

**BITS Pilani - Hyderabad Campus**  
**CS F303 (Computer Networks)**  
**Second Semester 2023-24, Lab Sheet 1**  
**Introduction to Mininet and Network Setup**

## 1. Overview

In this lab, you will deploy a virtual machine pre-installed with Mininet which is a virtualized environment to emulate hosts, switches and routers. You will design basic network setups and understand the impact of parameters like delay and packets losses in the network.

## 2. Mininet: A virtualized environment for networking

Some key points regarding Mininet:

- It is a network emulator and create a network of virtual hosts, switches, controllers, and links.
- It supports research, development, learning, prototyping, testing, debugging.
- It can be used to quickly create a realistic virtual network complete experimental network on a laptop or other PC. Specifically,
  - It provides a simple and inexpensive **network test bed**.
  - It enables **complex topology testing**, without the need to wire up a physical network.

## 3. Installation of Mininet on Ubuntu

- Prerequisites.
  - Python
  - Git
- Install Python:
  - Using *sudo apt-get install python3.6* (or higher versions).
  - Note that in lab, you don't need to install Python (you will see Python3.10. Use *python3 -V* to see the version of Python installed).
- Install Git: *sudo apt-get install git*
- Download and use Mininet using:
  - git clone <https://github.com/mininet/mininet.git>
  - *cd mininet*
  - *git tag #* list of available versions

- `cd util`
- `sudo ./install.sh -a`
- Verify the installation using `sudo mn` (this will automatically create a topology with two hosts h1 and h2 and a switch). See Figure 1 for the reference.
- Test whether the ping is working using “`sudo mn --test pingall`”

```

cclab@cclab-OptiPlex-3060:~/mininet/util$ sudo mn
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>

```

Figure 1 Illustration of “`sudo mn`” after Mininet is installed

#### 4. Mininet GUI and its components

You can open Mininet GUI and see the various components using:

- `cd ~/mininet/examples`
- `sudo python3 miniedit.py`
- See Figure 2

You can access the Mininet prompt in the terminal using `sudo mn`. Functionalities of different component can be read it from here: <http://mininet.org>

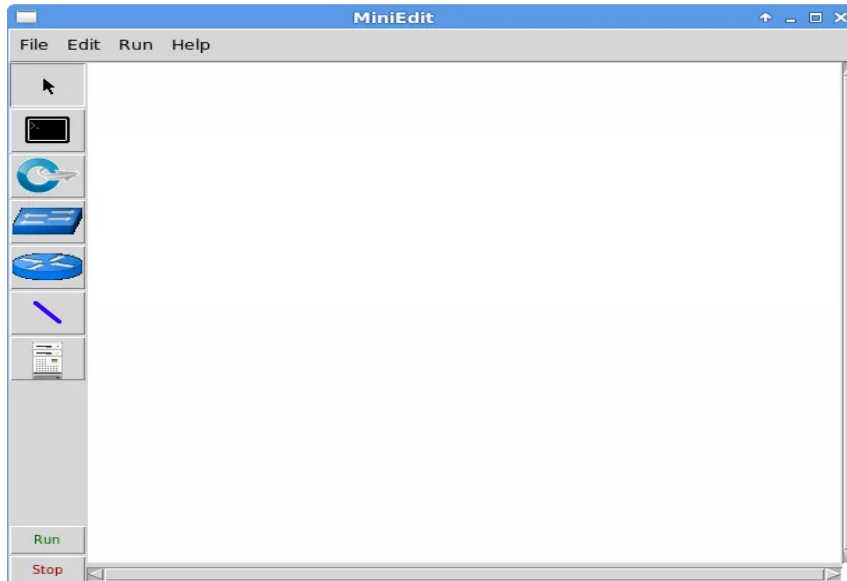


Figure 2 Mininet GUI

## 5. Lab Exercises

**Exercise 1:** Open GUI and create a topology with two hosts (h1 and h2) connected with a switch. Right click on the host icons, select properties, and set the hostname and IP address. Further select the “preferences” from the “Edit” on the top left and check the *Start CLI* box. Save the file and Run the setup using “run”. You can check the topology using the following commands:

- `net #` For checking the topology
- `h1 ifconfig #` checking the host system configuration
- `nodes #` For checking the nodes
- `dump #` listing of host and switch
- `h1 ping h2 (hostname ping hostname)`

**Exercise 2:** For the above topology (created in Exercise 1), configure the delay value for each link, i.e, from h1 to switch and switch to h2. Use Ping to find out:

- What is RTT and how will it change by configuring different values of delay in the links?
- Introduce packet losses in the two links and observe the output of the ping command.

**Exercise 3:** Create a topology with multiple switches between the two hosts. You can start with 4 switches between the two hosts. Configure the network setup and set delay and packet losses for each link. Observe the output by using Ping command and find out the differences as compared to the topology used in Exercise 1 and 2.

**Exercise 4:** So far you have set up the network topology using the GUI. Use the command line to set up the network topology. Practice the following:

- `sudo mn --test pingall --topo linear,4`
- `Sudo mn -topo single,3`
- `sudo mn --test pingall --topo linear,4`
- `sudo mn --link tc,bw=10,delay=10ms`