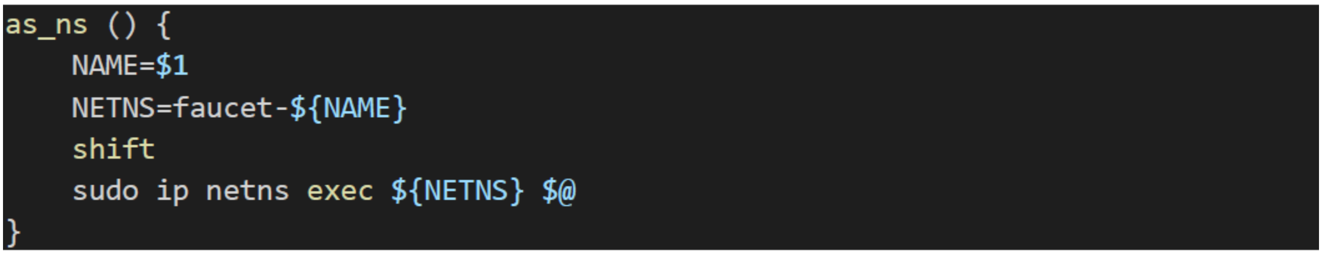
LAB3 - Faucet - Suite

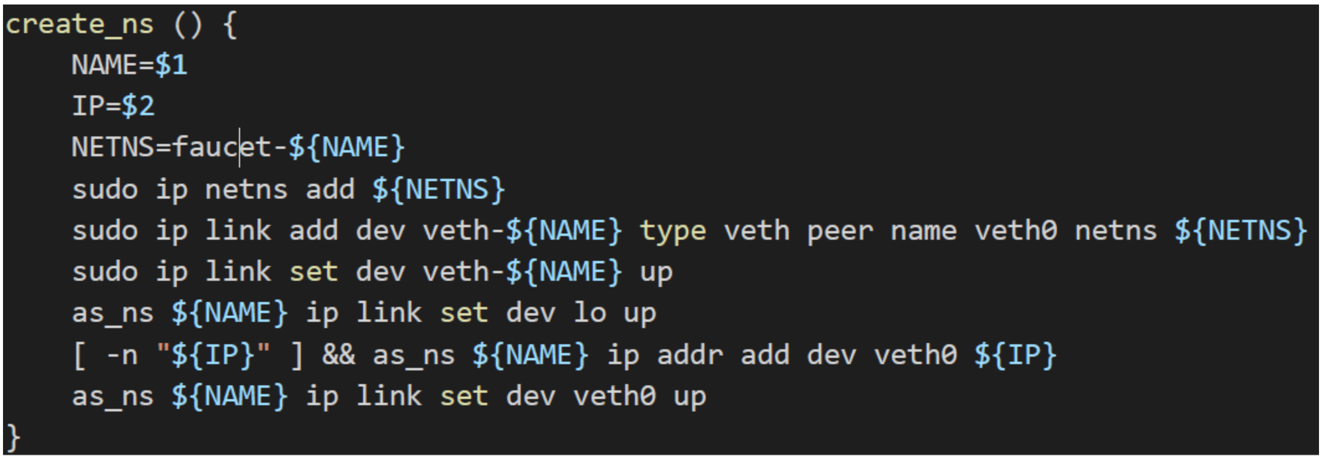
Advanced Network

You have already installed Faucet and Open vSwitch.

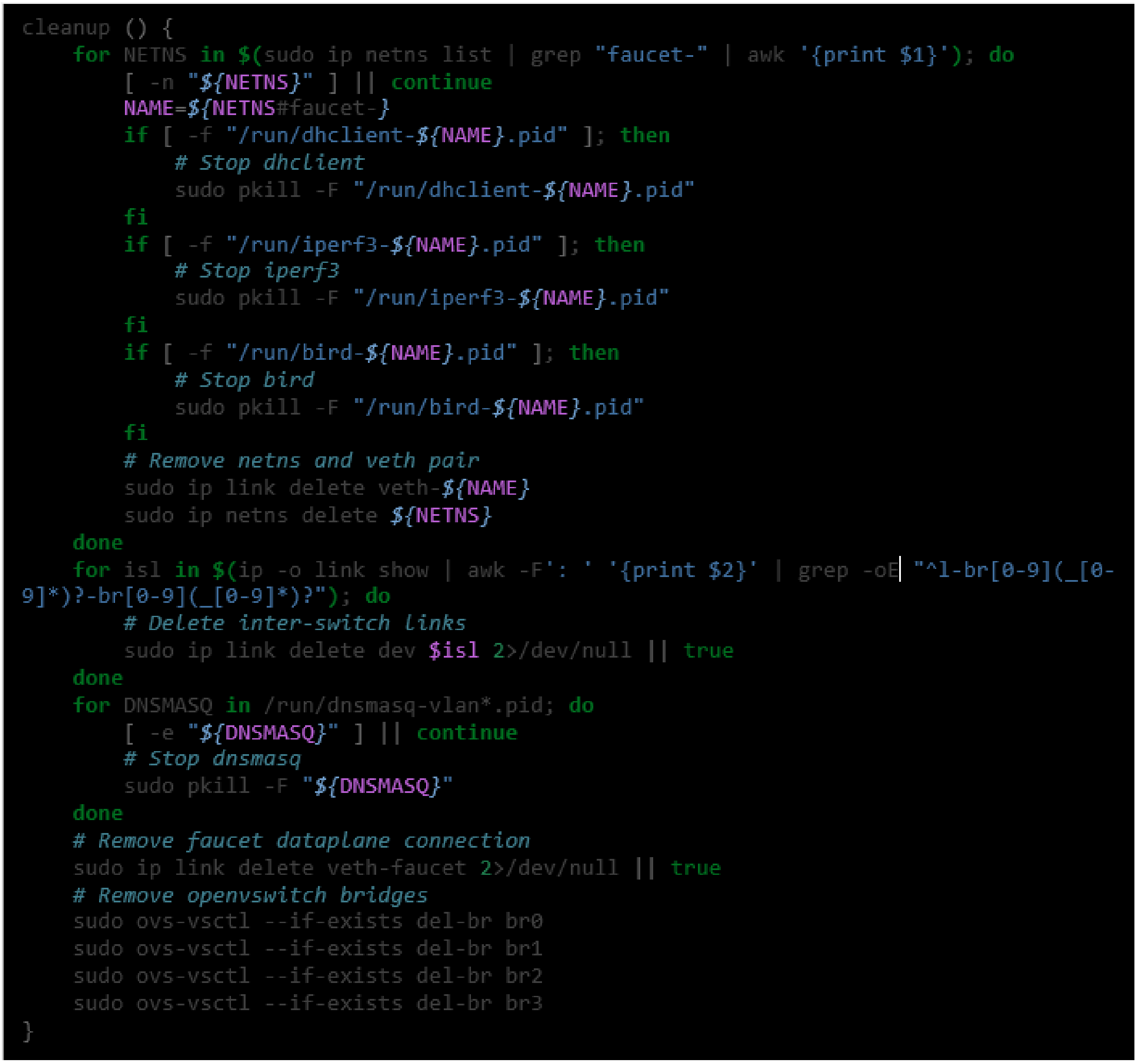
Copy the following definitions into your bash terminal:



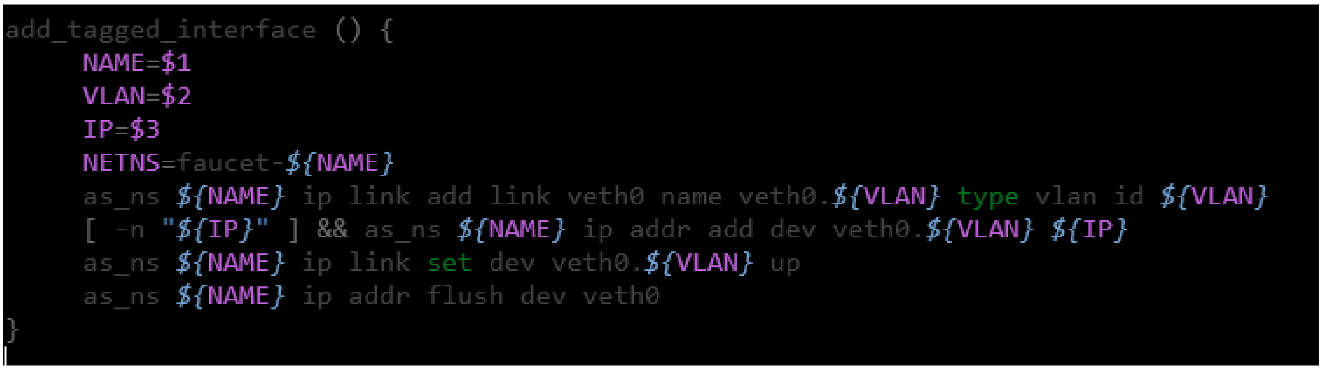
```
as_ns () {
    NAME=$1
    NETNS=faucet-${NAME}
    shift
    sudo ip netns exec ${NETNS} $@
}
```



```
create_ns () {
    NAME=$1
    IP=$2
    NETNS=faucet-${NAME}
    sudo ip netns add ${NETNS}
    sudo ip link add dev veth-${NAME} type veth peer name veth0 netns ${NETNS}
    sudo ip link set dev veth-${NAME} up
    as_ns ${NAME} ip link set dev lo up
    [ -n "${IP}" ] && as_ns ${NAME} ip addr add dev veth0 ${IP}
    as_ns ${NAME} ip link set dev veth0 up
}
```



```
cleanup () {
    for NETNS in $(sudo ip netns list | grep "faucet-" | awk '{print $1}'); do
        [ -n "${NETNS}" ] || continue
        NAME=${NETNS#faucet-}
        if [ -f "/run/dhclient-${NAME}.pid" ]; then
            # Stop dhclient
            sudo pkill -F "/run/dhclient-${NAME}.pid"
        fi
        if [ -f "/run/iperf3-${NAME}.pid" ]; then
            # Stop iperf3
            sudo pkill -F "/run/iperf3-${NAME}.pid"
        fi
        if [ -f "/run/bird-${NAME}.pid" ]; then
            # Stop bird
            sudo pkill -F "/run/bird-${NAME}.pid"
        fi
        # Remove netns and veth pair
        sudo ip link delete veth-${NAME}
        sudo ip netns delete ${NETNS}
    done
    for isl in $(ip -o link show | awk -F': ' '{print $2}' | grep -oE "^l-br[0-9](_[0-
9]*)?-br[0-9](_[0-9]*)?"); do
        # Delete inter-switch links
        sudo ip link delete dev $isl 2>/dev/null || true
    done
    for DNSMASQ in /run/dnsmasq-vlan*.pid; do
        [ -e "${DNSMASQ}" ] || continue
        # Stop dnsmasq
        sudo pkill -F "${DNSMASQ}"
    done
    # Remove faucet dataplane connection
    sudo ip link delete veth-faucet 2>/dev/null || true
    # Remove openvswitch bridges
    sudo ovs-vsctl --if-exists del-br br0
    sudo ovs-vsctl --if-exists del-br br1
    sudo ovs-vsctl --if-exists del-br br2
    sudo ovs-vsctl --if-exists del-br br3
}
```



```
add_tagged_interface () {
    NAME=$1
    VLAN=$2
    IP=$3
    NETNS=faucet-${NAME}
    as_ns ${NAME} ip link add link veth0 name veth0.${VLAN} type vlan id ${VLAN}
    [ -n "${IP}" ] && as_ns ${NAME} ip addr add dev veth0.${VLAN} ${IP}
    as_ns ${NAME} ip link set dev veth0.${VLAN} up
    as_ns ${NAME} ip addr flush dev veth0
}
```

1. VLANs Configuration

in this part, we will create a network where a single open vswitch br0 connects to 9 hosts.

- What are the different VLAN mode a port can be used for?

- To create the hosts and switch again run this commands

  ```
  cleanup
  create_ns host1 10.20.0.1/24
  create_ns host2 10.20.0.2/24
  sudo ovs-vsctl add-br br0
  -- set bridge br0 other-config:datapath-id=0000000000000001
  -- set bridge br0 other-config:disable-in-band=true
  -- set bridge br0 fail_mode=secure
  -- add-port br0 veth-host1 -- set interface veth-host1
  ofport_request=1
  -- add-port br0 veth-host2 -- set interface veth-host2
  ofport_request=2
  -- set-controller br0 tcp:127.0.0.1:6653 tcp:127.0.0.1:6654
  ```

- Keep host1, host2 on the native VLAN 100 (office VLAN)

- Assign host3 and host4 a VLAN interface (vid:100) as they are on a tagged port.

  ```
  create_ns host3 0.0.0.0
  create_ns host4 0.0.0.0
  add_tagged_interface host3 200 10.20.0.3/24
  add_tagged_interface host4 200 10.20.0.4/24
  ```

- Assign host5 and host6 an IP address from the VLAN 200 range.

  ```
  create_ns host5 10.20.2.5/24
  create_ns host6 10.20.2.6/24
  ```

- Add host7 and host8 a Vlan interface (vid:300) as they are on a tagged port.

  ```
  create_ns host7 0.0.0.0
  create_ns host8 0.0.0.0
  add_tagged_interface host8 300 10.20.3.7/24
  add_tagged_interface host8 300 10.20.3.8/24
  ```

- Add host9 to all VLANs (100, 200, 300)

  ```
  create_ns host9 0.0.0.0
  add_tagged_interface host9 100 10.20.1.9/24
  ```

```
add_tagged_interface host9 200 10.20.2.9/24
add_tagged_interface host9 300 10.20.3.9/24
```

- Connect all the hosts to the switch br0

```
sudo ovs-vsctl add-port br0 veth-host3 -- set interface veth-host3 ofport_request=3
-- add-port br0 veth-host4 -- set interface veth-host4 ofport_request=4
-- add-port br0 veth-host5 -- set interface veth-host5 ofport_request=5
-- add-port br0 veth-host6 -- set interface veth-host6 ofport_request=6
-- add-port br0 veth-host7 -- set interface veth-host7 ofport_request=7
-- add-port br0 veth-host8 -- set interface veth-host8 ofport_request=8
-- add-port br0 veth-host9 -- set interface veth-host9 ofport_request=9
```

- Now we need to modify **/etc/faucet/faucet.yaml** to apply our settings:

```
vlans:
vlan100:
vid: 100
vlan200:
vid: 200
vlan300:
vid: 300
dps:
sw1:
dp_id: 0x1
hardware: "Open vSwitch"
interfaces:
1:
name: "host1"
```

```
description: "host2 network namespace"
native_vlan: vlan100
2:
name: "host2"
description: "host2 network namespace"
native_vlan: vlan100
3:
name: "host3"
tagged_vlans: [vlan100]
4:
name: "host4"
tagged_vlans: [vlan100]
5:
name: "host5"
native_vlan: vlan200
6:
name: "host6"
native_vlan: vlan200
7:
name: "host7"
tagged_vlans: [vlan300]
8:
name: "host8"
tagged_vlans: [vlan300]
9:
name: "host9"
tagged_vlans: [vlan100,vlan200,vlan300]
```

- Send SIGHUP signal to reload the configuration file, and check how its log the new configuration in **/var/log/faucet/faucet.log** file :
  ```
  sudo systemctl reload faucet
  cat /var/log/faucet/faucet.log
  ```

Ping Tests

- Ping between hosts in the same vlan:
  ```
  as_ns host1 ping 192.168.0.2
  as_ns host3 ping 192.168.0.4
  as_ns host5 ping 192.168.2.6
  as_ns host7 ping 192.168.3.8
  ```
- Now we can test the trunk link to host9 from different VLANs:
  ```
  as_ns host1 ping 10.20.0.9
  as_ns host3 ping 10.20.0.9
  as_ns host5 ping 10.20.2.9
  as_ns host7 ping 10.20.3.9
  ```

2. ACL Configuration

Let’s apply an ACL onVLAN 200. We will block any ICMP packets on VLAN 200.

- First create an ACL to block the ping. Open **/etc/faucet/faucet.yaml** and add the ‘acls’ section.
  ```
  acls:
  block-ping:
  - rule:
  dl_type: 0x800 # IPv4
  ip_proto: 1 # ICMP
  actions:
  allow: False
  - rule:
  dl_type: 0x86dd # IPv6
  ip_proto: 58 # ICMPv6
  actions:
  allow: False
  ```
- Apply this ACL on VLAN 200 , go to **etc/faucet/faucet.yaml** and add
  ```
  vlans:
  vlan100:
  vid: 100
  vlan200:
  ```

```
vid: 200
acls_in: [block-ping] # Apply ACL only on vlan200
vlan300:
vid: 300
```

- Before we reload the configuration file. Let’s verify that pinging is working between hosts in VLAN 200.
  ```
  as_ns host3 ping 10.20.2.4
  ```
- Send SIGHUP signal to reload the configuration file.
  ```
  sudo systemctl reload faucet
  ```
- Try to ping again from host5 and host6, what do you notice?
- Apply this ACL on VLAN 300 and try to ping from host7 and host8.