

Department of Artificial Intelligence & Data Science

Experiment No.10
Simulation of software defined network using mininet
Name: Pratik Mankar
Roll Number: 34
Date of Performance: 15/9/25
Date of Submission:9/10/25
Marks:
Sign:

CSL501: Web Computing and Network Lab



Department of Artificial Intelligence & Data Science

Aim: To simulate a Software Defined Network (SDN) environment using Mininet and observe communication between hosts.

Objective:

To understand the concept of Software Defined Networking

To simulate a virtual network topology using Mininet

To configure and test connectivity between hosts using ping command

To integrate a controller (such as POX/OVS) for centralized control of the

SDN Requirement:

Ubuntu Linux (or VM with Ubuntu installed)

Mininet installed (mininet.org)

Open vSwitch (default in Mininet)

Python support for running Mininet scripts

Theory:

Software Defined Networking (SDN) is a networking paradigm that separates the control plane from the data plane. In SDN, a central controller manages the flow of traffic in the network, while switches and routers only forward packets based on rules defined by the controller.

Mininet is a popular network emulator that can create a realistic virtual network with hosts, switches, and controllers on a single machine. It allows testing of SDN applications quickly and efficiently. Key components:

Host: Represents end devices in the network

Switch: Open vSwitch used for packet forwarding

Controller: Centralized controller (like POX, Ryu, ONOS) that manages the network Link: Virtual connections between hosts, switches, and controllers

CSL501: Web Computing and Network Lab

Department of Artificial Intelligence & Data Science

Procedure:

Step 1: Launch Mininet
Open a terminal in Ubuntu and run:
sudo mn --topo single,3 --mac --switch ovsk --controller remote

This command creates a simple topology with 1 switch and 3 hosts.

Step 2: Test connectivity

Use the command:

pingall

This sends ICMP packets between all hosts to verify connectivity.

Step 3: Start Mininet CLI

Run commands inside Mininet CLI:

h1 ping h2

h1 ping h3

Step 4: Create custom topology using Python

Create a Python script (topo.py):

from mininet.topo import Topo from mininet.net import Mininet from mininet.node import RemoteController from mininet.cli import CLI

class MyTopo(Topo):

def build(self):

h1 = self.addHost('h1')

h2 = self.addHost('h2')

s1 = self.addSwitch('s1')

self.addLink(h1, s1)

self.addLink(h2, s1)

topo = MyTopo()

net = Mininet(topo=topo, controller=RemoteController)

net.start()

CLI(net)

net.stop()

Run the script using:



Department of Artificial Intelligence & Data Science

sudo python3 topo.py

Step 5: Attach a controller Install and run POX controller: git clone https://github.com/noxrepo/pox.git

cd pox ./pox.py forwarding.l2_learning

Step 6: Connect Mininet to POX controller Run Mininet with remote controller option: sudo mn --controller=remote,ip=127.0.0.1,port=6633

Output:

Pingall shows 100% packet delivery between hosts

Hosts communicate via switch controlled by the SDN controller

Routing and forwarding decisions are handled dynamically by the controller

Output:-

```
Preparing to unpack .../915-libglib2.0-dew-bin 2.72.4-8ubuntu2.6 amd64.deb ...
Unpacking libglib2.0-deta 2.72.4-9ubuntu2.3 ...
Preparing to unpack .../916-libglib2.0-deta 2.72.4-9ubuntu2.3 ...
Preparing to unpack .../916-libglib2.0-data 2.72.4-9ubuntu2.5 amd64.deb ...
Unpacking libglib2.0-data (2.72.4-9ubuntu2.6) over (2.72.4-9ubuntu2.3) ...
Preparing to unpack .../917-libglib2.0-bin 2.72.4-9ubuntu2.6 amd64.deb ...
Unpacking libglib2.0-data (2.72.4-9ubuntu2.6) over (2.72.4-9ubuntu2.3) ...
Preparing to unpack .../917-libglib2.0-bin 2.72.4-9ubuntu2.6 amd64.deb ...
Unpacking libglib2.0-data (2.72.4-9ubuntu2.6) over (2.72.4-9ubuntu2.3) ...
Preparing to unpack .../918-libglib2.0-be 2.72.4-9ubuntu2.6 amd64.deb ...
Unpacking libglib2.0-data (2.72.4-9ubuntu2.6) over (2.72.4-9ubuntu2.3) ...
Preparing to unpack .../919-python-apt-common 2.4.9ubuntu3.3 ...
Preparing to unpack .../920-python3-apt 2.4.9ubuntu3.3 ...
Preparing to unpack .../920-python3-apt 2.4.9ubuntu3.3 ...
Preparing to unpack .../922-python3-apt 2.4.9ubuntu3.3 ...
Preparing to unpack .../923-python3-apt 2.4.2ubuntu3.3 ...
Preparing to unpack .../923-python3-apt 2.4.9ubuntu3.3 ...
Preparing to unpack .../923-python3-apt 2.4.9ubuntu3.5 ...
Unpacking python3-setuptools (536.6-1.2ubuntu3.2) ...
Preparing to unpack .../923-yubuntu3.5 ...
Unpacking python3-setuptools (536.6-1.2ubuntu3.2) ...
Preparing to unpack .../923-suda 1.9 -yubuntu2.5 ...
Unpacking side (1.9.9-lubuntu2.5) over (1.9.9-lubuntu3.2) ...
Preparing to unpack .../924-regata 2.9ubuntu3.2 ...
Unpacking side (1.9.9-lubuntu2.5) over (1.9.9-lubuntu3.2) ...
Preparing to unpack .../925-oubuntu3.2 ...
Unpacking side (1.9.9-lubuntu3.2) over (2.9.4-1.0ubuntu3.2) ...
Preparing to unpack .../925-bubuntu3.2 ...
Unpacking side (1.9.9-lubuntu3.2) over (2.9.4-1.0ubuntu3.2) ...
Preparing to unpack .../925-oubuntu3.2 ...
Unpacking side (1.9.9-lubuntu3.2) ...
Preparing to unpack .../925-oubuntu3.2 ...
Unpacking side (1.9.9-lubuntu3.2) ...
Preparing to unpack .../925-oubuntu3.2 ...
Unpacking side (1.9.9-lubuntu3.2) .
```

```
Terminal - labex@68eb288d0f92ec70f139cd6b:
       Edit View Terminal Tabs Help
Setting up python3-dev (3.10.6-1~22.04.1) ..
Setting up python3-dev (3.10.8 1 22.01.1) ...

Setting up python3-pip (22.0.2+dfsg-lubuntu0.7) ...

Setting up gfortran-11 (11.4.0-1ubuntu1~22.04.2) ...

Setting up libwebkit2gtk-4.0-37:amd64 (2.48.7-0ubuntu0.22.04.2) ...

Setting up openjdk-17-amd64 (43.1 1.4buntu0.1)
Setting up libyelp0:amd64 (42.1-lubuntu0.1) ...
Setting up apport (2.20.11-0ubuntu82.10) ...
Installing new version of config file /etc/init.d/apport ...
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of restart.
Setting up yelp (42.1-lubuntu0.1) ...
Setting up language-pack-zh-hans (1:22.04+20240902) ...
Setting up language-pack-zh-hans-base (1:22.04+20240902) ...
Generating locales (this might take a while)...
Generation complete.
Setting up language-pack-gnome-zh-hans (1:22.04+20240902) ...
Setting up language-pack-gnome-zh-hans-base (1:22.04+20240902) ...
Processing triggers for mailcap (3.70+nmulubuntu1) ...

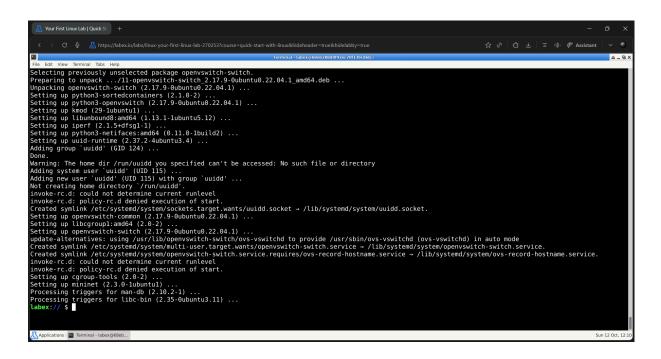
Processing triggers for desktop-file-utils (0.26-lubuntu3) ...

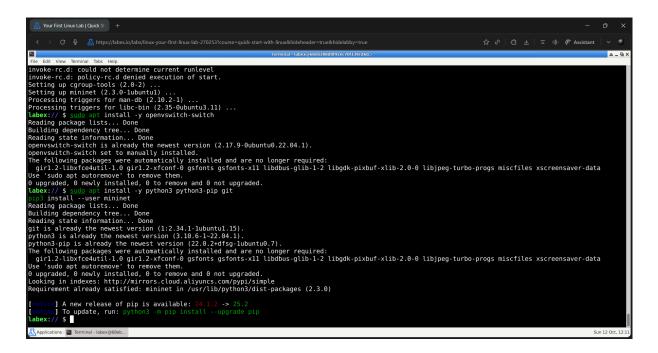
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for hibotof-iceme (0.17-2) ...

Processing triggers for libc-bin (2.35-0ubuntu3.11) ...

Processing triggers for man-db (2.10.2-1) ...

Processing triggers for plymouth-theme-ubuntu-text (0.9.5+git20211018-1ubuntu3) ...
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Processing triggers for ca-certificates (20240203~22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
 labex:// $
```





Conclusion:

Simulation of SDN using Mininet demonstrates how networks can be virtualized and centrally managed using controllers. This experiment shows host-to-host connectivity and highlights the role of the controller in defining packet forwarding behavior