

Network Namespace Simulation Assignment

Step 1: Creating namespaces and bridges:

```
NS1=ns1
NS2=ns2
ROUTER_NS=router-ns
BR0=br0
BR1=br1
VETH1=veth-ns1
VETH2=veth-ns2
VPEER1=veth-br0
VPEER2=veth-br1
VR1=vr1
VR2=vr2
VRP1=vrp1
VRP2=vrp2
```

```
echo "Creating namespaces and bridges..."
```

```
sudo ip netns add $NS1
sudo ip netns add $NS2
sudo ip netns add $ROUTER_NS
sudo ip link add $BR0 type bridge
sudo ip link add $BR1 type bridge
sudo ip link set $BR0 up
sudo ip link set $BR1 up
```

```
sudo ip netns list
```

Step 2: Create Virtual Interfaces for namespaces, router and bridges:

```
sudo ip link add $VETH1 type veth peer name $VPEER1
sudo ip link add $VETH2 type veth peer name $VPEER2
sudo ip link add $VR1 type veth peer name $VRP1
sudo ip link add $VR2 type veth peer name $VRP2
```

Step 3: Namespace assignments and bridge connections:

```
sudo ip link set $VETH1 netns $NS1
```

```
sudo ip link set $VETH2 netns $NS2
sudo ip link set $VR1 netns $ROUTER_NS
sudo ip link set $VR2 netns $ROUTER_NS
sudo ip link set $VPEER1 master $BR0
sudo ip link set $VPEER2 master $BR1
sudo ip link set $VRP1 master $BR0
sudo ip link set $VRP2 master $BR1
```

Step 4: Bring up veth interfaces

```
sudo ip link set $VPEER1 up
sudo ip link set $VPEER2 up
sudo ip link set $VRP1 up
sudo ip link set $VRP2 up
```

Step 5: Configure IP Addresses and bring up veth interfaces in namespaces:

```
sudo ip addr add 10.11.0.254/24 dev $BR0
sudo ip addr add 10.12.0.254/24 dev $BR1
sudo ip netns exec $NS1 ip addr add 10.11.0.2/24 dev $VETH1
sudo ip netns exec $NS2 ip addr add 10.12.0.2/24 dev $VETH2
sudo ip netns exec $ROUTER_NS ip addr add 10.11.0.1/24 dev $VR1
sudo ip netns exec $ROUTER_NS ip addr add 10.12.0.1/24 dev $VR2
```

```
sudo ip netns exec $NS1 ip link set $VETH1 up
sudo ip netns exec $NS2 ip link set $VETH2 up
sudo ip netns exec $ROUTER_NS ip link set $VR1 up
sudo ip netns exec $ROUTER_NS ip link set $VR2 up
```

Step 6: Ensure correct MAC addresses are set

```
sudo ip netns exec $NS1 ip link set $VETH1 address 02:42:ac:11:00:02
sudo ip netns exec $NS2 ip link set $VETH2 address 02:42:ac:12:00:02
sudo ip netns exec $ROUTER_NS ip link set $VR1 address 02:42:ac:11:00:01
sudo ip netns exec $ROUTER_NS ip link set $VR2 address 02:42:ac:12:00:01
```

Step 7: Enable IP forwarding

```
echo "Enabling IP forwarding and setting routes..."
sudo ip netns exec $ROUTER_NS sysctl -w net.ipv4.ip_forward=1
```

Step 8: Default routes in namespaces

```
sudo ip netns exec $NS1 ip route add default via 10.11.0.1
sudo ip netns exec $NS2 ip route add default via 10.12.0.1
```

Step 9: Flush ARP tables to refresh entries

```
echo "Flushing ARP tables..."
sudo ip netns exec $NS1 ip neigh flush all
sudo ip netns exec $NS2 ip neigh flush all
sudo ip netns exec $ROUTER_NS ip neigh flush all
```

Step 10: Adding iptables forwarding

```
sudo iptables --append FORWARD --in-interface $BR0 --jump ACCEPT
sudo iptables --append FORWARD --out-interface $BR0 --jump ACCEPT

sudo iptables --append FORWARD --in-interface $BR1 --jump ACCEPT
sudo iptables --append FORWARD --out-interface $BR1 --jump ACCEPT
```

Step 11: Setting up NAT in router namespace

```
sudo ip netns exec $ROUTER_NS iptables -t nat -A POSTROUTING -o $VR1 -j
MASQUERADE
sudo ip netns exec $ROUTER_NS iptables -t nat -A POSTROUTING -o $VR2 -j
MASQUERADE
```

Step 12: Connectivity Test

```
echo "Testing connectivity from ns1 to ns2..."
sudo ip netns exec $NS1 ping -c 4 10.12.0.2
```

Step 13: Verify routing

```
sudo ip netns exec $NS1 ip route
sudo ip netns exec $NS2 ip route
sudo ip netns exec $ROUTER_NS ip route
```

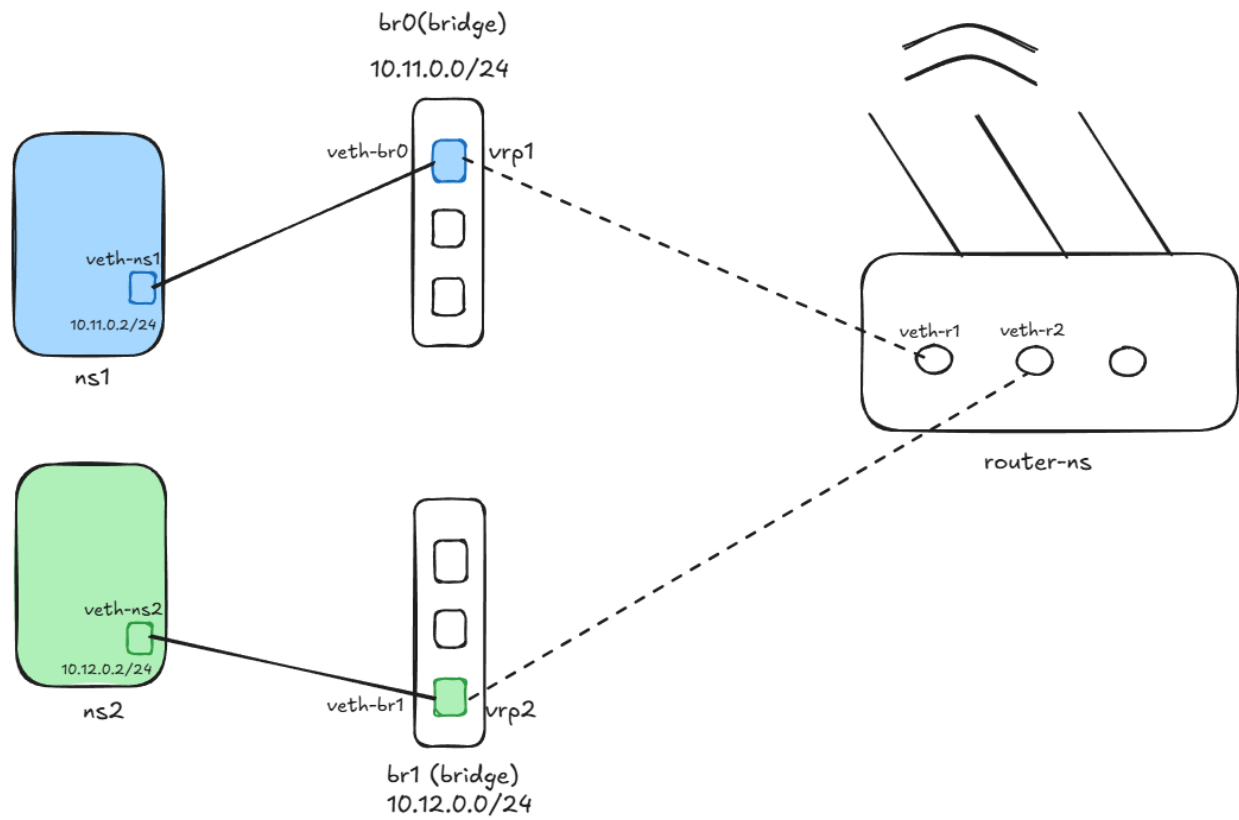


Fig: Network Diagram

IP Addressing Scheme:

ns1: 10.11.0.2/24

ns2: 10.12.0.2/24

router-ns (br0): 10.11.0.1/24

router-ns (br1): 10.12.0.1/24