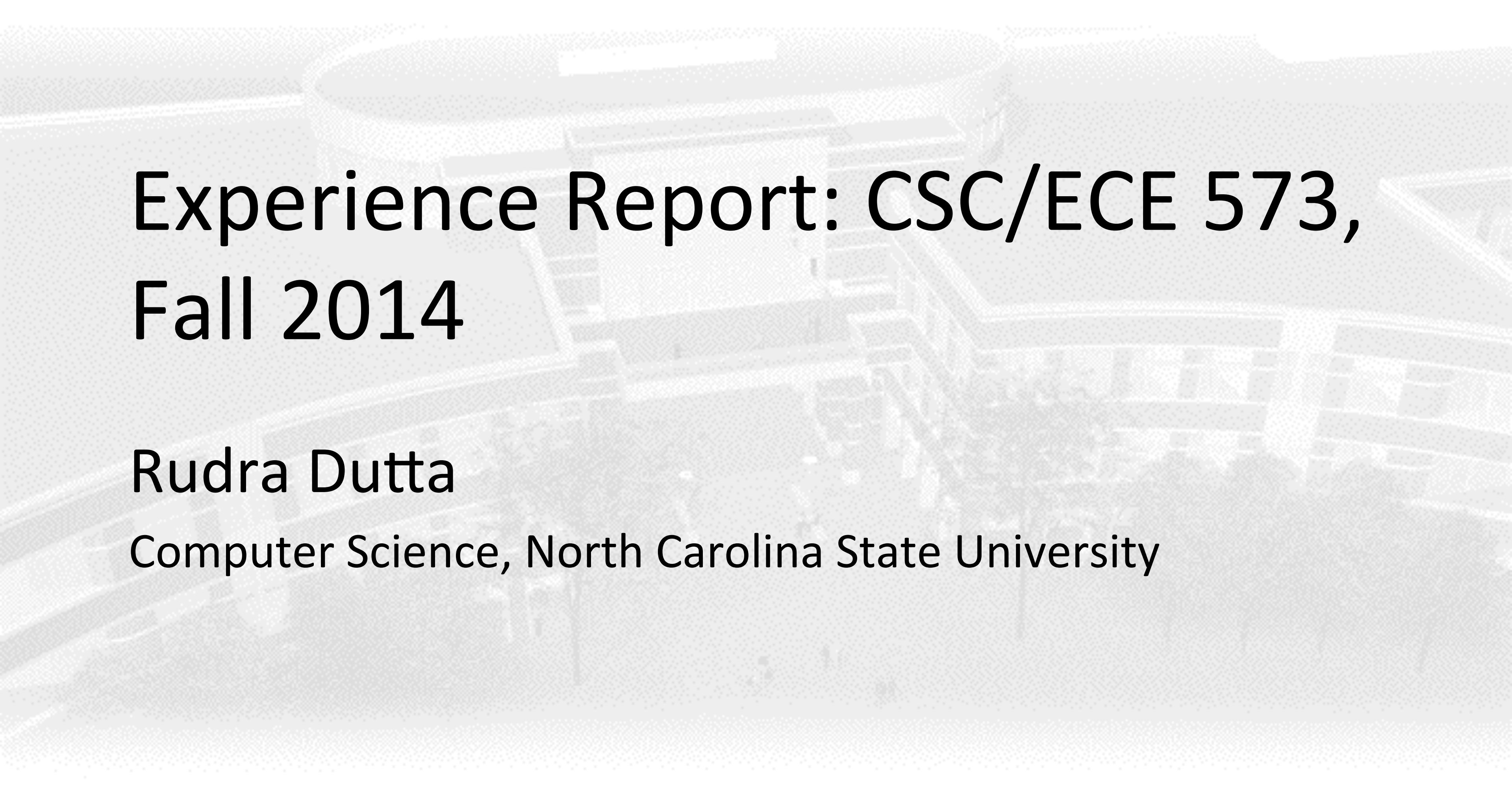


Experience Report: CSC/ECE 573, Fall 2014

Rudra Dutta

Computer Science, North Carolina State University

Historical

- Previous experience as GENI-IMF project lead
  - All students have long since graduated
  - Continue using GENI for research
- Some previous smaller attempts at using GENI as a teaching environment
  - Fall 2012: Class of 50+
  - Spring 2013: Small (12) specialized seminar class
  - Even older attempts, also mostly small classes
- For us, grads are easier than undergrads

Course description

- CSC/ECE 573: Internet Protocols
- Intermediate graduate course
  - Assumes introductory networking course, TCP/IP, socket programming, and general knowledge of common Internet functionality such as HTTP, …
  - Refresher
  - Forwarding engine concepts – buffering, scheduling, QoS in forwarding
  - Netfilter kernel modules, Click, routing, SDN, architecture
- Half lectures and individual homeworks, half group project (proposal, interim and final report, demo)

Prep – Thinking of Strength/weakness

- Using instruments in teaching, in general
- Two possible goals
  - Teach underlying concept or something
  - Teach use of instrument
  - Usually some of both
- With respect to first goal, instrument can help or hinder
  - Even worse when instrument interfaces are complex
  - Even worse when they are virtualizations of
  - Even worse when behavior may be erratic or non-repeatable
- Scale can be an advantage – or a trump

Decisions

- Use a mix of paper, real lab, local virtual (single VM), GENI for as many things as possible
  – Fortunate to have fairly large lab, and campus cloud
- Use ExoGENI
  – In my limited experience, more stable and more predictable
  – Also, good working relationship with support folk
  – Might be just biased

Homeworks and Projects

- Six homeworks assigned over first 2/3 of semester
  - All but HW6 were individually due
  - HW6 was group (same group as project, instructor-assigned)
- HW3 had 2 of 10 questions on GENI
- HW4 had 1 of 4 questions on GENI
- HW6 was a single GENI question
- Group projects

Prep

- Train the TA
  - Also in-house help from GRAs with previous GENI experience
- Inform GENI folks
- Design exercises with gradability in mind
  - Same as traditional lab environment: grade products, but be prepared to go to experiment if needed
- Big-picture presentation in class

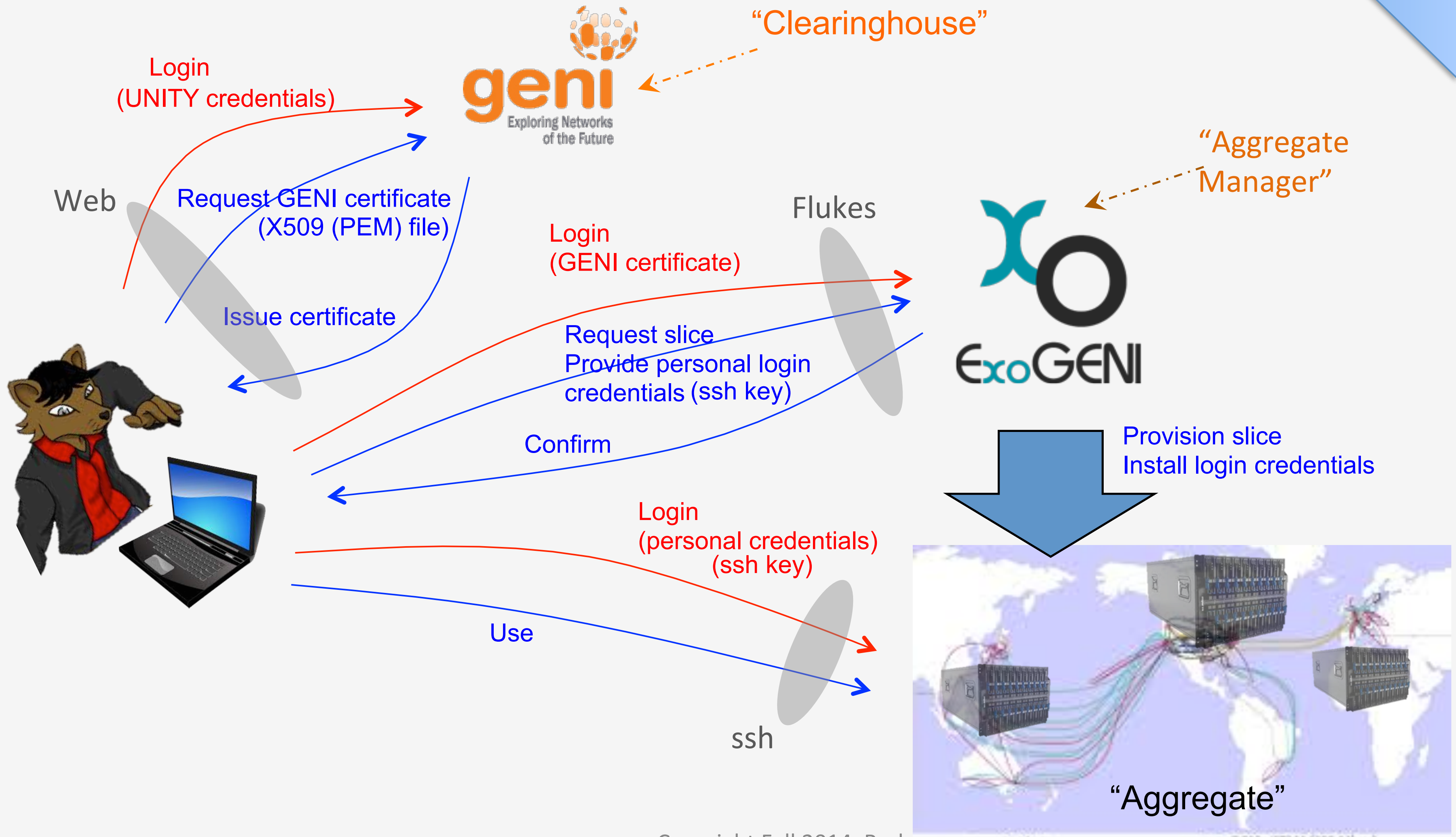
Using GENI (First steps)

- Must complete, in HW3:
  - Configuration and setup
  - A simple exercise
- Must be member of a “GENI project”
  - Request to become member of “ncsu_teaching”
  - Receive approval from instructor
- Now, you can log into the GENI portal
  - Visit portal.geni.net, click “Use GENI”
  - This is the authentication step (you are who you say you are)
- GENI uses a single sign-on system with federation
  - Shibboleth will bring up a list of authentication (login) domains
  - Choose “NC State University” if not already pre-selected
  - Log in with your UNITY ID/PW

Creating a Slice

- GENI returns, upon request, a slice certificate
  - This is the authorization to use resources from aggregates
- Now, you can create "slices"
  - A "slice" is a virtual network
  - Each "node" is a VM, on which you have root access
    - Can use for any processing, including forwarding packets
  - Each "link" is a virtual circuit, L2 (VLAN) or L3 (OF)
    - You should not have to understand distinction, or allow for any differences
- GENI provides:
  - Various aggregates (see previous slides from GPO)
  - Various tools to view and access those aggregates
  - All tools access all aggregates, but some pairs match better
- We shall use ExoGENI aggregates with Flukes
  - This combination supported some extra functionalities – we shall try to keep within GENI set

Creating and Using Slices



"Clearinghouse"

Login
(UNITY credentials)

Web

Request GENI certificate
(X509 (PEM) file)

Issue certificate

Flukes

"Aggregate
Manager"

Login
(GENI certificate)

Request slice
Provide personal login
credentials (ssh key)

Confirm

Provision slice
Install login credentials

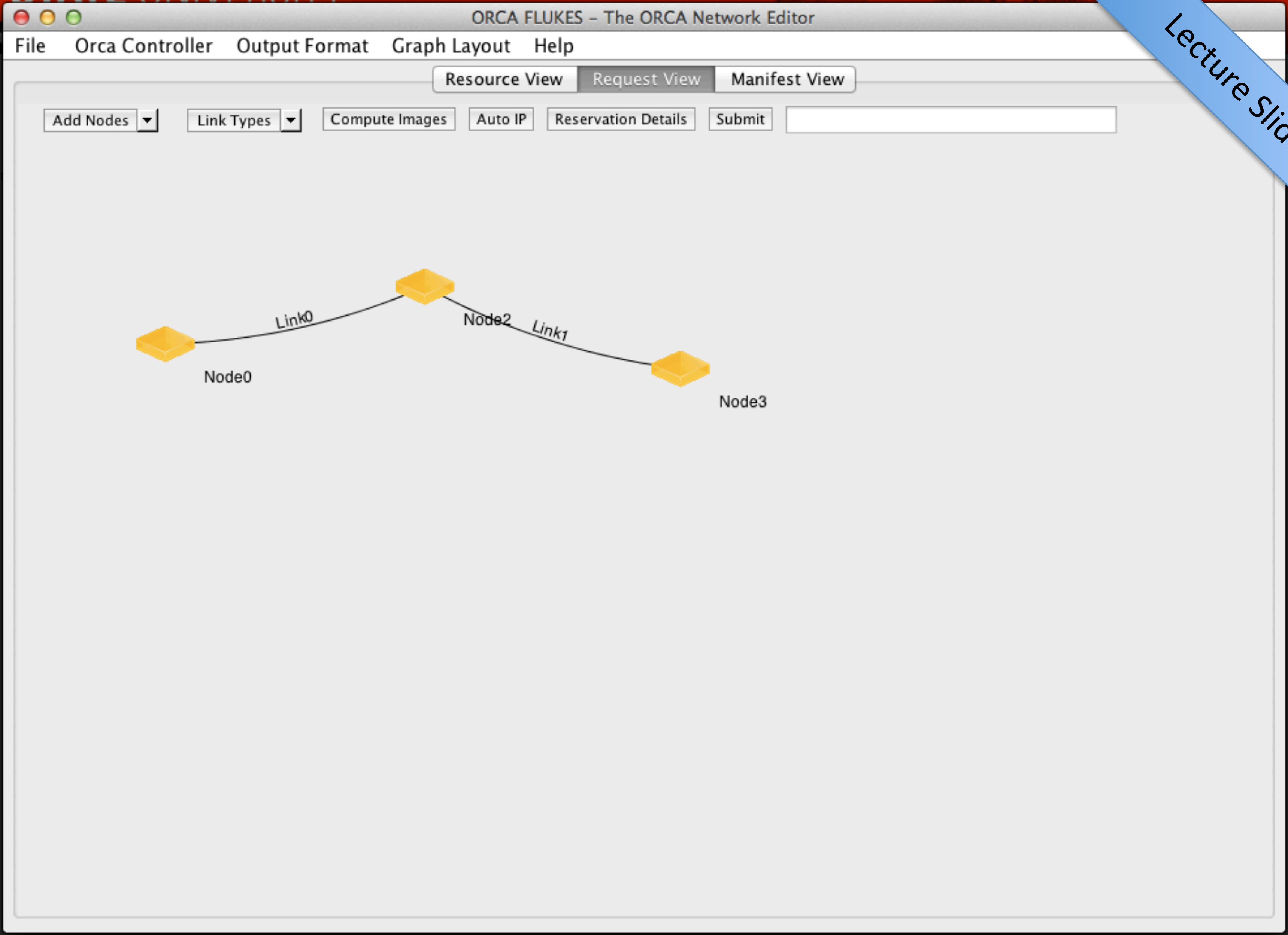
Login
(personal credentials)
(ssh key)

Use

ssh

"Aggregate"

Lecture Slide

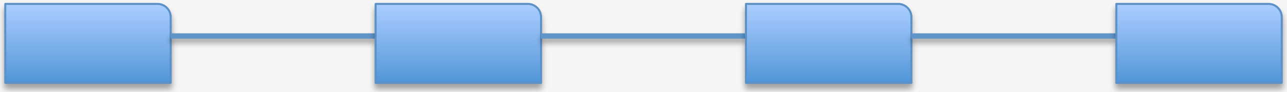


Create and Use Your Slice

- The previous step only *designed* the slice
  - Still just a picture on a screen – no actual resources
- “Submit” this design to be instantiated
  - Should succeed if you have authorization, aggregate has available resources, and all goes well
  - Can check in Flukes if provisioning successfully finished or not
- When finished, can log in using ssh
  - X11 display pushback may be possible
  - When used is finished, release slice
  - Will eventually expire even without release

Homework Assignments

- Exercise 1 – Obtain credentials !
- Implement 4-node path topology



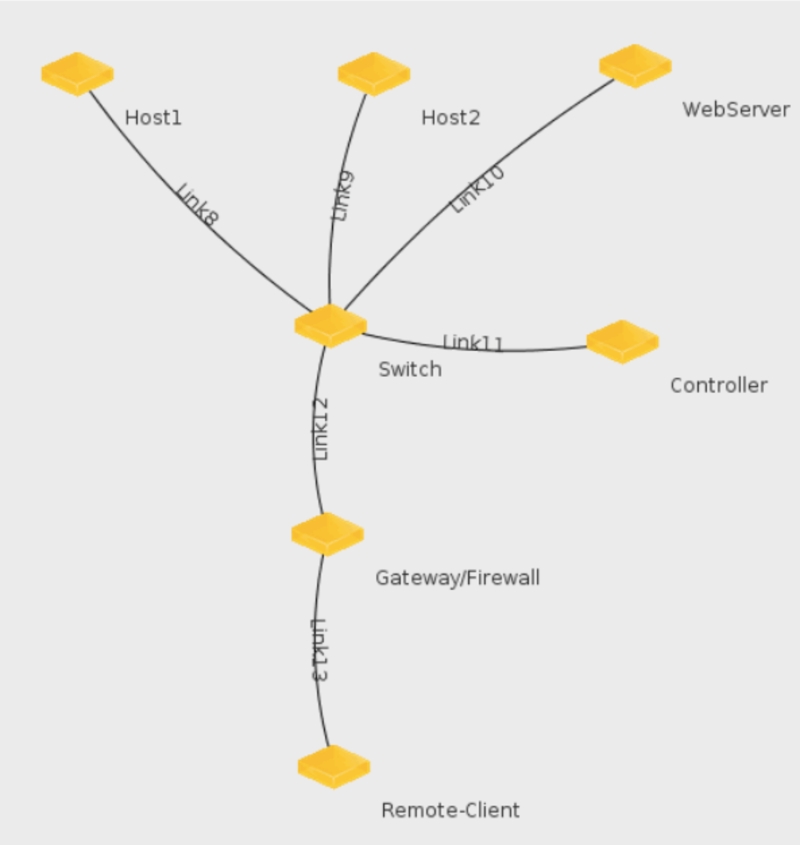
1. Examine the arp table of each
2. Examine the routing table of each
3. Ping B, C and D from A
4. Examine the arp table of each
5. Add routes at B and C to allow packets to be forwarded along the path
6. Make sure forwarding is turned “on” at each machine
7. Ping B, C and D from A
8. Examine the arp table of each
9. Clear the ARP table on all nodes, run wireshark on each node, and repeat the exercise
10. From A, run "curl www.ncsu.edu", and observe the wireshark output

Homework Assignments

- Next – OpenFlow assigment
  - Accompanied before and after by reading-and-writing questions
- Supplied long "how-to" walking student through setting up slice, and OVS
  - Query the switch for available features. Interpret and explain the output. Refer to the OpenFlow specification document.
  - Install a flow to enable ICMP echo requests/replies. Hint: This would involve flows for ARP resolution and ICMP packets.
  - Modify the flow's idle and hard timeout to 20/30 seconds. Stop sending ICMP traffic from h2 to h1 for 10 seconds. Observe the switch flow table state after 30 seconds. Explain your interpretation of flow table before changing the flow timeouts and after changing it (changing it + waiting for 30s).
  - In your answer, include the OpenFlow utility commands you used, the actual output of the command, the flows you have installed, and the flow table output.

Homework Assignments

- Kernel module firewall
  - Group assignment
  - See HW
- (Builds on earlier kernel module expertise gained through previous homeworks)



Homework Feedback/Learning

- Two categories
- Problems with material/background
  - Not everybody knew how to view/set routing table in Linux, OF learning curve, …
- Problems introduced by using GENI as platform, or exacerbated
  - Confusion with management interface
  - “Access never works”
  - “Very sluggish”
  - Flukes .properties changed during semester
  - ExoSM upgraded during semester – in the interim needed to point students to specific aggregate SMs (load-balancing manually)

Sample Projects

- See list, reports
  - http://dutta.csc.ncsu.edu/csc573_fall14/view_project.html
- Common themes
  - Demonstrate something on a large topology, e.g. multipath
    - But with real forwarding
  - Fiddle with forwarding engine, OF, kernel, but put them in real network
    - But not too real and have ComTech descending on us
  - Do-over and restart as many times as needed
    - Some pain here

Student Feedback - Sample

Pros

1. Were able to allocate resources as needed on-demand, with different specifications.

2. Allowed us to work remotely as per our convenience.

3. We could create a topology in Flukes with the start-up script created for the different nodes.

This allowed us to create new slices with the nodes already configured with some basic tools like tshark, Open vSwitch installed on the nodes required.

Cons

1. There is no way to modufy a slice once created. This necessitated creation of new slices even for slight change in requirements. (For instance the bandwidth of links could not be modified after slice creation).

2. The environment wasn't very predictable in its behavior.

Even if a reservation was successful, the nodes would not initialize to a successful state sometimes.

Also, it was difficult to reserve slices on some racks sometimes.

Takeaways

- Like it or not, GENI is still evolving
  - Instructor and TAs need to be comfortable doing every single thing students are asked to do, correct and explain
- Instructor and course missions are individual
  - GENI Education is different from GENI in Education
- Like any instrument, GENI can get in the way of learning rather than help invisibly
  - For my purpose, it has now "turned the corner"
- Overarching advantage – personalized "arbitrary" sized lab for "arbitrary" class size: no substitute
  - All networking labs like this, someday?