
Networking in the Wild

Getting started with GENI

Lab 6.1: CS3210 - Computer Networks Lab

Instructor: Krishna M. Sivalingam

Due Date: In-lab, 6th April, 2015

Jan – May 2015, Indian Institute of Technology Madras

Introduction

Welcome to the world of die-hard networking. Working on GENI will expose you to the real challenges in programming computer networks.

In this lab, you will set up everything necessary to access the nodes on GENI via SSH. You will then create your very first *slice* and run programs on the network spawned. Think of this assignment as a warm-up (and independent) exercise for the OSPF assignment.

Pre-requisites

- You must have an approved account on the GENI portal and must be able to log in using netaccess in the hostels - [Link](#)
- You must have access to the project that we have set up for IIT Madras students - IITM_CN_3210_Lab Project on the GENI portal. If you don't have access to the project yet, contact us immediately.

Some Important Notes

GENI requires the usage of several ports in addition to the port 8444, the port that is used to access the GENI portal on the browser. The CS department operates a subnetwork of its own in IIT Madras (independent of the Computer Centre and hence netaccess) and the CS department firewall blocks most of these ports (for security reasons). This means that the GENI portal is inaccessible from the department (or does it?). Nevertheless, you can use netaccess in the hostels to access all these ports.

The Multi-hop SSH

Today's session is in lab. So you will need to access the GENI portal via the DCF. Instead of freeing the ports that GENI requires from the department firewall, we're going to use this opportunity to learn something practical. We will be accessing GENI from the department using a workaround - i.e via an SSH tunnel to

one of the computers in the hostel zone. Also, you will access the nodes on GENI via a [ssh-within-a-ssh](#) or a multi-hop SSH tunnel.

Task 1: The SSH Tunnel

In this section, you will first set up a SOCKS proxy via a SSH tunnel. You will then use this proxy to access netaccess from the department (which is otherwise not possible). The computer in the hostel zone that you will ssh into is on ip address 10.22.20.166. All of you have been given individual accounts on that computer: username and password are both your roll number in small case (e.g., cs10b060).

- Setup a SOCKS proxy on your computer by typing the following command in your terminal.

```
ssh -D 1234 <roll_no>@10.22.20.166
```

Type in the password when prompted. You should now be ssh-ed into that machine. Minimize this terminal and leave it alone.

- Now, in the proxy settings of your favorite browser, set the http, ssl and ftp proxy fields to blank values. In the SOCKS host, enter localhost as the hostname, with the port 1234.
- You should now be able to activate netaccess by going to <http://netaccess.iitm.ac.in> and authenticating yourself. Once done, you should be able to access any website, including the GENI portal.

Checkpoint: Login into the GENI portal and open the project home page of IITM_CN_3210_Lab.

Task 2: The First Slice

In this section, you will create your first slice of the GENI network.

- In the main screen of the project, click on Create a Slice.
NOTE: DO NOT create more than one slice. You will create one slice and exactly one slice, which you will be using to spawn different topologies for the rest of the course. The slice will have the following details.
- **Name of the slice:** Your roll number in small case. **Description:** “This is the slice that I, <enter your name here>, will be doing my experimentation on”.
- Once your slice has been created, you will see your slice’s homepage. For creating the desired topology in your slice, you will use the SliceJacks Beta Graphical Tool. Click on Add resources in the interface.
- Choose RSpec, and upload the rspec file that we have put up on moodle. Once you choose the rspec, you should see the graphical interface updating itself with the topology. By default, the rspec file uses the Cornell InstaGENI physical network. Click on the box that says Cornell InstaGENI and choose the corresponding aggregate from the dropdown menu, corresponding to your roll number, as given below:

Roll Numbers (last 2 digits)	Aggregate to use
1-10	Cornell InstaGENI
11-20	Wisconsin InstaGENI
21-30	CaseWestern InstaGENI
31-40	NYU InstaGENI
41-50	Kentucky InstaGENI
51-62	Stanford InstaGENI

Once done, click on reserve resources and wait for the nodes to be spawned and loaded.

- Once your resources have been added, you will be taken to a page that lists the ssh commands that you need to run to login to the corresponding GENI machines.

Eg. `ssh abhir01@pc3.case.geni.edu -p 32456`

Do not execute them yet. There are a few more things we need to do.

Checkpoint: You have now successfully spawned the given topology in your slice. Wait till the nodes change color to green in the GUI (in the home page of your slice) - This indicates that your node is up and running.

Task 3: Accessing the GENI nodes

In this section, you will setup ssh access to the nodes on GENI.

3.1 Adding SSH Keys

GENI allows you to log onto nodes only via SSH keys (and not via username-password authentication). So we will first set up the SSH keys on your computer and on the GENI portal.

- The first step is to generate an SSH key on your computer. Run the following command on your terminal

```
ssh-keygen -t rsa -C "your_email@example.com"
```

Let the key be stored in `~/.ssh/id_rsa`. Enter a passphrase (set any password of your choice, you can also leave it blank). Once done, a message saying that your public key has been saved should appear on the terminal screen. You should see two files in the `.ssh` folder – `id_rsa` (the private key) and `id_rsa.pub` (the public key).

- Add the key to your ssh agent: Run the following two commands on your terminal

```
eval "$(ssh-agent -s)"
ssh-add ~/.ssh/id_rsa
```

- Now in your GENI Portal, go to your profile, and in the SSH Keys section, choose Upload a public key, and choose `~/.ssh/id_rsa.pub`.

Checkpoint: You have now set up your SSH public key on the GENI portal, and anyone with the corresponding private key can login to the GENI nodes in your slice.

3.2 An Additional Step

If you were using netaccess directly in the hostel zone, this step would not be needed, but since you are in the department and will be doing a multi-hop ssh tunnel to the GENI nodes, you must also do the following:

- Copy your private key (the `id_rsa` file) into the `.ssh` folder in your account in the hostelzone computer (10.22.20.166). Assuming that you are `cd`-ed into the `.ssh` folder,

```
scp id_rsa <roll_no>@10.22.20.166:~/.ssh/id_rsa
```

3.3 Logging into the nodes

- In the Resource Details page of your slice, you should see the ssh commands necessary to access the GENI nodes.

Eg. `ssh abhir01@pc3.case.geni.edu -p 32456`

- If you were using netaccess directly in the hostels, you could have entered the above command (call the above command CMD) and directly logged in to those nodes. Note that we are ssh-ing on port 32456 in the above command, a port that is blocked by the department. So we cannot perform the login from the DCF internet.
- Now instead, we will do a multi-hop ssh via the hostelzone computer using the following command (Choose any one node to ssh into) :

```
ssh -t <roll_no>@10.22.20.166 ssh abhir01@pc3.case.geni.edu -p 32456  
i.e
```

```
ssh -t <roll_no>@10.22.20.166 CMD
```

You should now be logged in to the GENI node. Instead, you could also manually ssh first into the 10.22.20.166 computer and then copy-paste the CMD command into that terminal.

Checkpoint: You have now gained access to the nodes you spawned.

Task 4: Running iperf

iperf is a tool to measure internet bandwidth

- Install iperf on two adjacent nodes in the topology. SSH into them and run

```
sudo apt-get install iperf
```

- Setup an iperf server on one of them (`iperf -s`), and a client on the other (`iperf -c node-x`, where `x` is the node number)
- Observe the measured bandwidth.
- Run the TCP Echo server and client sample code on the same two nodes.

Final Checkpoint: Show the output of both iperf and the working TCP echo server-client to your TA.

Task 5: Freeing resources

This is probably the most crucial part of using GENI. Once you are done running your programs on your topology, delete your topology by freeing the resources that you have consumed (Delete resources button). This will allow other experimenter to use those resources if they need to.

Next Steps

For the OSPF assignment, this is how you will test your code.

- We will give you an rspec with the topology (unknown to you till evaluation time) that we want you to spawn.
- You will spawn the topology on GENI, ssh into those nodes and run your ospf programs with corresponding parameters for the corresponding nodes (Details of the parameters to be used will be given along with the GENI version of the assignment description).
- You will show the OSPF routing tables changing accordingly in each of the terminals of the nodes in the topology.

Optional

Document what you did in this lab, the reasons for every single step in the process, along with problems you faced and how you solved them. It will be helpful if you run into issues later.