Lab 3: Installing the SDN Controller and Establishing the Control Plane

Objective: To separate the **Control Plane** from the **Data Plane** by connecting the Open vSwitch (OVS) to an external SDN Controller, enabling programmatic control over network traffic.

---

1. Technical Context

In Section 10.6, we established that SDN makes a network programmable by separating the control plane from the data plane. While Lab 2 used OVS as a standalone learning switch, this lab introduces the **SDN Controller**, which acts as the central "brain" of the network. The link between the controller and the switch is known as the **Control Plane** link, typically utilizing the **OpenFlow** protocol to communicate flow rules.

2. Topology Requirements

- **Nodes**:
  - 1x SDN Controller (e.g., **ONOS** or **OpenDaylight** running as a GNS3 Docker container or external VM).
  - 1x Open vSwitch (OVS) Appliance.
  - 2x VPCS (Host A, Host B).
  - 1x NAT Node (to provide internet access for the controller/OVS updates if needed).
- **Connections**:
  - SDN Controller (Eth0) $\rightarrow$ OVS (Management/Control Port).
  - Host A (Eth0) ⟶ OVS (Port 1).
  - Host B (Eth0) ⟶ OVS (Port 2).

3. Procedure

Step A: Deploying the SDN Controller

1. **Import Appliance**: In GNS3, import the ONOS or OpenDaylight container/appliance.
2. **Resource Allocation**: Ensure the **GNS3 VM** (acting as your **NFVI**) has sufficient RAM (at least 4GB) to host the controller.

3. **Start the Node**: Power on the controller and wait for the management console to become accessible.

Step B: Establishing the Control Plane Link

You must tell the OVS bridge to stop acting independently and start listening to the controller.

1. **Console into OVS**:
   Bash
   ```
   # Set the controller IP (Replace <CONTROLLER_IP> with the actual IP of your SDN node)
   ovs-vsctl set-controller br0 tcp:<CONTROLLER_IP>:6633
   ```

2. **Verify Connection**:
   Bash
   ```
   ovs-vsctl show
   ```

   - **Observation**: Look for is_connected: true under the Controller section. This indicates the switch is now a "slave" to the controller's logic.

Step C: Observing the "Silent" Network

1. Configure IPs for Host A (10.0.0.1) and Host B (10.0.0.2).
2. Attempt a ping 10.0.0.2.
   - **Logic Check**: The ping will fail initially. Why? Because the OVS no longer has a "MAC learning" table, it is waiting for the SDN Controller to provide specific **Flow Rules**.

Step D: Activating Reactive Forwarding

1. Access the SDN Controller's web interface (usually port 8181 for ONOS, check for the ODL controller you are using).
2. Activate the **Reactive Forwarding** application.
3. **Result**: The ping should now succeed as the controller dynamically pushes rules to the OVS upon seeing the first packet of the flow.

---

4. Technical Analysis: The Synergistic Relationship

This lab demonstrates the synergy described in the ETSI framework. The **SDN Controller** manages the network logic, while the **OVS** (residing within the **NFVI**) handles the high-speed packet forwarding. By 2026 standards, this separation allows for the dynamic "spin-up" of services without modifying physical hardware.

Lab 3 Assessment Questions

1. Describe the difference in traffic handling between Lab 2 (standalone) and Lab 3 (controller-led). Refer to the **Control Plane** and **Data Plane** in your answer.

2. How does the "Inventory & State Database" relate to the controller's ability to manage the OVS bridge?.