Royal University of Bhutan

LESSON – 18-2

NETWORKING CONCEPTS

LEARNING OUTCOMES

- Describing networking concepts
- Validating network configuration

Networking Concepts

- **IP Address:** A unique numerical label assigned to each device on a computer network. An example is ***192.168.1.10.***
- **Subnet Mask**: Defines the network and host portions of an IP address. For example, a subnet mask of ***255.255.255.0 or /24*** indicates that the first three parts of the IP address are for the network, and the last part is for the host.
- **Default Gateway**: The router that connects your local network to the internet or another network.
- **DNS (Domain Name System)**: Translates human-readable domain names (e.g., google.com) into IP addresses.
- **NIC (Network Interface Card):** The hardware component that connects a computer to a network. In Ubuntu, it's represented as a device like ***eth0 or enp0s3.***

Validating Network Configuration

- Several commands can be used to check your network settings.
  - **ip addr:** Displays the IP addresses and network interfaces.
  - **ip route:** Shows the routing table, including the default gateway.
  - **ping** [destination]: Tests connectivity to another device or website. For example, ping google.com.
  - **ss -tunl:** Shows active network connections and listening ports.
  - **cat /etc/netplan/*.yaml:** Displays the current network configuration file.

Validating Network Configuration

- **ss -tunl:** command is a powerful networking tool in Linux that displays active network connections and listening ports.
- It's a modern, more efficient replacement for the older netstat command.
- Let's break down the command's components:
  - **ss**: The command itself, which stands for "socket statistics."
  - **-t**: Filters the output to show only **TCP** (Transmission Control Protocol) sockets.
  - **-u**: Filters the output to show only **UDP** (User Datagram Protocol) sockets.
  - **-n**: Displays numerical addresses and port numbers instead of attempting to resolve hostnames and service names. This makes the output faster and easier to parse for scripts.
  - **-l**: Shows only **listening sockets**. These are the ports on your server that are open and waiting for incoming connections.

Configuring Networking from the Command Line

- The ip command is the modern way to manage network settings temporarily.
  - To assign a static IP address:

```
sudo ip addr add 192.168.1.10/24 dev enp0s3
```

  - To set the default gateway:

```
sudo ip route add default via 192.168.1.1
```

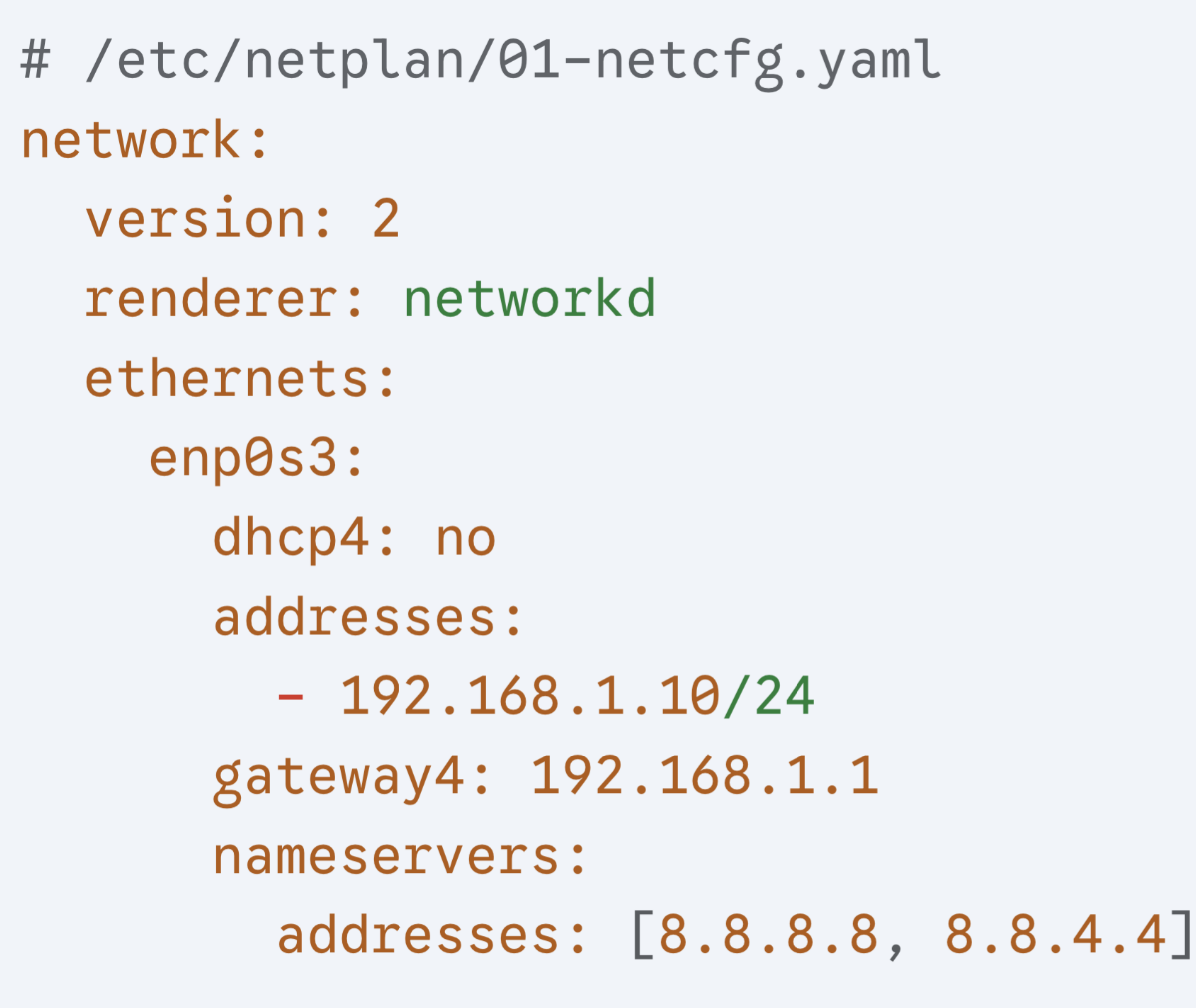
  - To bring an interface up or down:

```
sudo ip link set enp0s3 up
sudo ip link set enp0s3 down
```

Note: *enp0s3 is the interface name*

Editing Network Configuration Files

- Ubuntu 18.04 and later versions use **Netplan** to manage networking permanently. Netplan generates the configuration for network services like *NetworkManager* or *systemd-networkd*.
- The configuration files are located in ***/etc/netplan/*** and have a ***.yaml*** extension.
- Example configuration for a static IP address:
- After editing, run:
  sudo netplan apply



```
# /etc/netplan/01-netcfg.yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: no
      addresses:
        - 192.168.1.10/24
      gateway4: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
```

Configuring Host Names

- ***Host Name:*** The name of your system on a network.

- To view it, use ***hostname***.

- To change it, edit */etc/hostname or*

```
sudo hostnamectl set-hostname new-hostname
```

Configuring Name Resolution

- is the process of converting a human-readable hostname (like www.google.com) into its corresponding machine-readable IP address (like 142.250.191.100).

- This is a fundamental concept in networking that allows devices to find and communicate with each other on a network or the internet.

How Name Resolution Works?

- The process typically follows a specific order of checks to resolve a hostname.
  - **Local Hosts File**
  - **DNS (Domain Name System)**
  - **Application Cache**:

Configuring Name Resolution – Local Host File

- This is managed by the **/etc/hosts** file and the **resolv.conf** file

- **/etc/hosts:** This file maps IP addresses to hostnames locally. It's useful for resolving hostnames on a small network without a DNS server.

- **/etc/resolv.conf:** This file specifies the IP addresses of the DNS servers your system should use to resolve domain names. **Netplan** automatically manages this file based on the nameservers directive in your **.yaml** configuration.