
Linux Server Configuration Guide: Traffic Control

This guide explains the prerequisites and steps to execute the script `wan-simulator.sh`. It covers the configuration of network interfaces without IP in persistent mode, setting up a network bridge, and configuring the `ifb0` interface.

Prerequisites:

- **PC** with 2 Ethernet cards
- **Monitor/Keyboard** (server will be in bridge mode and fully transparent on the network)
- **Linux Operating System** (e.g., Ubuntu 24.04 LTS)
- **Graphical Environment Installation** (e.g., `lightdm + ubuntu-desktop`)

```
bash
```

```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install lightdm
sudo apt-get install ubuntu-desktop
```

```
bash
```

```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install lightdm
sudo apt-get install ubuntu-desktop
```

1. Configuring Network Interfaces Without IP in Persistent Mode

To prevent network interfaces (e.g., `enp1s0` and `enp2s0`) from automatically retrieving an IP address on boot, modify the network configuration using **Netplan**.

Step 1: Configure Netplan to Disable IP on Interfaces

1. Open the Netplan configuration file:

```
bash
```

```
sudo nano /etc/netplan/01-netcfg.yaml
```

```
sudo nano /etc/netplan/01-netcfg.yaml
```

2. Add the following configuration for IP-less interfaces (replace `enp1s0` and `enp2s0` with your network interfaces):

yaml

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp1s0:
      dhcp4: no
      dhcp6: no
      addresses: []
    enp2s0:
      dhcp4: no
      dhcp6: no
      addresses: []
```

yaml

Copy code

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp1s0:
      dhcp4: no
      dhcp6: no
      addresses: []
    enp2s0:
      dhcp4: no
      dhcp6: no
      addresses: []
```

3. Apply the configuration:

bash

```
sudo netplan apply
```

bash

Copy code

```
sudo netplan apply
```

2. Setting Up a Persistent Network Bridge

To create a bridge between `enp1s0` and `enp2s0` in persistent mode, configure **Netplan** again.

Step 2: Configure a Network Bridge with Netplan

1. Modify the same Netplan configuration file `/etc/netplan/01-netcfg.yaml` to include a bridge called `nm-bridge`:

```

yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp1s0:
      dhcp4: no
      dhcp6: no
    enp2s0:
      dhcp4: no
      dhcp6: no
  bridges:
    nm-bridge:
      interfaces: [enp1s0, enp2s0]
      dhcp4: no
      dhcp6: no

```

```

yaml
Copy code
network:
  version: 2
  renderer: networkd
  ethernets:
    enp1s0:
      dhcp4: no
      dhcp6: no
    enp2s0:
      dhcp4: no
      dhcp6: no
  bridges:
    nm-bridge:
      interfaces: [enp1s0, enp2s0]
      dhcp4: no
      dhcp6: no

```

2. Apply the configuration:

```

bash
sudo netplan apply

```

```

bash
Copy code
sudo netplan apply

```

3. Creating and Configuring the `ifb0` Interface

The `ifb0` interface is a virtual interface that is dynamically created to redirect incoming traffic (bandwidth management for simulating WAN links with different speeds).

Step 3: Make the `ifb0` Interface Persistent

1. Add the `ifb` module to the `/etc/modules` file to ensure it loads automatically on boot:

```
bash
sudo nano /etc/modules
```

```
bash
Copy code
sudo nano /etc/modules
```

2. Add the following line at the end of the file:

```
ifb
```

```
Copy code
ifb
```

3. Save and close the file.
4. Restart the machine and check if the `ifb` module is loaded:

```
bash
lsmod | grep ifb
```

```
bash
Copy code
lsmod | grep ifb
```

5. Bring up the `ifb0` interface:

```
bash
ip link add ifb0 type ifb
ip link set ifb0 up
```

```
bash
Copy code
ip link add ifb0 type ifb
ip link set ifb0 up
```

4. Installing Required Packages

Ensure the following packages are installed for the script to work properly:

Required Packages:

1. `tc` (traffic control)
2. `bridge-utils` (for managing bridges)
3. `ifb` (virtual interface for incoming traffic)
4. `iproute2` (tools for managing network interfaces)
5. `dhclient` (for DHCP management if necessary)

Step 4: Install Required Packages

Install these packages using the following commands:

```
bash
```

```
sudo apt-get update  
sudo apt-get install iproute2 bridge-utils isc-dhcp-client ifb
```

```
bash
```

Copy code

```
sudo apt-get update  
sudo apt-get install iproute2 bridge-utils isc-dhcp-client ifb
```

5. Verifying Configuration After Reboot

After following the steps and rebooting the system, verify that everything is correctly set up.

1. Check IP-less Interfaces:

```
bash
```

```
ip addr show enp1s0  
ip addr show enp2s0
```

```
bash
```

Copy code

```
ip addr show enp1s0  
ip addr show enp2s0
```

Both `enp1s0` and `enp2s0` should have no assigned IP addresses.

2. Check the Bridge:

```
bash
```

```
brctl show nm-bridge
```

```
bash
```

```
Copy code
```

```
brctl show nm-bridge
```

The nm-bridge should be active and include `enp1s0` and `enp2s0`.

3. Check the `ifb0` Interface:

```
bash
```

```
ip link show ifb0
```

```
bash
```

```
Copy code
```

```
ip link show ifb0
```

The `ifb0` interface should be in `UP` mode.

This guide provides a comprehensive configuration process for creating a robust network simulation environment on Linux. By following these steps, you'll be ready to simulate WAN conditions using traffic control tools.