

Assignments for Networking and Operating Systems Classes Using GENI

Zongming Fei

University of Kentucky

Outline

- Introduction
- A List of Possible Projects
- Three Assignments
 - Path Characteristics of the Internet
 - Reliability Protocols
 - Network Configurations and Automatic Route Control
- Experiences and Lessons

Introduction

- Early adopter of Emulab
 - Build the Kentucky Emulab and started using it in teaching undergraduate courses.
 - An emulated experiment allows you to specify an arbitrary network topology, giving you a *controllable, predictable, and repeatable environment*, including PC nodes on which you have *full "root" access*, running an operating system of your choice. (from Utah Emulab site)
- Work with Calvin College on the Edulab project
 - A joint project funded by NSF (Jim Griffioen and Zongming Fei from Kentucky and David Laverell from Calvin College) to build Edulab
 - enhance Emulab facilities for educational use
 - set up a smaller scale edulab at Calvin College
 - develop assignments based on Dr. Comer's Hands-on Networking Labs

The Edulab Project

- A low budget Emulab at Calvin (\$15,000 in 2006)
 - 26 PCs (one as boss and one as ops)
 - Two Cisco 2950T switches
 - A Rocketport serial card
 - Web power switches from Digital Loggers
- Simplified project management
 - Create accounts
 - Monitor node usage
 - Schedule project
- Status Monitoring
 - Status of nodes and links (Time scale)
 - Take snapshot of an experiment

GENI Instrumentation and Measurement Tools (INSTOOLS)

- Project Team: Jim Griffioen, Zongming Fei, Hussam Nasir, Xiongqi Wu, Jeremy Reed, Charles Carpenter
- Set up measurement infrastructure for users in ProtoGENI control framework automatically
- Monitor node status and link status
- Develop graphical user interface to display the status data as they change over time
- Implement archive services to store measurement data permanently
- Integrated with ProtoGENI Flack interface

ProtoGENI Flack Interface

The screenshot displays the ProtoGENI Flack Interface. The window title is "Generate request RSPEC: Started" and the version is "v13.32". The interface is divided into several sections:

- Map:** A vertical sidebar on the left containing:
 - Slices:** A dropdown menu set to "uky.emulab.net" and a "Create slice..." button.
 - Managers:** A list of managers with checkboxes for "Show/Hide". Checked managers include cmulab.cm, cron.loni.org.cm, ETRI-CM1.cm, jonlab.cm, mygeni.cm, plc.sa, shadowgeni.cm, ukgeni.cm, utahemulab.cm, utahemulab.cm, utahemulab.cm, wail.cm, and wall3geni.cm.
- testslice:** A central panel with a "View" dropdown and an "Import" button. It lists various slices with icons for configuration and status:
 - bbn-pgeni.cm
 - beelab.cm
 - cis.fiu.edu.cm
 - cmulab.cm
 - ETRI-CM1.cm
 - jonlab.cm
 - plc.sa
 - shadowgeni.cm
 - ukgeni.cm
 - utahemulab.cm
 - utahemulab.cm
 - wail.cm
 - wall3geni.cm
- Diagram:** A network diagram on the right showing a central node labeled "PC" (excl. raw-pc) connected to two intermediate nodes (each with a red 'X' and an 'i' icon). These intermediate nodes are connected to two VM nodes labeled "VM" and "VM-0", both associated with "sh.emulab-o-penvz". A link between the VM nodes is labeled "100 Mb/s".
- Output:** A dropdown menu set to "GENIv3".
- Submit:** A button at the bottom right of the interface.

INSTOOLS

The screenshot displays the 'testslice' web interface. On the left is a control panel with the following elements:

- Buttons: View, Import, and a gear icon (labeled '1' with a red arrow).
- Dropdown menu: INSTOOLS.
- Buttons: Read Tutorial and Instrumentize (labeled '2' with a red arrow).
- Dropdown menu: APIv 1.
- Radio buttons: Raw MCs (selected) and Virtual MCs.
- Button: Go to portal (labeled '3' with a red arrow).
- Button: Submit.

The main area shows a network diagram with the following components and connections:

- Top node: 'excl.raw-pc' (PC icon).
- Two intermediate nodes: Two boxes with 'X' and 'i' icons, connected to the top node.
- Bottom nodes: 'sh.emulab-openvz' (VM icon) on the left and 'sh.emulab-openvz' (VM-0 icon) on the right.
- Connections: The two intermediate nodes connect to the top node. The left intermediate node connects to the left VM. The right intermediate node connects to the right VM. The two VMs are connected to each other via a link labeled '100 Mb/s'.

INSTOOLS

The screenshot shows a web browser window with the following elements:

- Browser Interface:** File Edit View History Bookmarks Tools Help menu bar. Address bar: https://portal.uky.emulab.net/geni/portal/map_network.php. Search engine: Google.
- Tabs:** ProtoGENI-Trac, Portal myslice1, Config | myslice1, http://pc6... Config.php.
- Main Content:** A map of the United States with a network overlay. Nodes are marked with purple squares and labeled: Utah1 (California), Utah2 (Utah), Uky1 (Kentucky), Uky2 (Missouri), Uky3 (Alabama), MC Cutahemulab (Wyoming), and MC Cukger (West Virginia). Blue lines connect these nodes, forming a network topology.
- Left Panel:** Three performance graphs:
 - POW50-1 Total CPU Utilization: Multi-Core / Multi-Processor (0.0 to 2.0). Legend: User CPU Usage (green), System CPU Usage (blue).
 - POW50-1 Top Traffic (0 to 20). Legend: Incoming Traffic (green), Outgoing Traffic (blue).
 - POW50-3 Top Traffic (0 to 20). Legend: Incoming Traffic (green), Outgoing Traffic (blue).
- Right Panel (myslice1):** Logout, Archive, Overview, Collapse, and search boxes for Utah1, Utah2, MC Cutahemulab, Uky3, Uky2, Uky1, and MC Cukger.

INSTOOLS

The screenshot displays the INSTOOLS web interface. At the top, there is a browser window with several tabs, including 'ProtoGENI - Trac' and 'Portal myslice1'. The main content area shows a map of the United States with a network overlay. Nodes are labeled 'Utah1', 'Utah2', 'Uky1', 'Uky2', and 'Uky3'. The 'Utah2' node is selected, and a configuration panel is open for it, listing various traffic and graph types with checkboxes. The panel includes options like 'cpugraph', 'icmpttraffic', 'iptraffic', 'memgraph', 'tcpgraph', 'totaltraffic', 'udpgraph', 'lan2 linkoctets', 'lan2 linkpackets', 'gre-tunnel1 linkoctets', 'gre-tunnel1 linkpackets', 'gre-tunnel2 linkoctets', and 'gre-tunnel2 linkpackets'. Below the configuration panel are buttons for 'VNC' and 'All Graphs'. On the left side, there are several graphs showing traffic and resource usage for different components, including 'PCVM50-3 UDP Traffic', 'PCVM50-3 TCP Traffic', 'PCVM50-3 Interface Traffic', 'PCVM50-2 Total CPU Utilization', 'PCVM50-3 Memory Usage', and 'PCVM50-2 CPU Traffic'. On the right side, there is a sidebar for 'myslice1' with options like 'Logout', 'Archive', 'Overview', and a list of nodes to search for.

INSTOOLS

The screenshot shows a web browser window with the following details:

- Browser:** Mozilla Firefox
- Address Bar:** `pc61.uky.emulab.net/drupal/?q=pc/pcvm50-2/dtype/Node/graphs`
- Page Title:** myslice1 (Live View)
- Navigation:** Home > ProtoGeni > Uky2 (PCVM50-2) > Graph > Node Graphs for pcvm50-2
- Left Sidebar (User: fei):**
 - My account
 - ProtoGeni
 - ProtoGeni
 - Archive
 - Config
 - S3
 - Uky1 (PCVM50-1)
 - Uky2 (PCVM50-2)
 - Uky2 (PCVM50-2)
 - Graph
 - Graph
 - Lan0
 - Lan1
 - Node
 - Table
 - VNC
 - Uky3 (PCVM50-3)
 - Create content
 - Administer
 - Log out

The main content area displays three traffic graphs for 'Node Graphs for pcvm50-2':

- IP Traffic:** Shows traffic in packets. Incoming (green) and Outgoing (blue) traffic are plotted. The y-axis ranges from 0 to 20. The x-axis shows time from 16:35 to 16:40.
- TCP Traffic:** Shows traffic in packets. Incoming (green) and Outgoing (blue) traffic are plotted. The y-axis ranges from 0.0 to 8.0. The x-axis shows time from 16:35 to 16:40.
- UDP Traffic:** Shows traffic in packets. Incoming (green) and Outgoing (blue) traffic are plotted. The y-axis ranges from 0 to 20. The x-axis shows time from 16:35 to 16:40.

A List of Possible Projects

- GENI supports at-scale networks
 - The number of resources offered
 - The types of resources offered
 - The geographical scope of the resources offered
 - The speed/performance of the resources offered
- An incomplete list of projects (or types of projects)
 1. Conventional OS/networking assignments
 - Make modifications to existing OS and networking code to create their own protocols and network services
 2. Network monitoring assignments
 - Write active and passive monitoring code to measure the performance of the Internet

A List of Possible Projects

3. Data center/cloud assignments

- Use GENI high-performance clusters to implement data center services with custom or conventional data center software

4. Wireless and/or mobile assignments

- Implement services using a variety of wireless and mobile resources and technologies supported by GENI

5. Home networking assignments

- Write home services/applications utilizing the resources of opt-in home users in GENI

A List of Possible Projects

6. High-performance networking assignments
 - Make use of GENI high-performance servers to test the scalability with regard to performance
7. Application-level monitoring assignments
 - Use low-power radar sensors and web cameras that are virtualized and accessible to users and the high-performance network links to move data off of these devices to network servers in real-time
8. Complete network assignment
 - Design and implement a complex/complete network ranging from mobile client nodes connected via wireless links to an optical backbone networks with advanced services built into the network structure, as well as data center computing power offering cloud services

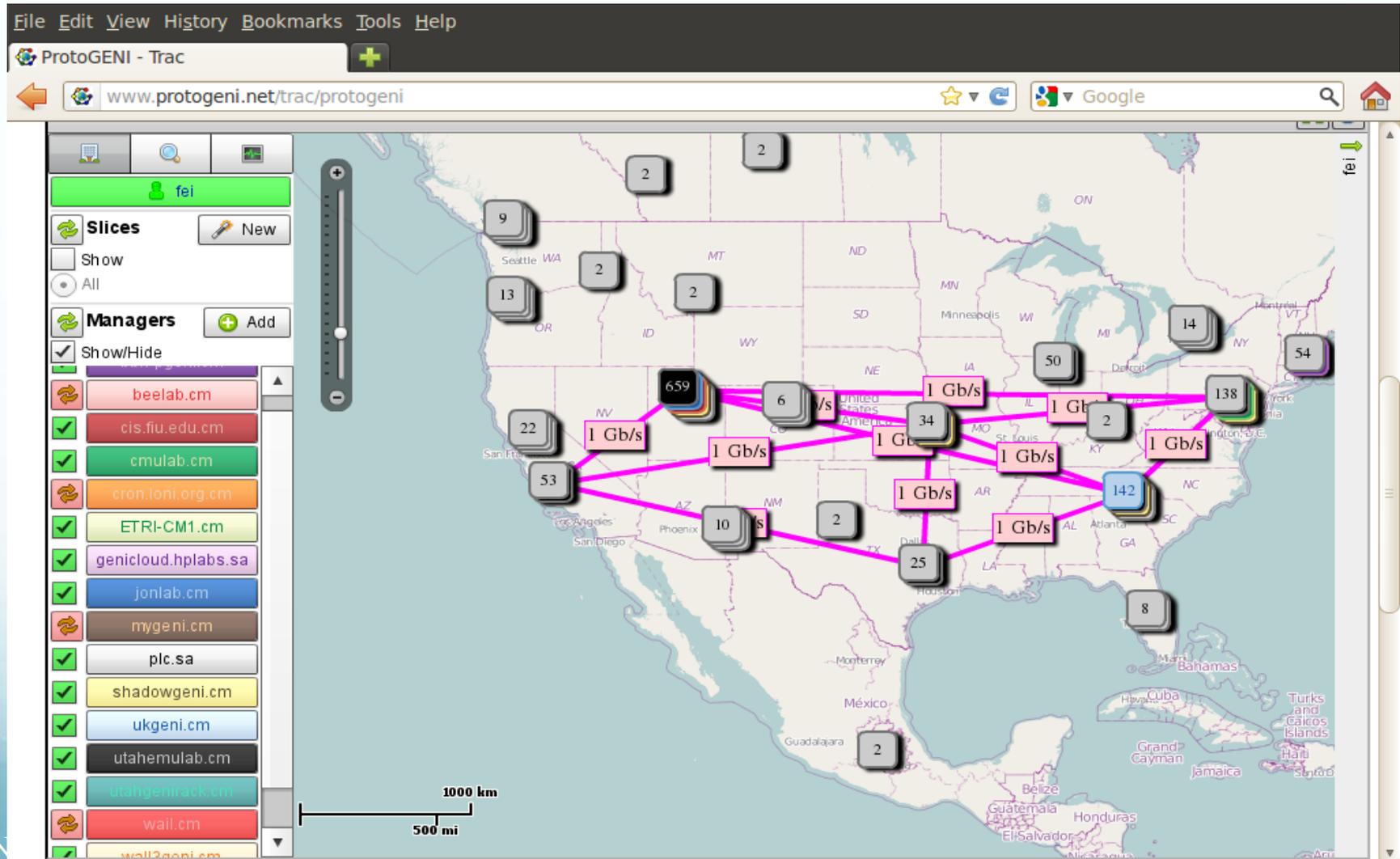
Assignment 1 --Path

Characteristics of the Internet

- Measure delay, bandwidth and loss rate of the links over the Internet and analyze their distributions
- Questions to be answered
 - What is the typical delay of a path from the east coast to the west coast?
 - How do delay, bandwidth and loss rate differ between a local link and a wide area link, or among different wide area links?
 - Do they change a lot over time? What is the distribution?
- GENI vs Guest accounts

Assignment 1 --Path

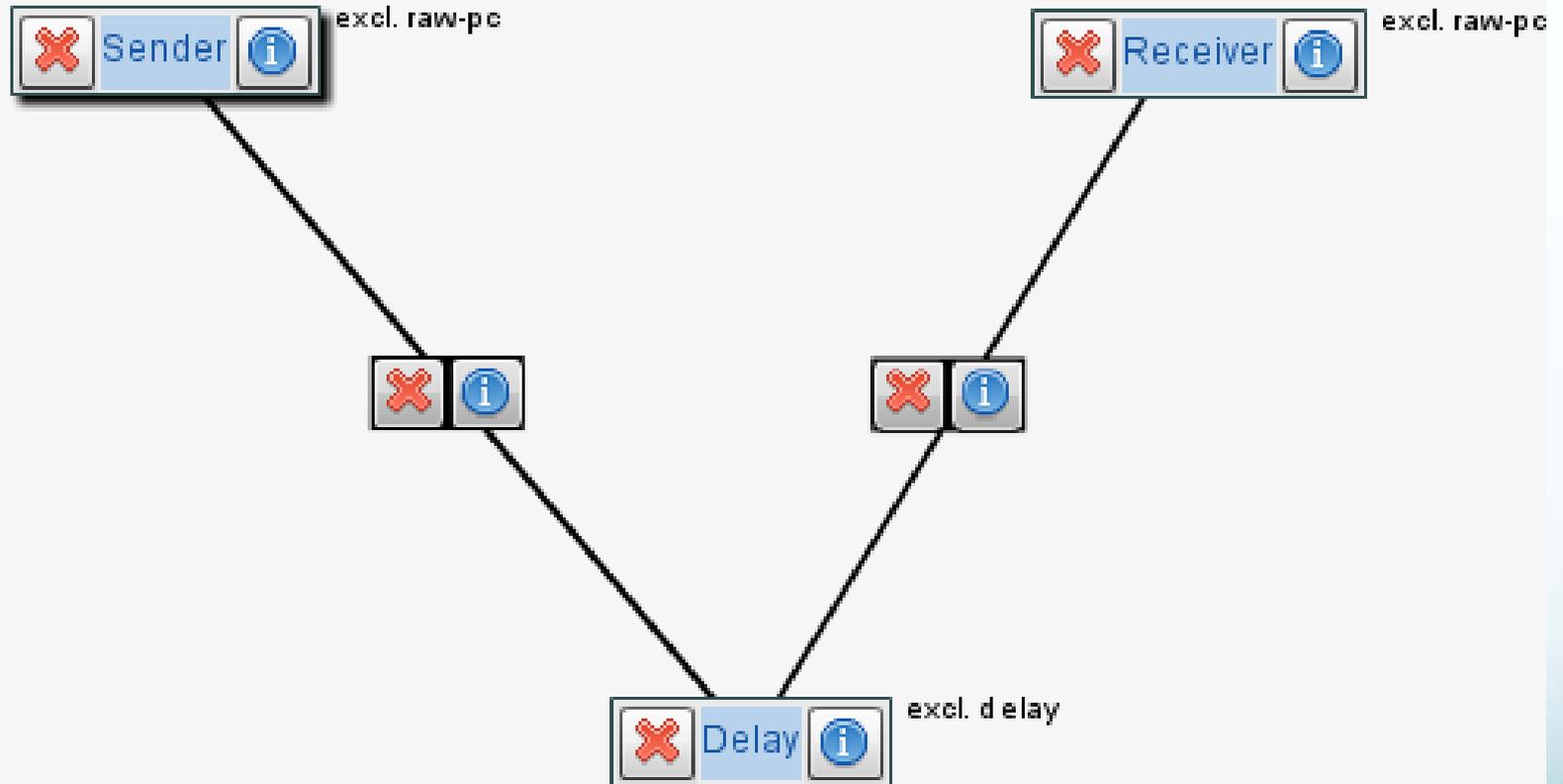
Characteristics of the Internet



Assignment 2 --Reliability Protocols

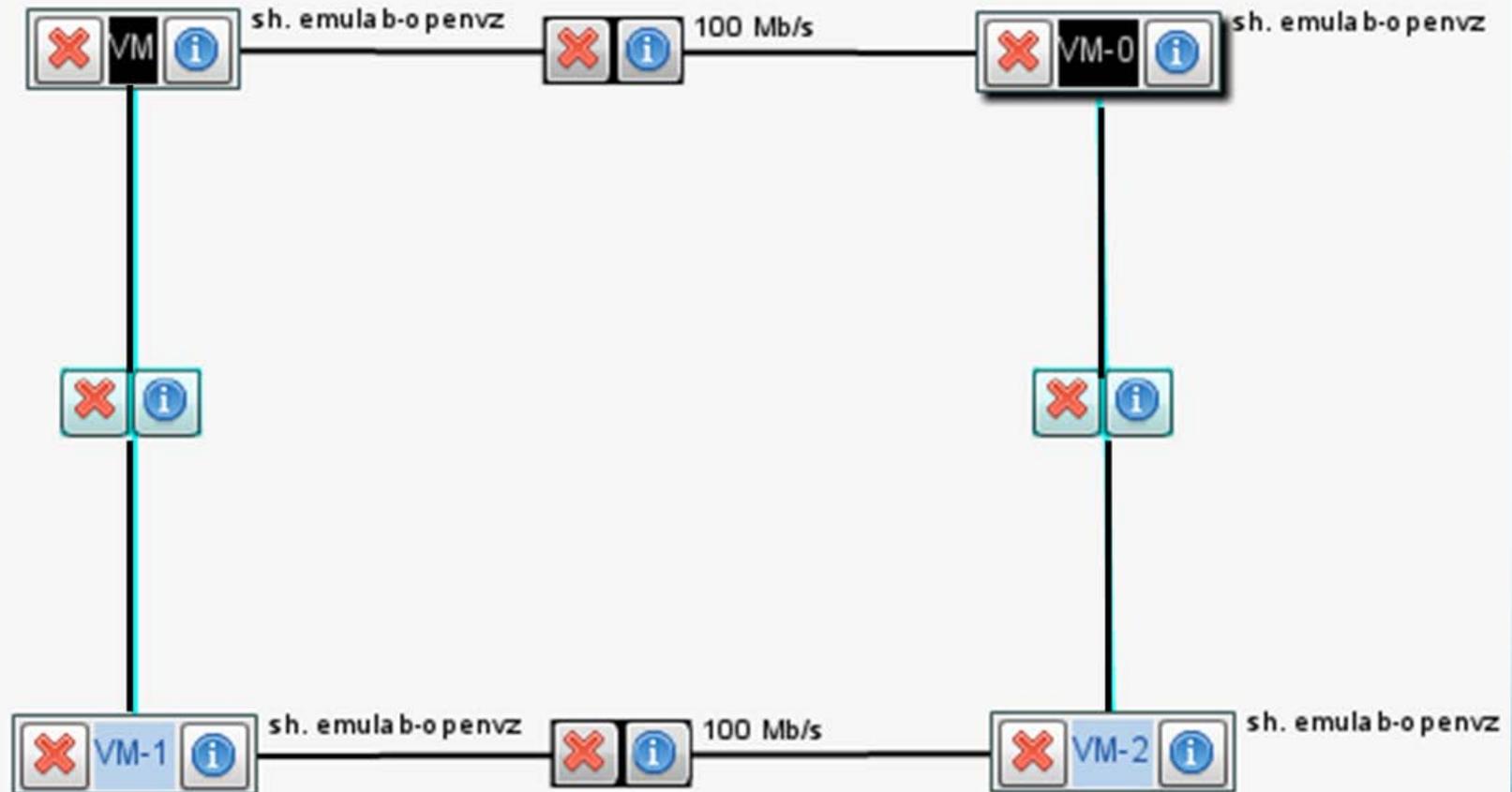
- Implement reliability protocols at the application layer using UDP.
 - Stop and wait
 - Go back N
 - Selective repeat
 - TCP reliability
- Almost no loss in a traditional general purpose lab
- Need a repeatable, predictable and easy to control environment with a certain loss property
- GENI provides an environment to do that

Assignment 2 --Reliability Protocols



Assignment 3 – Network Configuration and Automatic Router Control

(thanks to Dr. Jim Griffioen)



Assignment 3 – Network Configuration and Automatic Router Control

- Manual configuration of network routing using conventional network administration software
- The goal is to gain experience with the basics of IP, ARP and routing
- Types of commands used
 - Discovering network settings
 - `/sbin/ifconfig`, `netstat -rn`, `/sbin/arp`
 - Set routing tables, and enable forward
 - `/sbin/route`,
 - `echo 1 > /proc/sys/net/ipv4/ip_forward`

Assignment 3 – Network Configuration and Automatic Router Control

- Implement a new router that separates routing from forwarding
- Write a Forward Information Base (FIB) controller at each router
 - accept commands from RD services below
 - modify routing tables at the router (execute route by calling `system()` or `exec()`)
- Write a Routing Decision (RD) server
 - make decision about the routing (e.g., multi-path routing, QoS routing)
 - send commands to FIB controllers
- Communications between FIB controllers and the RD server follow FIB controller protocol (FCP)

Experiences and Lessons

- Positive
 - Use cool and most advanced technology
 - Keep pace with most recent development in the real world
 - Easy setup and user friendliness of the tools
- Improvements needed
 - Set up the keys and access to the nodes
 - Distinction between experiment network and control network
 - Which part goes wrong? (not a production software)
- Lesson: Importance of step-by-step instructions

Thank You!