Mininet Network Emulator

I.T. de Villiers - 17502292
Study Leader: Dr. H.A. Engelbrecht
University of Stellenbosch

Introduction

The Purpose of this project, was to evaluate Mininet Network Emulator, for the purpose of Emulating various network conditions like: **bandwidth limitation**, **packet loss**, **network delay**, **jitter**.

Mininet is a network emulator that creates virtual networks using virtual hosts, switches links and controllers. Mininet connects virtual nodes using Virtual Ethernet (veth) pairs, and allows communication between virtual devices using the OpenFlow protocol.

Aims & Objectives

Evaluate the performance of Mininet as a network emulator by;

1. Creating a Graphical User Interface (GUI) that allows a user to;
   (a) Set up custom, paramaterized Mininet network topologies quickly and easily
   (b) Save custom Mininet topologies
2. Creating performance tests to;
   (a) Generate performance graphs of bandwidth-,delay-,packet loss- and jitter consistency between Mininet hosts over a (i) Software interface and a (ii) Hardware interface.
   (b) Test Mininet's scalability
3. Compare results of the emulated network, with a real network

Experimental Approach

1. Create a GUI to enable customization of test networks
2. Test Mininet's performance on a Software Platform
3. Test Mininet's performance qualitatively
4. Test Mininet's performance on a Hardware Platform
5. Test Mininet's ability to emulate a real network

Graphical User Interface

The GUI was designed using tkinter. Using Python's GUI toolkit simplified front-end to back-end integration. The GUI allows the user to specify the network Topology and set various network conditions, like bandwidth, delay, packet loss and jitter. the main GUI window consists of four widget frames; the **Hosts Widget**, **Switches Widget**, **Links Widget** and the **Visuals Widget**. These widgets allow the user to add hosts and switches to the network and to customize link parameters.

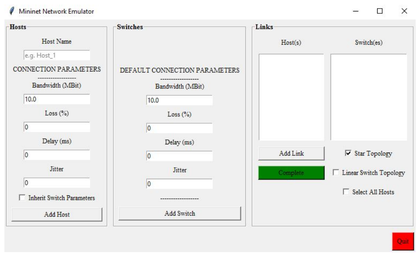

Figure 1: GUI Main Menu

Software Platform Experiment

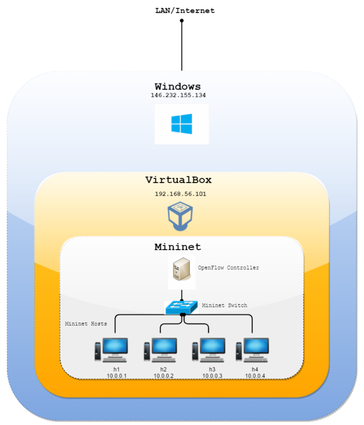

Figure 2: Software Platform setup overview

The purpose of this experiment was to test Mininet's ability to accurately emulate user defined network effects like bandwidth, packet loss, delay and jitter, on a **Software Platform**. Iperf TCP & UDP data stream tests were used to measure the consistency between user defined network effects and actual measured effects as the size of the network increased. A further qualitative test was conducted by using VLC to stream a video from one virtual host to another.

Hardware Platform Experiment

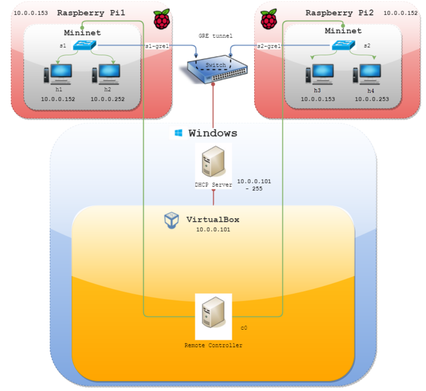

Figure 7: Hardware Platform setup overview

The purpose of this experiment was to test Mininet's scalability and its ability to accurately emulate user defined network effects like bandwidth, packet loss, delay and jitter, on a **Hardware Platform**. Iperf TCP & UDP data stream tests were used to measure the consistency between user defined network effects and actual measured effects as the size of the network increased.

Mininet Network vs. Real Network

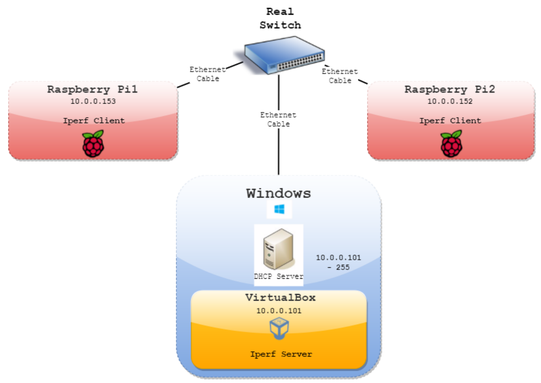

Figure 12: Physical network setup overview

The purpose of this experiment was to test Mininet's ability to accurately emulate a physical network's network conditions. Iperf TCP & UDP data stream tests were used to measure the network conditions of the physical network. The exact network was set up within Mininet and Iperf tests were then used to measure the actual emulated network conditions.

Results

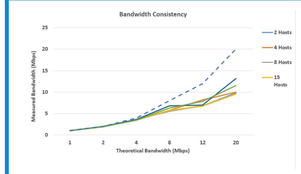

Figure 3: Bandwidth Consistency

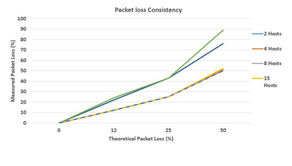

Figure 4: Packet loss Consistency

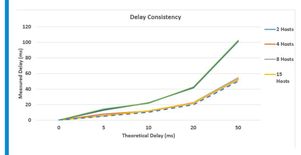

Figure 5: Delay Consistency

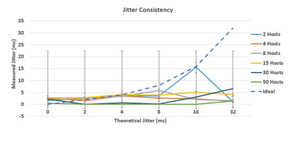

Figure 6: Jitter Consistency

From Figures 3 - 6: Mininet is somewhat limited on a Software Platform. Due to working in a completely virtual environment, Mininet cannot produce the necessary resources for each of its virtual hosts, causing emulated network effects to be inaccurate.

1. Accurate Bandwidth emulation restricted to 5 Mbps
2. Accurate Packet loss emulation for small networks (<20 Hosts)
3. Accurate Delay emulation for small networks (<20 Hosts)
4. Measured Jitter values inaccurate and inconsistent
5. A video stream test confirmed emulated network effects to be qualitatively accurate

Results

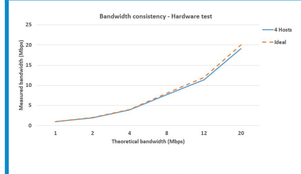

Figure 8: Bandwidth Consistency

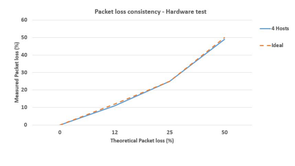

Figure 9: Packet loss Consistency

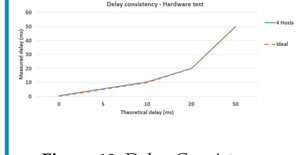

Figure 10: Delay Consistency

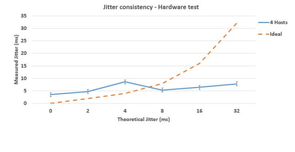

Figure 11: Jitter Consistency

From Figures 8 - 11: Mininet performs better on a Hardware Platform. Due to working in a physical environment, Mininet could produce the necessary resources for each of its virtual hosts, resulting in accurate emulation of network conditions.

1. Accurate Bandwidth emulation
2. Accurate Packet loss emulation
3. Accurate Delay emulation
4. Measured Jitter values consistent but inaccurate

Results

Network Effect	RP 1	RP 2
Bandwidth	94.3 Mbps	94.3 Mbps
Packet Loss	0 %	0 %
Delay	0.889 ms	1.173 ms
Jitter	0.064 ms	0.201 ms

Table 1: Real Network

Network Effect	RP 1	RP 2
Bandwidth	63.1 Mbps	60.56 Mbps
Packet Loss	0 %	0 %
Delay	2.59 ms	2.16 ms
Jitter	0.215 ms	0.216 ms

Table 2: Mininet Network

Conclusion

1. Mininet is an accurate and scalable network emulator
2. Network effects are qualitatively accurate
3. Emulation performance is restricted by the host PC's resources
4. Mininet can therefore not emulate large networks (>20 Hosts) on a software platform
5. It is therefore recommended to run Mininet on a hardware Platform