

HW 2 : Emulating networks using Mininet

14–740 Fundamental of Telecommunication Systems

Released: January 21, 2016, 11:59pm EDT

Due: January 28, 2016 11:59 pm EDT

1 Objectives

The objective of this homework is to get you to install and become familiar with Mininet – the network emulator we will be using in this course. In this homework you will be using Mininet to emulate different networks and test their performance. As we will discuss in the lecture, Mininet creates realistic virtual networks consisting of hosts running real kernels, switches and application code on a single machine. Mininet allows you to interact with your network through either the Mininet CLI interface or through Mininet Python APIs. Mininet also allows you to customize your network, share it with others, or deploy it on real hardware.

As part of this homework you will download, install and set up the Mininet VM on your laptop. After installing the Mininet VM you will create the topology specified below and run a few performance tests.

2 Installation

The first step in emulating network topologies will be to download and install Mininet VM on your laptops. This assumes that your laptop has a suitable virtualization program installed to open the Mininet VM. While you can install the Mininet VM to be opened with VMWare Fusion or VirtualBox or qemu (for Linux), we highly recommend using VirtualBox.

The instructions for downloading and installing the Mininet VM are described in detail at <http://mininet.org/download/>. We highly recommend that you use "Option: 1" and follow all the steps listed in the section until and including Walkthrough step. MacOS users may have to download and install the X11.app from the XQuartz project to enable X forwarding to work.

3 Testing Your Setup

Once you are done installing your Mininet VM, you can test it by running some of the commands we will discuss during Lecture 4 and trying out different topologies through the Mininet CLI and APIs. The python scripts corresponding to some of the tests will be uploaded on Blackboard so that you can understand and modify them based on your requirements. You can go through additional documentation provided at <http://mininet.org> to get comfortable with creating custom topologies and tests.

4 Testing a Custom Setup

To ensure that you have been able to install the Mininet VM and have learned the Mininet API, you will need to emulate the fairly trivial un-balanced tree topology as shown in Fig. 2. The numbers on the edges between switches represent the link properties – bandwidth (in Mb/s), delay (in ms) and link loss rate (in % of packets lost). The links between the hosts and switch ports can be left with the default properties assigned to them by Mininet.

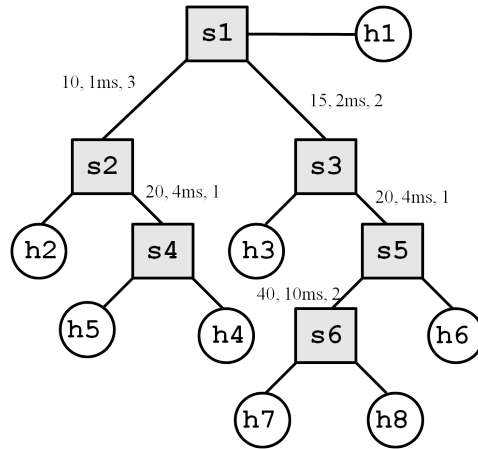


Figure 1: An unbalanced tree topology

4.1 Deliverables

For this homework, you will perform the following tests

1. Connectivity test between all the hosts by sending 10 ping messages between each other.
2. Measure the TCP bandwidth between every pair of hosts
3. Measure the UDP packet loss between every pair of hosts at a bandwidth of 15 Mb/s.

You will upload your Python script along with a word document that has the output from all the tests that were conducted on Mininet.