**PROCESS COMMUNICATION BETWEEN NAMESPACES**

**Communicating between process and different namespaces.**

This guide outlines the steps to create two namespaces named ***blue-namespace*** and ***lemon-namespace***, and establish a virtual Ethernet network between them using ***veth*** interfaces. The goal is to enable communication between the namespaces and process. Besides allowing them to ping or curl from one namespace to process.

**Prerequisites**

Linux operating system

Root or sudo access

Packages

sudo apt update

sudo apt upgrade -y

sudo apt install iproute2 -y

sudo apt install net-tools

sudo apt install curl telnet tcpdump iputils-ping -y

**Steps**

**1. Enable IP forwarding in the Linux kernel:**

sudo sysctl -w net.ipv4.ip\_forward=1

This step enables IP forwarding in the Linux kernel, allowing the namespaces to communicate with each other.

**2. Create namespaces:**

sudo ip netns add blue-namespace

sudo ip netns add lemon-namespace

This step creates two namespaces named ***blue-namespace*** and ***lemon-namespace***. We can list our namespaces by running ip netns list

**3. Create the virtual Ethernet link pair:**

sudo ip link add veth-blue type veth peer name veth-lemon

This command creates a virtual Ethernet link pair consisting of veth-blue and veth-lemon at ***root namespace***.

In order to verify, run sudo ip link list

**4. Set the cable as NIC**

sudo ip link set veth-blue netns blue-namespace

sudo ip link set veth-lemon netns lemon-namespace

This command acts as ***NIC*** link pair consisting of veth-blue and veth-lemon.

To verify run sudo ip netns exec blue-namespace ip link and sudo ip netns exec lemon-namespace ip link

But as we see, interface has been created but it's **DOWN** and has no ip. Now assign a ip address and turn it **UP**.

**5. Assign IP Addresses to the Interfaces**

sudo ip netns exec blue-namespace ip addr add 192.168.0.1/24 dev veth-blue

sudo ip netns exec lemon-namespace ip addr add 192.168.0.2/24 dev veth-lemon

In this step, IP addresses are assigned to the veth-blue interface in the blue-namespace and to the veth-lemon interface in the lemon-namespace.

To verify run sudo ip netns exec blue-namespace ip addr and sudo ip netns exec lemon-namespace ip addr

**6. Set the Interfaces Up**

sudo ip netns exec blue-namespace ip link set veth-blue up

sudo ip netns exec lemon-namespace ip link set veth-lemon up

These commands set the veth-blue and veth-lemon interfaces ***up***, enabling them to transmit and receive data.

Now run again sudo ip netns exec blue-namespace ip link and sudo ip netns exec lemon-namespace ip link to verify

**7. Set Default Routes**

sudo ip netns exec blue-namespace ip route add default via 192.168.0.1 dev veth-blue

sudo ip netns exec lemon-namespace ip route add default via 192.168.0.2 dev veth-lemon

These commands set the **default routes** within each namespace, allowing them to route network traffic.

In order to verify run sudo ip netns exec blue-namespace ip route and sudo ip netns exec lemon-namespace ip route

**In addition, the route command in the context of the ip netns exec allows you to view the routing table of a specific network namespace. The routing table contains information about how network traffic should be forwarded or delivered.**

To view the routing table of the lemon-namespace, we can execute the following command:

sudo ip netns exec lemon-namespace route

sudo ip netns exec blue-namespace route

**8. Test Connectivity**

sudo ip netns exec blue-namespace ping 192.168.0.2

sudo ip netns exec lemon-namespace ping 192.168.0.1

Use these commands to test the connectivity between the namespaces by pinging each other's IP address.

**9. Create a server and run**

Now we are ready to create a server inside one namespace and ping or curl from another namespace. Let's create a simple hello-world flask application.

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route('/')

def hello\_world():

return 'Hello, World!'

if \_\_name\_\_ == '\_\_main\_\_':

app.run(host='0.0.0.0', port=3000, debug=True)

Use sudo nano server.py and write down these lines. Press control+O , Enter and control+X.

To run this server we need to create virtual environment and install packages

python3 -m venv venv

source venv/bin/activate

pip3 install flask

Run python3 server.py, but before that, we need to enter the inside of one namespace. Let's try with blue namespace.

sudo ip netns exec blue-namespace /bin/bash

Lets check the ip info. Run ifconfig.

Now lets run the application.

source venv/bin/activate

python3 server.py

**10. Now let's curl from another namespace**

Let's get into lemon namespace. Run sudo ip netns exec lemon-namespace /bin/bash. Run ifconfig

curl -v <http://192.168.0.1:3000>

**11. Let's create one more server**

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route('/')

def hello\_world():

return 'Hello world from process 2'

if \_\_name\_\_ == '\_\_main\_\_':

app.run(host='0.0.0.0', port=3001, debug=True)

Run it from the blue namespace and curl again from the lemon namespace.

curl -v http://192.168.0.1:3001

**12. Clean Up (optional)**

sudo ip netns del blue-namespace

sudo ip netns del lemon-namespace