Anthony Feudale

Project Proposal CS557

Title: Investigating the Performance of Last In First Out (LIFO) as a Network Queuing Mechanism

Participant: Anthony Feudale

Project Type: Option 1, Investigation

Project Summary: The goal of the project is to investigate the usage of LIFO queueing as a network queuing mechanism, in comparison to the more commonly used First Come First Serve (FCFS) protocol. If time allows a random queueing mechanism may be investigated as well. By random queuing, I mean packets will be queued in the order they arrive and will subsequently be dequeued via random selection.

The primary means of investigation will be done via simulation of network traffic. I had initially thought to use Mininet to accomplish this, but discovered that the queueing mechanics are built deep into the Linux OS and changing them would be beyond the scope of a simple term project. Instead, I will utilize the Simpy python package to simulate network traffic as discrete events. I will be making use of some existing network simulation code distributed by Grotto Networking under the MIT license to serve as a framework for simulating network traffic, modifying code as needed.

I had initially thought to recreate an entire TCP, FTP, or UDP protocol between endpoint nodes. Upon reflection I decided this would be both too limiting and too time consuming. Given the tight time constraints of the project I would likely only have the bandwidth (pun intended) to work out one protocol, thus missing out on investigating the others. Instead, I will utilize shaped random stochastic distributions to model different types of traffic flows through the network. Each packet emitting node sending data to a router will have two knobs to tune: packet size and rate of packet sending. By fitting these knobs to a carefully chosen distribution, I will be able to model different traffic types. For example, I could monitor bursty traffic by choosing distributions which usually send infrequent small packets of data, but occasionally send long strings of very large packets. I intend on mocking out

bursty, normally-distributed, steady-state, and exponentially-distributed traffic shapes. I may include others if research into network traffic shapes reveals promising patterns.

By mixing and matching nodes with different packet generation patterns together on the same network, I should be able to create a variety of model networks to compare and contrast the different queueing mechanisms. I will draw inspiration for which sorts of networks to model from other papers, most notably our *Analysis and Simulation of a Fair Queueing Algorithm* paper.

Goals:

- Create LIFO and FIFO queue routed network traffic simulation code: accomplished using Simpy and Grotto Networking code and PyCharm IDE
- Create traffic shaped distributions: Accomplished using the Random python module and above code
- Model various traffic scenarios and collect data: Accomplished using above code and matplotlib
- Gather previous research: Accomplished using Papers read in class and their references, as well as Google Scholar and CSU library search
- Write Paper in IEEE format recording findings of experiments: Accomplished using LibreOffice and IEEE template
- Create presentation and recording: Accomplished using LibreOffice and Windows Video Editor

Schedule

- March 27th– April 5th: Create necessary simulation python code, gather initial research, begin running simulations
- April 5th– April 17th: Complete gathering of research, finish simulations, begin writing paper
- April 17th – April 28th: Finish writing paper integrating findings from project, complete presentation and recording

References: https://www.grotto-networking.com/DiscreteEventPython.html