Extra Credit Assignment for ECE 375
Due: 05/10/2013 at midnight

Note: If you get full credits for this assignment you can improve your final grade by 10%

Preparations:
To be able to carry out this experiment you initially have to carry out the steps listed below.

Account:
1. Go to https://panther.gpolab.bbn.com/ press the “Use GENI” button and log in with your UMass OIT credentials.
2. Click “Join a Project” and select “UMASS-ECE374”.
3. I will then invite you to join the project.

VirtualBox:
To carry out the assignment from your own computer you need to install VirtualBox and a specific image:
1. Download VirtualBox from: https://www.virtualbox.org/
2. Install the following VM image: http://www.planet-lab.org/GEC16/GEC16-Tutorials-Final.ova
3. The account for the VM is “geni” and the account’s PW is “gec16user”.

Credentials:
1. Open browser in VM and goto: https://panther.gpolab.bbn.com/
2. Log in using your UMass OIT credentials.
3. When you are logged in, select “Profiles” from the top, right menu.
4. Next, click on “Generate SSH keypair” and enter your passphrase. Make sure you remember that passphrase.
5. On the next page click on “Download Private Key” and save key to ~/.ssh locally.
6. In the browser, still on the same page under “My Stuff, Configure Omni” click on the link “create an SSL certificate”.
7. Download the certificate to ~/.ssl locally.
8. Make sure the files that hold your secret keys are adequately protected. To make sure that’s the case you can execute the following commands:
   chmod 0600 .ssh/id_geni_ssh_rsa
   chmod 0600 .ssl/geni-mzink.pem (remember that your .pem file is named differently)
9. Generate a you public ssh key by executing the following command:
   ssh-keygen -y -f .ssh/id_geni_ssh_rsa > .ssh/id_geni_ssh_rsa.pub
10. Open a terminal and execute the following command:
   curl http://emmy9.casa.umass.edu/create_jks.py -o create_jks.py
11. Change it’s access rights by executing: chmod a+x create_jks.py
12. Execute command:
    python create_jks.py -c ~/.ssl/geni-mzink.pem -s ~/.ssl/mzink.jks -k
   ~/.ssl/geni-mzink.pem
   Note: ~/.ssl/geni-mzink.pem should be replace by the path to your pem file
   and ~/.ssl/geni-mzink.pem should be replaced by the name under which you
   would like to store your jks file.
13. Edit the 3 last lines in the .flukes.properties file:

   user.keystore=/home/geni/.ssl/mzink.jks
   ssh.key=/home/geni/.ssh/id_geni_ssh_rsa
   ssh.pubkey=/home/geni/.ssh/id_geni_ssh_rsa.pub

   by typing “nano .flukes.properties”.
   Note: Make sure the paths to your java keystore and the files hold cert and
   private key are correct!
14. Now execute the following command do download the topology description
    for our virtual network:

**Flukes:**
1. Now start up Flukes by double-clicking on the “flukes.jlnp” icon on the
desktop. (Be patient, it might take a moment until Flukes comes up!)
2. When Flukes is up, under the “File” menu choose the “Open Request” and
   select “gec14.rdf” and open it. This will show you the topology we will be
   using for this assignment.
3. Next to “Submit request” enter the slice name for this experiment. E.g., you
   can choose your username. Any other name you find appropriate for your
   slice will also do.
4. Hit “Submit Request”.
5. Then select the “Manifest View” tab.
6. Select “My Slices”. A window will pop up which should show the slice with
   the name you defined in step 3. Select your slice and then click “Query”.
   Note: You cannot use your slice until all 6 components have “Active” as resource
   state.
7. After all resources are active your slice is fully useable.
8. You can ssh into the nodes by righ-clicking on the nodes and selecting “Log
   into node” from the menu.
9. After node B is up ssh into it and execute:
   /etc/init.d/neuca stop

Congratulations. You know have set up everything to get started with carrying out
your experiment.
Experiment

The goal of your final experiment is to setup the routing as indicated in Figure 1. You might want to review the slides on IPv4 addressing and routing (also chapter 4.4 in the book).

![Figure 1 Topology and routing](image_url)

After you have set up the routing correctly, packets from A send to IP address 192.168.2.12 on node C should be routed via node B. Packets from A send to IP address 192.168.3.12 on node C should be directly forwarded to C. You can verify this by performing a traceroute to the respective IP addresses of node C from node A.

Since we’re not using a routing algorithm but set the route statically, we will make use of the “route” command. The following link gives you an introduction on how to use the command:


Remember that you can use “ifconfig” to determine which Ethernet interface (e.g., eth0) is bound to what IP address at each of the nodes.
To enable forwarding of packets on node B you have to execute the following command:

`echo 1 > /proc/sys/net/ipv4/ip_forward`

To demonstrate that you have set up the routing correctly you have to submit a screenshot from node A that shows traceroutes for both IP addresses of node C.

**Note:** You should document the steps you perform to set up the static routes since all configurations will be lost when your slice is deleted. A new slice will always be in its initial state with NO routing set up!