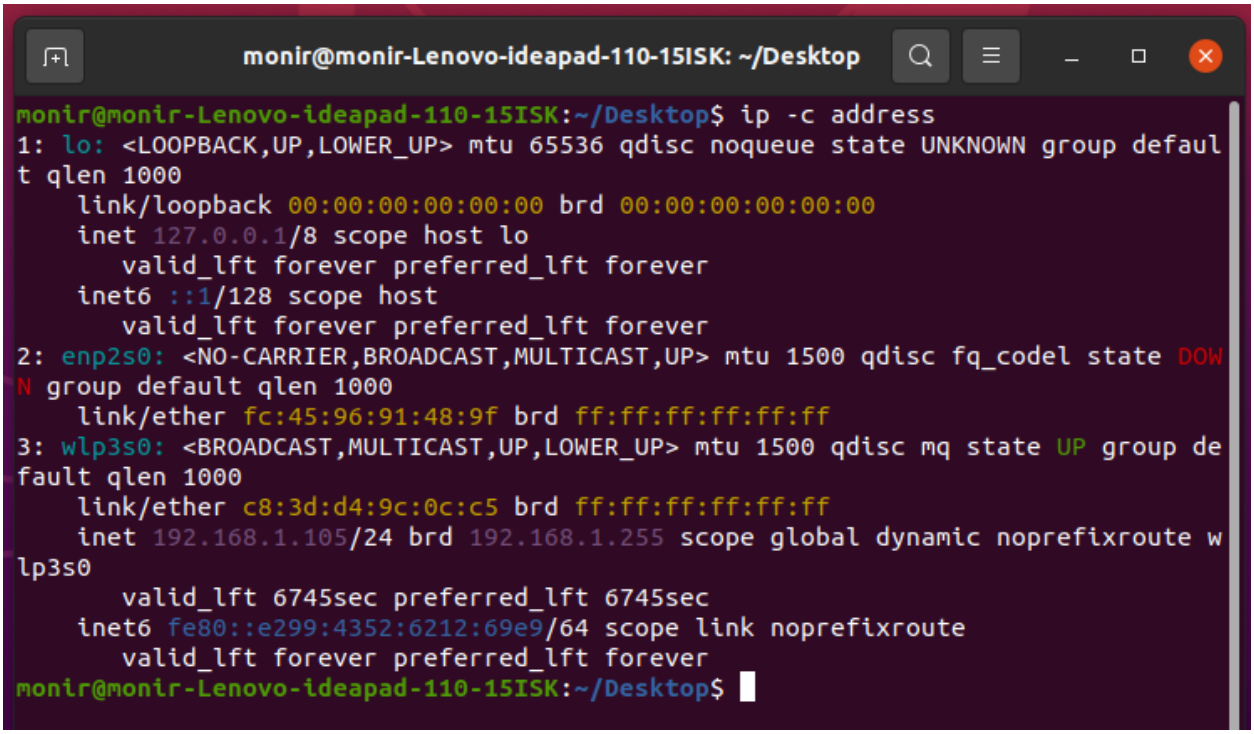
2) Find IP & MAC

Find out about network and hardware information for the computer you are currently using**. (Write down the IP and MAC address of your computer in your written report).**

Answer:

My IP address is 192.168.42.197 and my MAC address is d4:6a:6a:e7:b9:45



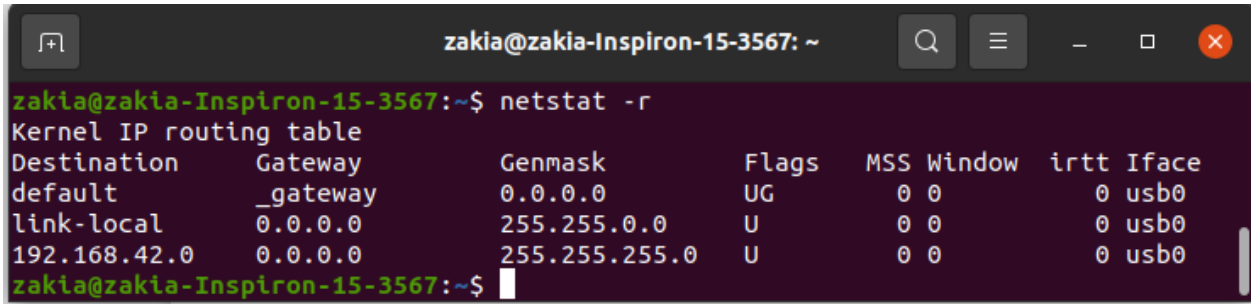


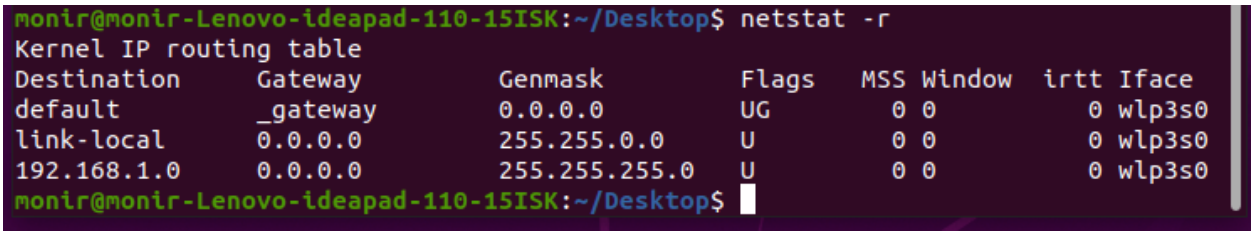
3) Routing Table basics

Now, enter the command: **"$ netstat -r"** to print your computers routing table. Explain (very briefly) the different columns**: Destination, Gateway, Genmask, Flags, MSS, Window, irtt and Iface, (write the answer in your written report in table form).**

Answer:

When a router receives a packet, it examines the destination IP address, and looks up into its Routing Table to figure out which interface packet will be sent out.





Destination: The network ID or destination corresponding to the route.

Gateway: This column indicates the defined gateway for the network. If you see an * in this column, it means that no forwarding gateway is needed for the specified network.

Genmask: The mask that is used to match a destination IP address to the network ID. Flags: The U output in this columns means that the route is up. The G output indicates that specified gateway should be used for this route. D stands for dynamically installed, M stands for modified, and R means reinstated.

MSS: The MSS column indicates the default Maximum Segment Size for TCP connections over this route.

Windows: The Window column indicates the default window size for TCP connections over this route.

Irtt: The Irtt column indicates the Initial Round Trip Time for this route.

Iface: The Iface column shows the network interface. If you had more than one interface, you would see lo (for loopback), eth0 (first Ethernet device), and eth1 (for the second Ethernet device), and so on for the number of interfaces you have installed.

4) Virtual Interfaces

Linux offers the possibility to set up interfaces according to your networking needs. For instance, if needed, you can configure an interface for multiple IP addresses by creating new virtual interfaces with another IP address.

a) Create a new virtual interface with following IP address, 192.168.2.32 an netmask 255.255.255.0 then check to see if the interface was created successfully? **(save a print screen of your interface table to display in your written report document).**

b) Now, you need to set up a route for this interface so that your computercan see it. Otherwise, everyone else on the network will be able to reach the new interface except you. Issue the needed command, then issue the "$ netstat" --r" command and check if the route to your added interface is visible **(save a print screen of your routing table to display in your written report document together with the command(s) you used to set up the interface).**

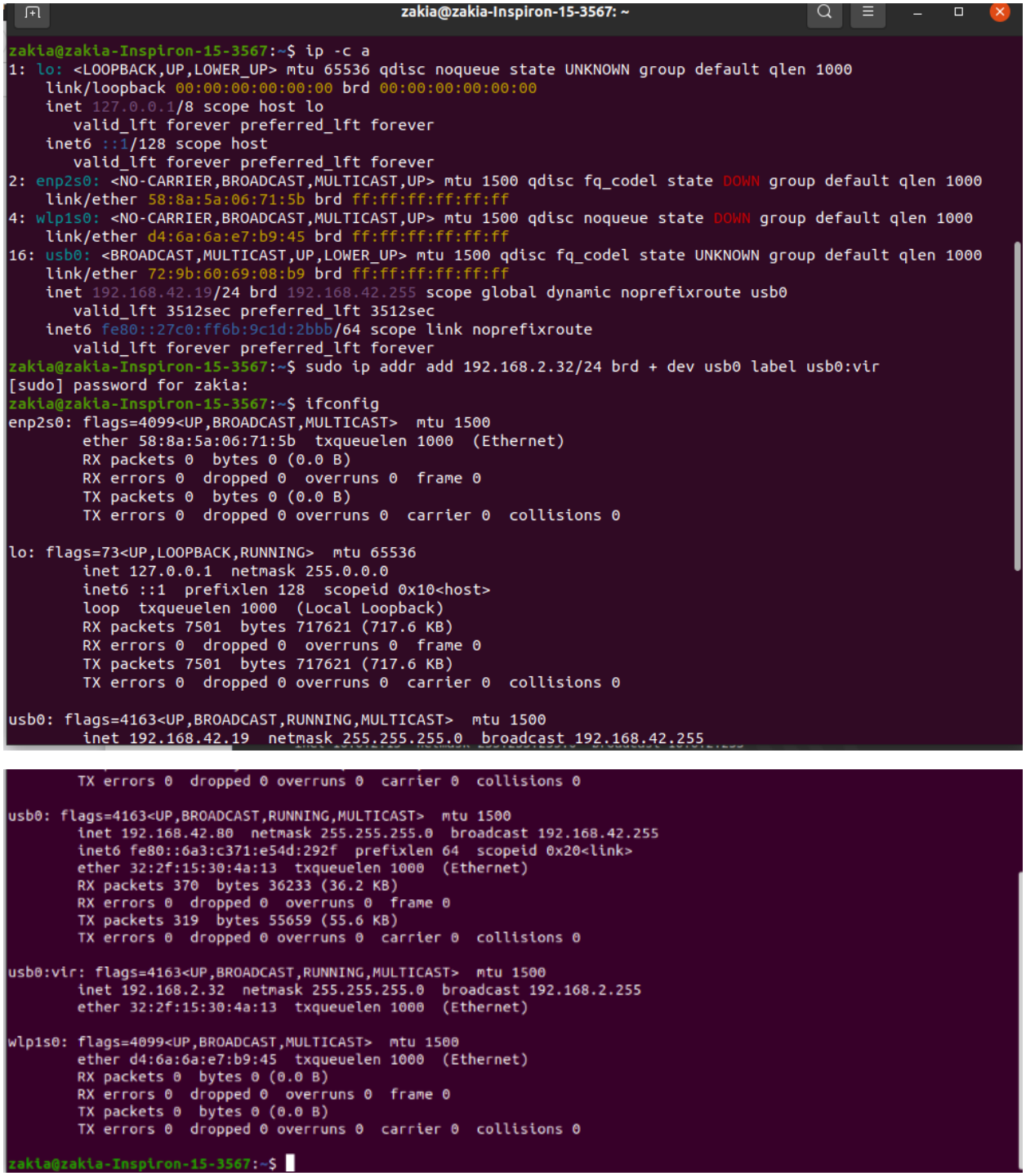
c) Next remove the route for this interface, **(write down the command(s) in your written report).**

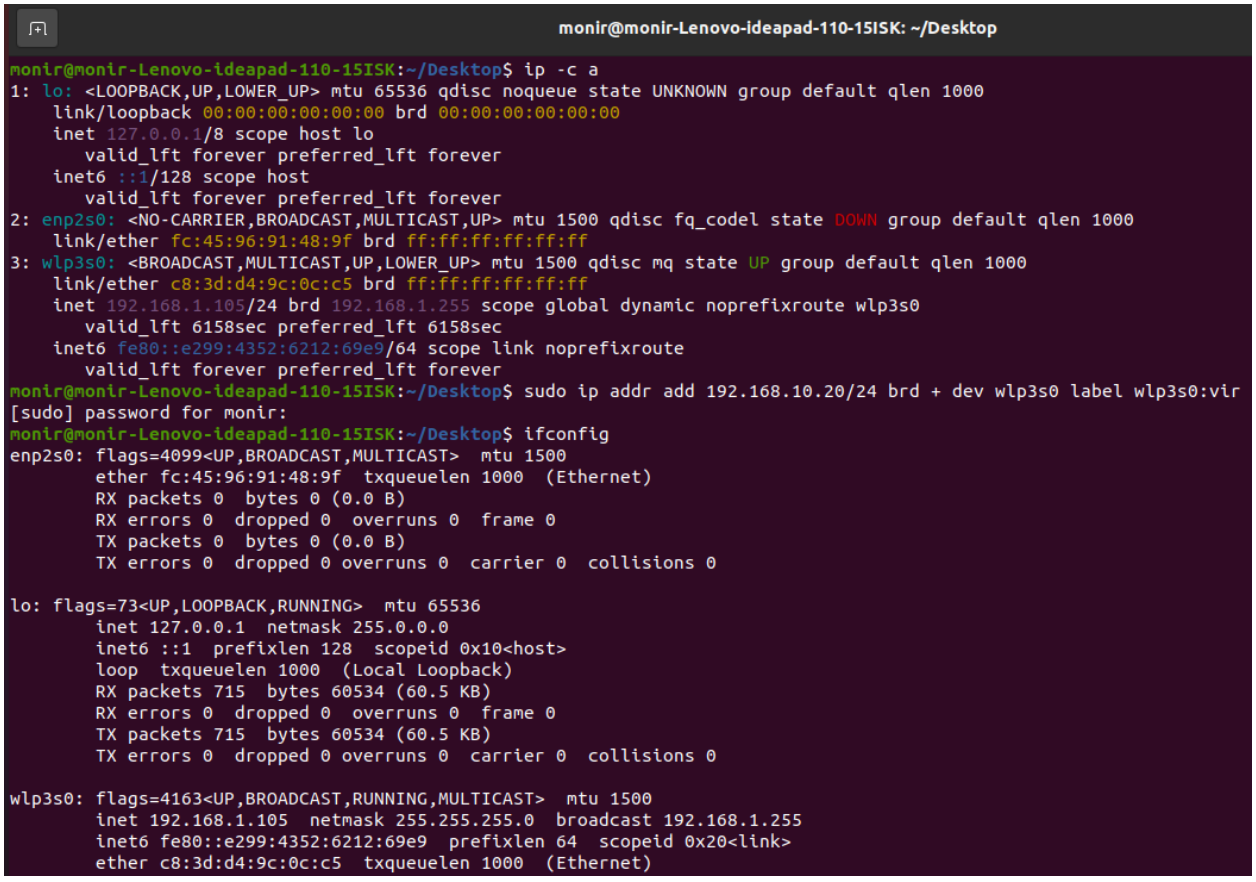
d) Then remove the interface completely, **(write down the command(s) in your written report).**

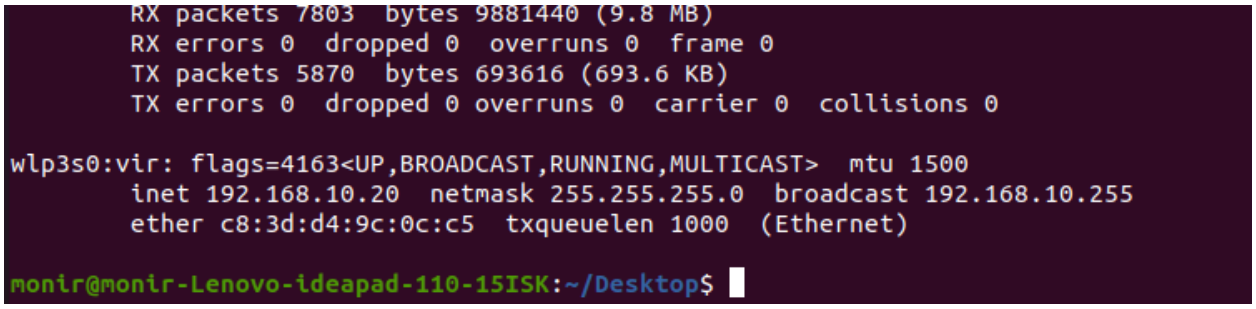
Answer:

a) **Creating Virtual Interface :**

sudo ip addr add 192.168.2.32/24 brd + dev usb0 label usb0:vir

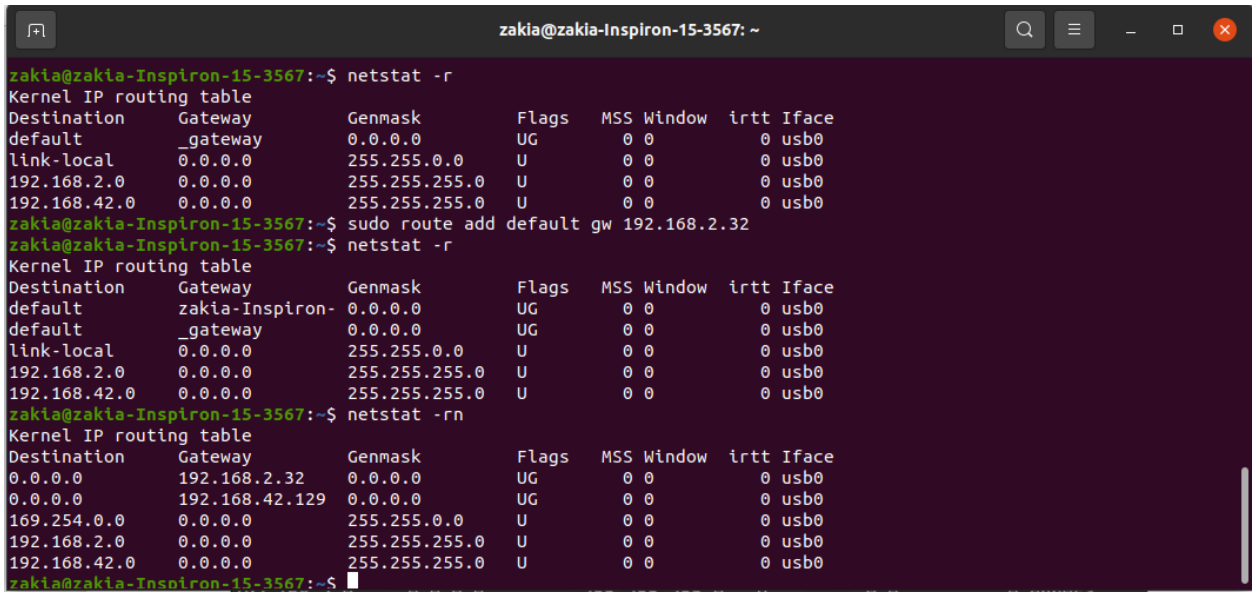

zakia@zakia-Inspiron-15-3567: ~
zakia@zakia-Inspiron-15-3567:~$ ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: enp2s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
link/ether 58:8a:5a:06:71:5b brd ff:ff:ff:ff:ff:ff
4: wlp1s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
link/ether d4:6a:6a:e7:b9:45 brd ff:ff:ff:ff:ff:ff
16: usb0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 1000
link/ether 72:9b:60:69:08:b9 brd ff:ff:ff:ff:ff:ff
inet 192.168.42.19/24 brd 192.168.42.255 scope global dynamic noprefixroute usb0
valid_lft 3512sec preferred_lft 3512sec
inet6 fe80::27c0:ff6b:9c1d:2bbb/64 scope link noprefixroute
valid_lft forever preferred_lft forever
zakia@zakia-Inspiron-15-3567:~$ sudo ip addr add 192.168.2.32/24 brd + dev usb0 label usb0:vir
[sudo] password for zakia:
zakia@zakia-Inspiron-15-3567:~$ ifconfig
enp2s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether 58:8a:5a:06:71:5b txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 7501 bytes 717621 (717.6 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 7501 bytes 717621 (717.6 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.42.19 netmask 255.255.255.0 broadcast 192.168.42.255
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.42.80 netmask 255.255.255.0 broadcast 192.168.42.255
inet6 fe80::6a3:c371:e54d:292f prefixlen 64 scopeid 0x20<link>
ether 32:2f:15:30:4a:13 txqueuelen 1000 (Ethernet)
RX packets 370 bytes 36233 (36.2 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 319 bytes 55659 (55.6 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
usb0:vir: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.2.32 netmask 255.255.255.0 broadcast 192.168.2.255
ether 32:2f:15:30:4a:13 txqueuelen 1000 (Ethernet)
wlp1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether d4:6a:6a:e7:b9:45 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
zakia@zakia-Inspiron-15-3567:~$

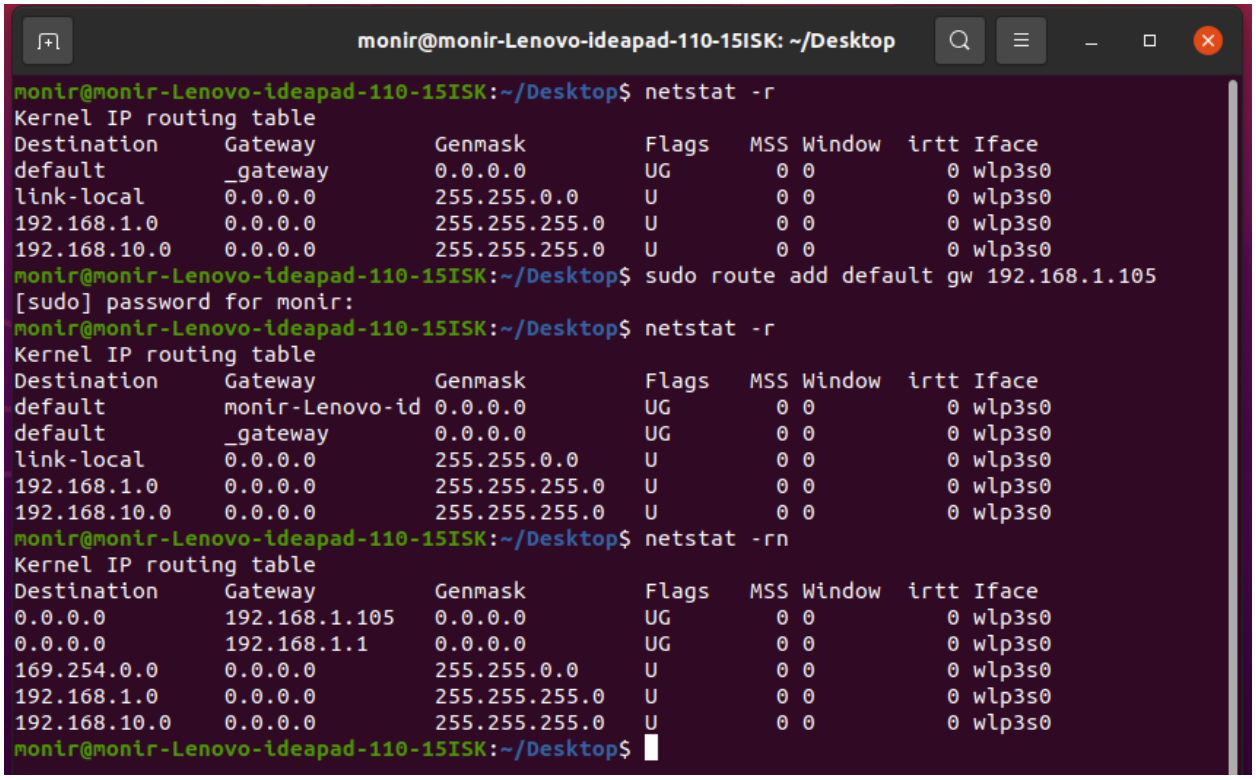




b) Inserting Interface in the Routing Table:

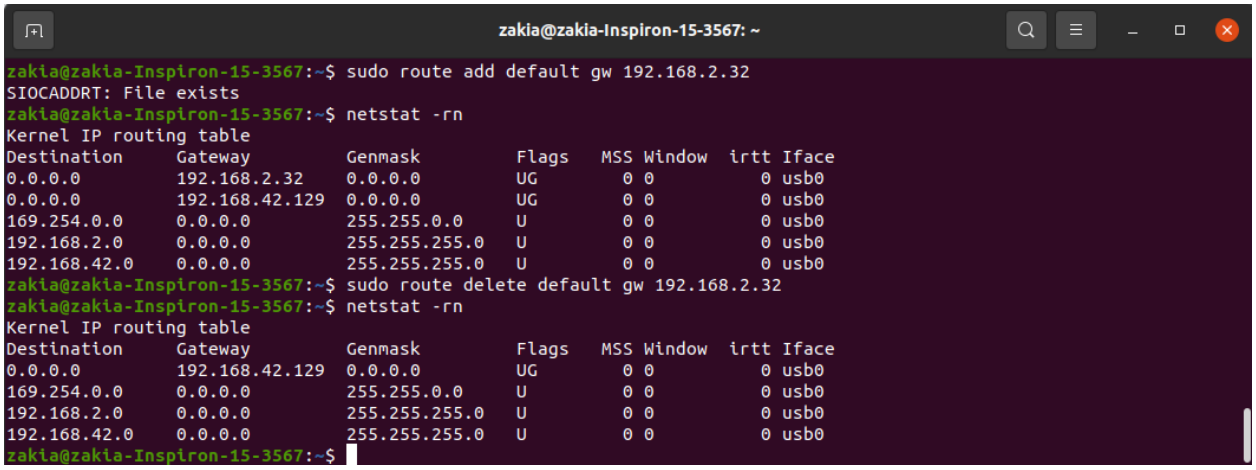
sudo route add default gw 192.168.2.32

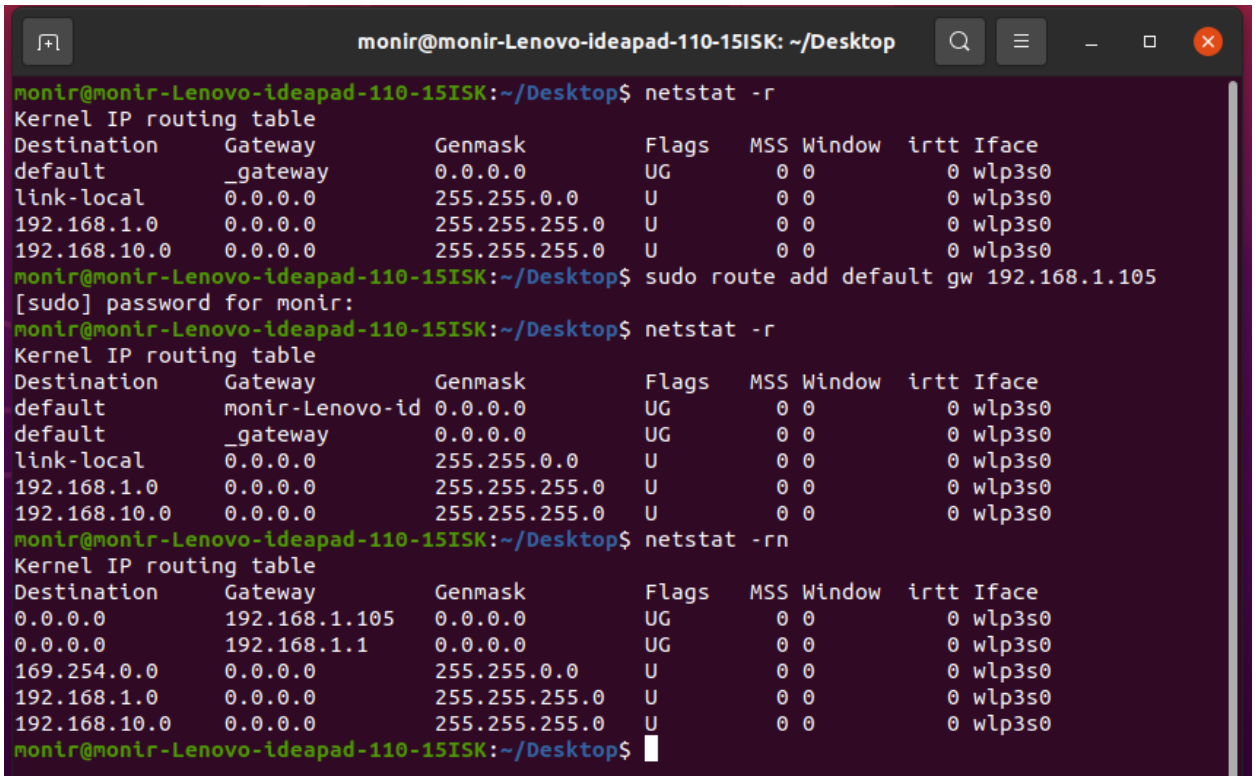




c) Deleting Interface from Routing table:

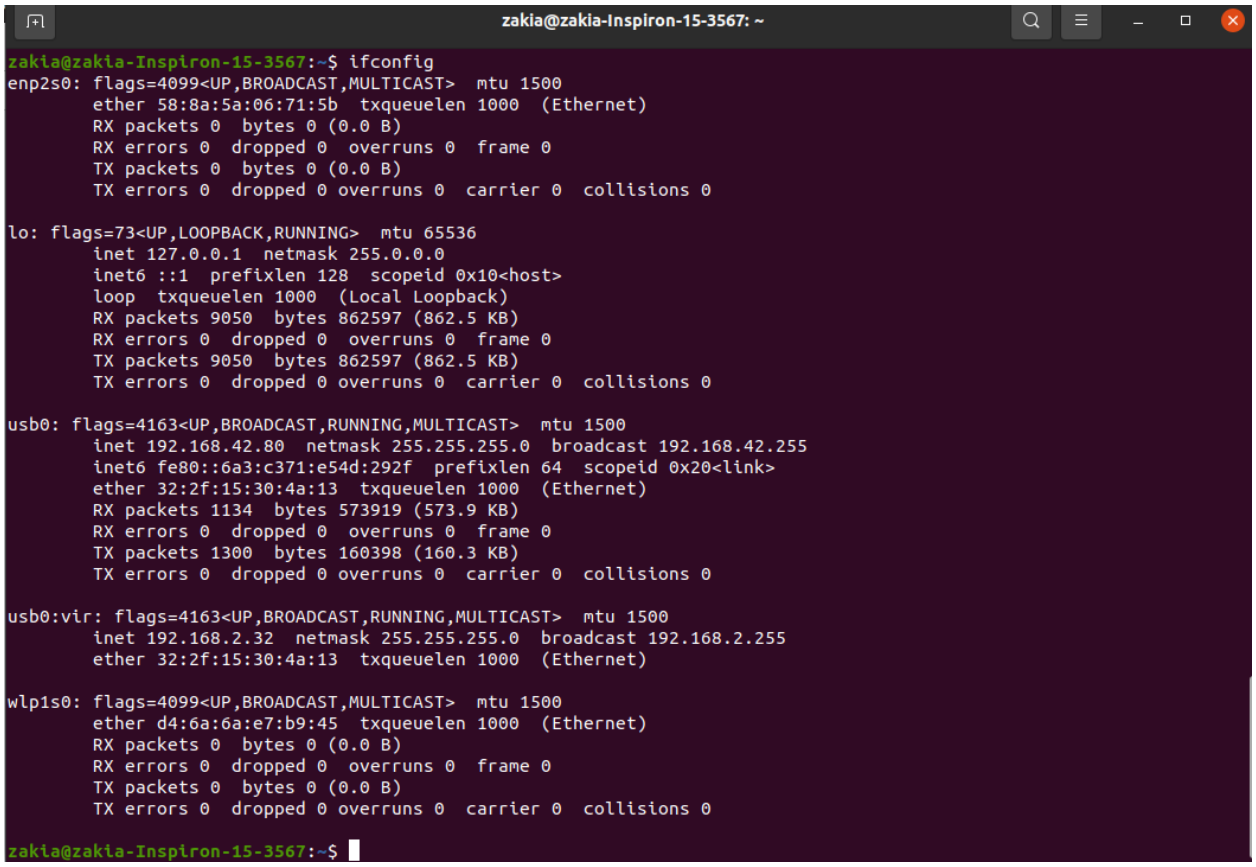
sudo route delete default gw 192.168.2.32

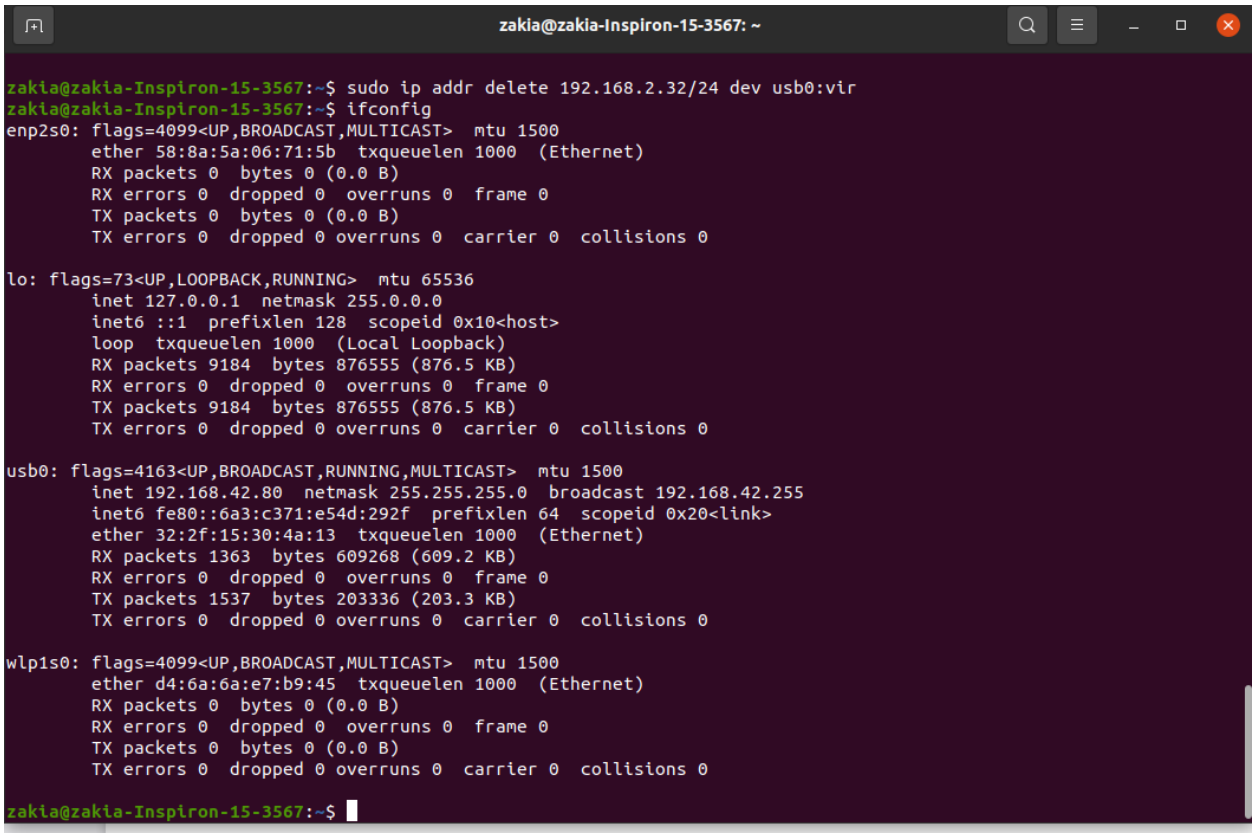


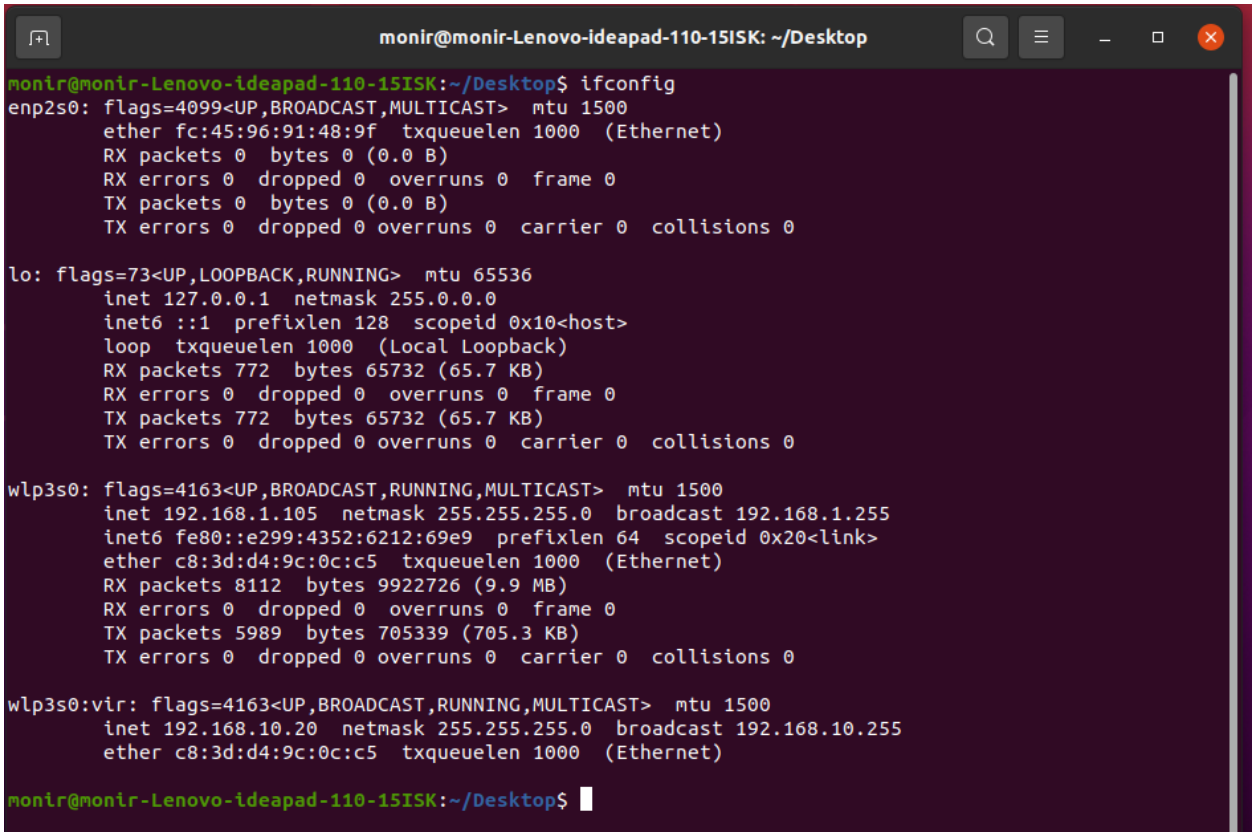


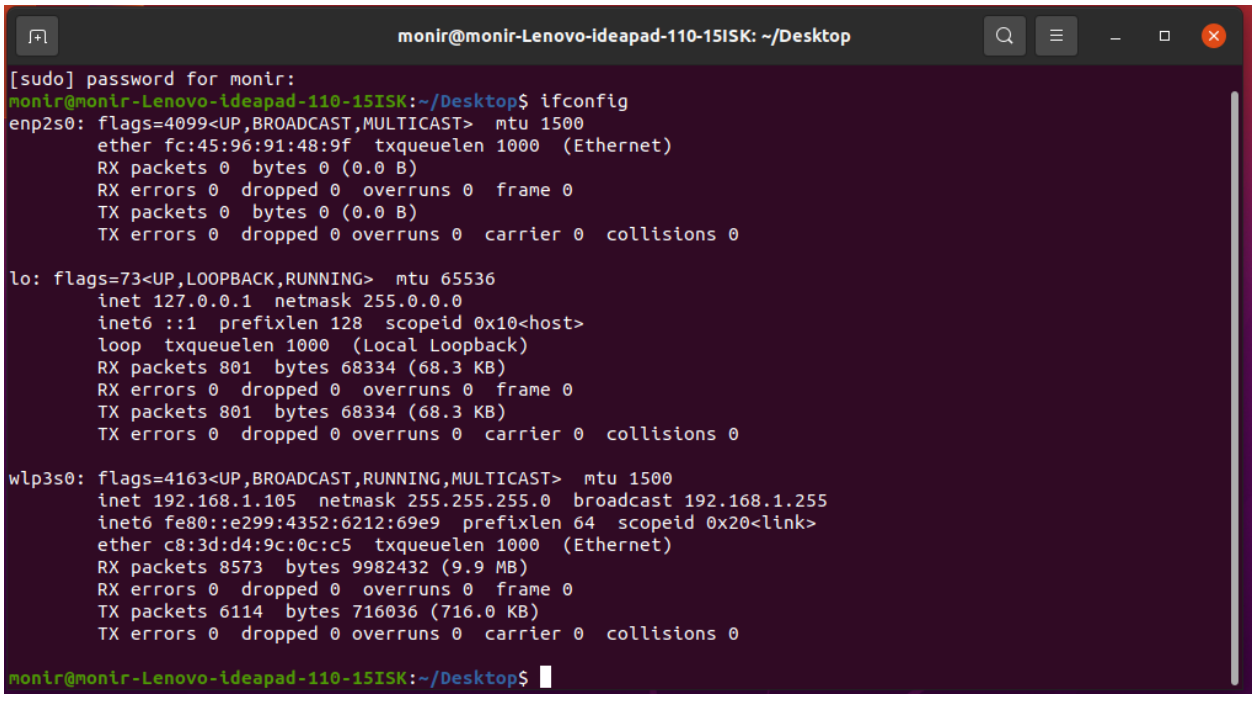
d) Deleting Virtual Interface:

sudo ip addr delete 192.168.2.32/24 dev usb0:vir


zakia@zakia-Inspiron-15-3567: ~
zakia@zakia-Inspiron-15-3567:~$ ifconfig
enp2s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether 58:8a:5a:06:71:5b txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 9050 bytes 862597 (862.5 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 9050 bytes 862597 (862.5 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.42.80 netmask 255.255.255.0 broadcast 192.168.42.255
inet6 fe80::6a3:c371:e54d:292f prefixlen 64 scopeid 0x20<link>
ether 32:2f:15:30:4a:13 txqueuelen 1000 (Ethernet)
RX packets 1134 bytes 573919 (573.9 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 1300 bytes 160398 (160.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
usb0:vir: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.2.32 netmask 255.255.255.0 broadcast 192.168.2.255
ether 32:2f:15:30:4a:13 txqueuelen 1000 (Ethernet)
wlp1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether d4:6a:6a:e7:b9:45 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
zakia@zakia-Inspiron-15-3567:~$


zakia@zakia-Inspiron-15-3567: ~
zakia@zakia-Inspiron-15-3567:~$ sudo ip addr delete 192.168.2.32/24 dev usb0:vir
zakia@zakia-Inspiron-15-3567:~$ ifconfig
enp2s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether 58:8a:5a:06:71:5b txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 9184 bytes 876555 (876.5 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 9184 bytes 876555 (876.5 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.42.80 netmask 255.255.255.0 broadcast 192.168.42.255
inet6 fe80::6a3:c371:e54d:292f prefixlen 64 scopeid 0x20<link>
ether 32:2f:15:30:4a:13 txqueuelen 1000 (Ethernet)
RX packets 1363 bytes 609268 (609.2 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 1537 bytes 203336 (203.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
wlp1s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether d4:6a:6a:e7:b9:45 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
zakia@zakia-Inspiron-15-3567:~$





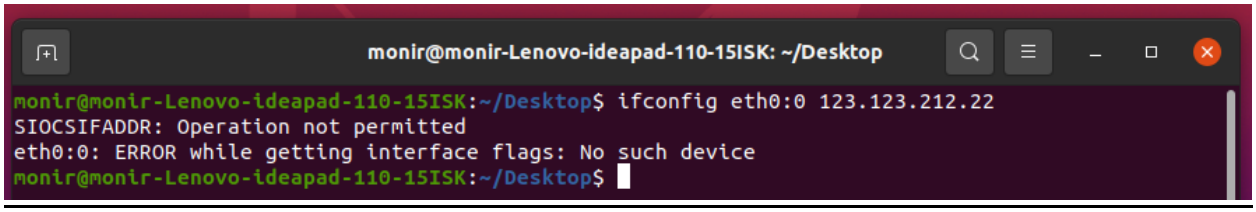
5) Add a New Network

Answer:

The process of creating a virtual network interface in Linux is a quite simple matter. It involves a single execution of the ifconfig command.

ifconfig eth0:0 123.123.22.22

The above command will create a new virtual network interface based on original eth0 physical interface. The only most important condition for creating the virtual network interface is the physical network interface, as in our case eth0 must exists. The whole example is shown below:



There was a problem in my linux platform .There was no such device while I was getting interface flags.So, the operation was not permitted.

6) Multinetwork Scenario Configuration

You should now set up a working routing table for a multi--network scenario. Assume that you have two network cards available connected to two different LANs. The destination of the first network is, 10.0.2.0 with netmask 255.0.0.0 and the second, 192.168.1. with netmask 255.255.255.0. Furthermore, a firewall is assumed to exist between the two networks, where network card eth0 is attached to the 10.0.2.0 network andeth1 is attached to the 192.168.1.0 network. To forward packets on the Internet the firewall needs to route packets from the 10.0.2.0 network through the 192.168.1.0 network. The firewall system must be set up with two IP addresses, 10.0.2.1 on eth0 and 192.168.1.25 on eth1. The gateway to the Internet on the 192.168.1.0 network should be 192.168.1.1.

Provide the necessary commands to route on the firewall/router system:

a) Assign the firewall IP addresses to eth1 and eth2.

b) Add the routesfor the networks, i.e.,192.168.1.0 on eth1 and 10.0.2.0 on eth0

c) Assign the Internet gateway (meaning: 192.168.1.1) as the default gateway.

(Write down the command(s) in your written report)

d) Enter the necessary command(s) in order for packets belonging to computers in the 10.0.2.0 network to be routed to the 192.168.1.0 network and the Internet. In other words this should tell each computer on the 10.0.2.0, which the default gateway is, i.e., your firewall/router. You do not need to worry about the route back configuration it is enough to assign the proper default gateway for the 10.0.2.0 network**. (write down the command(s) in your written report)**

Answer:

Linux easily manages multiple network interface adapters. Laptops typically include both wired and wireless interfaces, and may also support WiMax interfaces for cellular networks. Linux desktop computers also support multiple network interfaces, and you can use your Linux computer as a multi-network client, or as a router for internal networks; such is the case with a couple of my own systems. Every network interface has its own configuration file in the /etc/sysconfig/network-scripts directory. Each interface has a configuration file named ifcfg-X, where X is the number of the interface, starting with zero or 1 depending upon the naming convention in use; for example /etc/sysconfig/networkscripts/ifcfg-eth0 for the first Ethernet interface. Most of the other files in the /etc/sysconfig/network-scripts directory are scripts used to start, stop and perform various network configuration activities.

Each interface configuration file is bound to a specific physical network interface by the MAC address of the interface.

There are many configuration options for the interface configuration files. These are some of the more common options:

• DEVICE: The logical name of the device, such as eth0 or enp0s2.

• HWADDR: The MAC address of the NIC that is bound to the file, such as 00:16:76:02:BA:DB

• ONBOOT: Start the network on this device when the host boots. Options are yes/no. This is typically set to "no" and the network does not start until a user logs in to the desktop. If you need the network to start when no one is logged in, set this to "yes".

• IPADDR: The IP Address assigned to this NIC such as 192.168.0.10

• BROADCAST: The broadcast address for this network such as 192.168.0.255

• NETMASK: The netmask for this subnet such as the class C mask 255.255.255.0

• NETWORK: The network ID for this subnet such as the class C ID 192.168.0.0-

• SEARCH: The DNS domain name to search when doing lookups on unqualified hostnames such as "example.com" BOOTPROTO: The boot protocol for this interface. Options are static, DHCP, bootp, none. The "none" option defaults to static.