LAB Networking

1. List your network interface cards using. *dmesg* reads the kernel ring buffer.

```
[root@centos11 net]# dmesg | grep -i nic
[2.261406] e1000: ens33 NIC Link is Up 1000 Mbps Full Duplex
(snipped)
[2.268399] e1000: ens34 NIC Link is Up 1000 Mbps Full
Duplex,(snipped)
```

If you want to clear the ring buffer, use --clear.

```
[root@centos2 bin]# dmesg --clear
```

You can also list the contents of the net devices in /proc

```
[root@centos11 net]# cat /proc/net/dev
 ens34:  210207    3376    0   85   (snipped)
 ens33:  294253    4338    0   85   (snipped)
```

In earlier releases you could use the ifconfig command which
no longer installed. From CentOS 7 up, you can use the
ip command.With just the ip command you can manage your links
and other things.

```
[root@centos11 net]# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
     link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
     pfifo_fast state UP mode DEFAULT qlen 1000
     link/ether 00:0c:29:81:32:bb brd ff:ff:ff:ff:ff:ff
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
     pfifo_fast state UP mode DEFAULT qlen 1000
     link/ether 00:0c:29:81:32:c5 brd ff:ff:ff:ff:ff:ff
```

The ip addresses can be managed with ***ip address*** or ***ip a***

```
rebo[root@centos11 net]# ip address show dev ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP qlen 1000
    link/ether 00:0c:29:81:32:bb brd ff:ff:ff:ff:ff:ff
    inet 192.168.4.131/24 brd 192.168.4.255 scope global ens33
```

```
       valid_lft forever preferred_lft forever
```

```
[root@centos11 net]# ip address add 192.168.55.55/24 dev ens33
```

```
[root@centos11 ~]# ip address show dev ens33
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP qlen 1000
    link/ether 00:0c:29:81:32:bb brd ff:ff:ff:ff:ff:ff
    inet 192.168.4.131/24 brd 192.168.4.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet 192.168.55.55/24 scope global ens33
```

To delete an ip address use ip address delete.

```
[root@centos11 ~]# ip address delete 192.168.55.55/24 dev ens33
```

To setup a static configuration for your NICs on centos you edit the relevant ifcg-xxx files.

```
[root@centos11 network-scripts]# cd /etc/sysconfig/network-scripts
[root@centos11 network-scripts]# ls ifcfg*
ifcfg-ens33  ifcfg-ens34  ifcfg-lo
```

In the *ifcfg-xxx* files you can set many parameters.

```
[root@centos11 network-scripts]# cat ifcfg-ens33
TYPE="Ethernet"
BOOTPROTO="none"
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
NAME="ens33"
UUID="5f24da4d-9fc9-4c91-a72c-aa26c96d0a8f"
DEVICE="ens33"
ONBOOT="yes"
IPADDR="192.168.4.131"
PREFIX="24"
GATEWAY="192.168.4.1"
DNS1="192.168.4.1"
```

2. To add a second ip address to the same network interface modify the ifcfg-xxx file.

```
[root@centos-11 network-scripts]# cat ifcfg-ens33
(snipped)
IPADDR1=192.168.122.11
PREFIX1=24
```

The above changes can be made manually but also with the nmcli command.
This command will change the ifcfg-xxx file.
In the following example we add an ip address to the ifcfg-ens34.

```
[root@centos-11 network-scripts]# nmcli connection show
NAME   UUID                                  TYPE      DEVICE
ens33  aed3d641-2e0b-32ba-945e-9932d9555439  ethernet  ens33
ens34  aed3e641-2e0b-32ba-945e-9932d9555439  ethernet  ens34
[root@centos-11 network-scripts]#

[root@centos-12 network-scripts]# nmcli con mod ens34 +ipv4.addresses
"192.168.122.12/24" ipv4.method manual
```

3. In pre CentOS 7 releases, the package net-tools contained commands like ifconfig, netstat, et cetera. In the current release this package is not installed by default. To get access to these commands, you need to install it first.

```
[root@centos-11 ~]# yum install net-tools -y
Last metadata expiration check: 1:12:15 ago on Wed 28 Apr 2021
(snipped)
Installed:
  net-tools-2.0-0.52.20160912git.el8.x86_64
Complete!
```

Now you can use ifconfig to display network card information.

```
[root@centos11 network-scripts]# ifconfig |grep 192
  inet 192.168.4.131  netmask 255.255.255.0  broadcast 192.168.4.255
  inet 192.168.5.22  netmask 255.255.255.0  broadcast 192.168.5.255
```

4. Routing tables

```
[root@centos-11 ~]# netstat -rn
```

```
Kernel IP routing table
Destination  Gateway      Genmask      Flags   MSS Window  irtt Iface
0.0.0.0       192.168.4.1   0.0.0.0         UG  0 0 0 ens33
192.168.4.0   0.0.0.0       255.255.255.0   U 0 0 0 ens33
192.168.4.0   0.0.0.0       255.255.255.0   U 0 0 0 ens33
192.168.5.0   0.0.0.0       255.255.255.0   U 0 0 0 ens34
```

Using the ***ip*** command you can show your routing entries as well.

```
[root@centos-11 ~]# ip route show
default via 192.168.4.1 dev ens33 proto static metric 100
192.168.4.0/24 dev ens33 proto kernel scope link src 192.168.4.131
metric 100
192.168.4.0/24 dev ens33 proto kernel scope link src 192.168.4.5
metric 100
192.168.5.0/24 dev ens34 proto kernel scope link src 192.168.5.131
metric 101
```

To add a route using the ip command, you need the network, the gateway and the CIDR.

```
[root@centos-11 ~]# ip route add 192.168.100.0/24 via 192.168.5.1
```