Date: 16/7/24 Practical-1

AIM- Study of Various Network Commands used in Linux and windows

Basic Networking Command

1) arp -a

output

Interface: 192.168.1.10 --- 0x11

Internet Address	Physical address	Type
192.168.1.1	00-14-22-01-23	static
192.168.1.2	00-16-17-18-19	dynamic
192.168.1.3	00-19-20-21-22	dynamic

2) hostname

output:

VJSTARK

3) ip config /all

Windows IP Configuration

Host name : VJStark

Primary DNS Suffix . . . :

Node Type . . . . . : Mixed

IP Routing Enable . . . : No

WINS Proxy Enabled . . . . : No

DNS Suffix Search List : hgu-lan

4) nbtStat -a

Output:

Displays protocol statistics and current TCP/IP connections using NBT (Net BIOS over TCP/IP)

5) netStat

Active Connections

Proto	Local Address	Foreign Address	State
TCP	127.0.0.1:49687	VJStaik:49689	Established
TCP	127.0.0.1:49689	VJStaik:49688	Established
TCP	127.0.0.1:49606	VJStaik:49607	Established
TCP	127.0.0.1:49607	VJStaik:49606	Established

6) nslookup

Default Server: multiplay.bsnl.in

Address: 218.248.112.65

7) pathping, Ping, ip

Server: multiplay.bsnl.in

Address: 218.248.112.65

8)

Linux Commands

ip address show

output : link/loopback, up 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
inet 10.255.157.254/32 brd 10.255.255.254
scope global lo valid_lft

ip address add 192.168.1.254/24 dev enp5s03
output
Cannot find device "enp5s03"

ip address del 192.168.1.254/24 dev enp5s03
deleted

ip link set eth0 up
RT Netlink answer: Operation not permitted

ip link set eth0 down, promisc on
Operation not permitted

ip route add default via 192.168.1.254 dev eth0
RT Netlink answer: file exists

ip route add 192.168.1.0/24 via 192.168.1.254
RT Netlink answer: Network is unreachable

9) ip route add 192.168.10.0/24 dev eth0

RTNETLINK answers: File exists

a) ip route delete 192.168.10.0/24 via 192.168.1.254

deleted successfully

10) ip route get 10.10.14.

Cache found

D. ifconfig

Output

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>
mtu 1500

inet 172.19.54.66 netmask 255.255.240.0 broadcast
172.19.63.255

RX packets 33966 bytes 99282425 (99.2 MB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 10176 bytes 1003496 (1.0 MB)

TX errors 0 dropped 0 overruns 0 carrier 0

Collisions 0

2) mtr

a) mtr google.com

s quit Packets Pings:

	Hst	Loss%	Snt	Last	Avg	Best	Wrst	StDev
1)	VJSTARK.mshome.N	0.0%	271	0.9	0.5	02	1.9	0.3
2)	192.168.1.1	0.0%	303	3.3	6.3	1.5	62.9	66
3)	10.0.14.1	4.2%	357	10.1	9.8	1.5	60.1	6.2
4)	10.219.31.13	4.5%	66	17.9	13.3	6.4	166.6	13.1
5)	waiting for replay							
6)	waiting for the replay							

b) mtr -g google.com

error

c) mtr -6 google.com

s quit Packets Pings

	Hst	Loss%	Snt	Last	Avg	Best	Wrst	StDev
1)	VJStark	0.0%	271	0.9	0.5	0.2	1.9	0.3
2)	192.168.1.1	0.0%	303	3.3	6.3	1.5	62.9	6.6
3)	waiting for replay							
4)	waiting for the replay							

9) tcpdump

a) dnf install -y tcpdump

Installed successfully

b) tcpdump -D

output

1. ethO

2. lo

c) tcp dump -i ethO

tcpdump: verbose output suppressed, use -v or -vv for full protocol byte captured

12:34:56.789123 IP 192.168.1.1.http > 192.168.1.100.3[illegible]
Flags [S.], seq 987654321, ack 1234567, win 65[illegible]

d) tcpdum -i ethO -c10

it capture traffic to and from one host

f) tcpdump -i ethO -c10 host 8.8.8.8

pass in traffic

g) tcp dump -i ethO dst host 8.8.8.8

Capture traffic to and from a network

H) tcpdump -i ethO net 10.1.0.0/24

Capture traffic to and from port number

4) tcpdump -i eth0 port 53

. it captures only DNS port 53 traffic

5) Ping

a) Ping google.com

Ping google.com (142.250.205.308) 56(84) bytes of data.
64 bytes from maa05s28-in-f14.1e100.net (142.250.205.238)
icmp_seq=2 ttl=58 time=11.3ms

b) Ping -c 10 google.com

google.com ping statistics
4 packets transmitted, 3 received, 25% packet loss,
time 3068 ms rtt/avg/max/mdev = 9.816/12.336/16.876/3

Configuring an Ethernet connection by using nmcli

If you connect a host the network over Ethernet, you can manage the connections settings on the cmd by using the nmcli utility

Procedure

1) List the network manager connection profiles:

nmcli connections show

Name	UUID	Type	Device
wired connection 1	a5eb6490-cc20-3668	ethernet	enp1s0

2) # nmcli connection add con-name <connection-name>
ifname <device-name> type ethernet
Skip this step to modify an existing profile

3) Optional: Rename the connection profile

nmcli connection modify "wired connection 1"
Here, "wired connection 1" is name of the connection

4) Display the current settings of the connection profile:

nmcli connection show

connection.interface-name	enp1s0
connection.autoconnect	yes
ipv4.method	auto
ipv6.method	auto

5) Configure the IPv4 settings:

* To use DHCP, enter:

nmcli connection modify "wired connection 1"
ipv4.method auto

Skip this step if ipv4.method is already set to auto (default)

* To set a static IPv6 address, network mask, default gateway, DNS servers, and search domain, use:

nmcli connection modify "Wired Connection" ipv6.method manual ipv6.address 2001:db8:1::ffbb/64 ipv6.gateway

b) Configure the IPv6 settings:

* To use Stateless address auto configuration (SLAAC), enter

nmcli connection modify "Wired Connection" ipv6.method auto

nmcli connection "1" ipv6.method manual ipv6.address 2001:db8:1::ffbb ipv6.dns-search example.com

1) Activation the profile

nmcli connection up "Wired Connection"

Verification

1) Display the IP settings of the NIC

ip address show ens160

link/ether 52:54:00:17:68:bb brd ff:ff:ff:ff:ff:ff

2) Display the the IPv6 default gateway

ip route show default

3) Display the IPv6 default gateway

ip -6 route show default

4) Display the DNS settings

cat /etc/resolv.conf

5) Use the ping utility to verify that the host can send packets to other hosts

ping <host name or ip address>

Result

Thus the Linux and windows networking command is verified