

# CSCI 39554: Networking Spring 2018 Lab Zero – Hello World!

*Instructor:* Prof. Saptarshi Debroy ([saptarshi.debroy@hunter.cuny.edu](mailto:saptarshi.debroy@hunter.cuny.edu))

*Lab Assistant:* Minh Nguyen ([minh.nguyen@hunter.cuny.edu](mailto:minh.nguyen@hunter.cuny.edu))

## 1. Goals

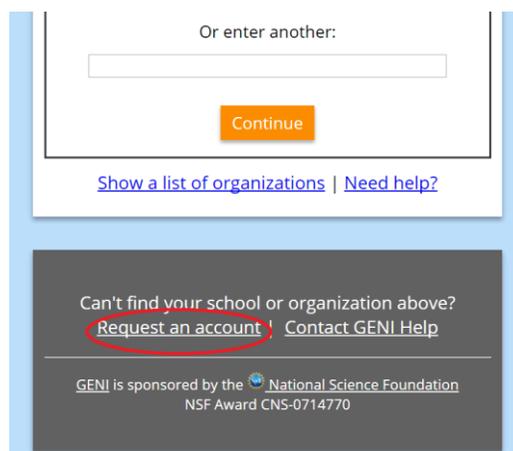
In this experiment, you will setup your GENI account, and use it to reserve an experiment a.k.a. “slice” that features two nodes (virtual machines or VMs) communicating over a Layer 3 connection. Test the experiment in your reserved slice by performing basic bandwidth measurement between the nodes.

## 2. GENI Account Setup

### 2.1. Create a GENI account

**Note:** This subsection is for those who do NOT have Lehman Once Access or Graduate Center Network accounts. If you have one, you can skip this one and see subsection 2.2.

Go to [www.geni.net](http://www.geni.net) and click on ‘Login’, then hit “Request an account” (**Figure 1**). You will be redirected to National Center for Supercomputing Applications (NCSA) Register page.



**Figure 1.**

Fill out all requested fields and hit “Create and Join” button, an email will be sent to your email address with a link and a code to validate your email. Click on the link and a new tab will pop up, enter your email and the code and click “Summit”. Then, select ‘reset password’ to set up a password for your account (Figure 2).

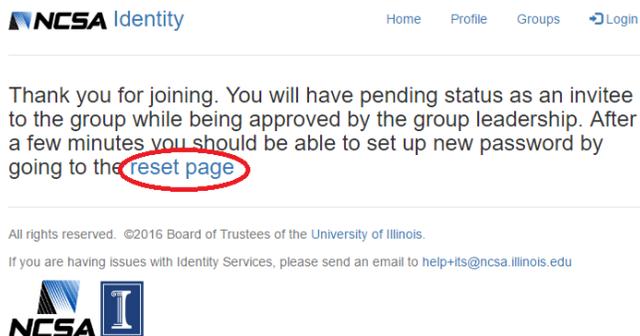


Figure 2.

Enter your username and a new email will be sent to your address containing a new code for resetting the password. Use that code to set up a new password and now you are good to go.

## 2.2. Associate your institution account with GENI

*Note: This subsection is for those who have Lehman Once Access or Graduate Center Network accounts. If you don't have one, skip this one and see subsection 2.1 above.*

## 2.3. Logging into GENI

Now, go to [www.geni.net](http://www.geni.net) and hit “Login”, then choose “Show a list of organizations” (Figure 3).

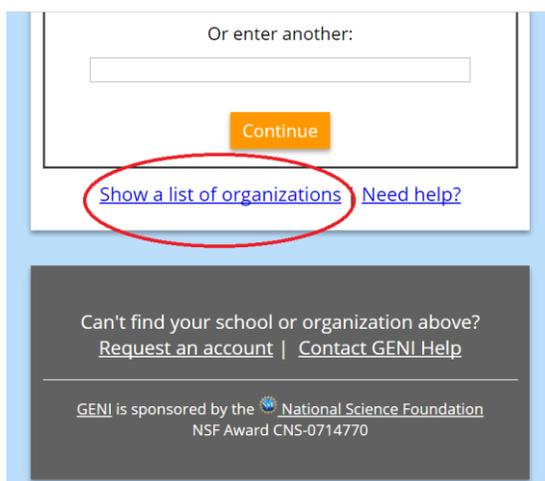
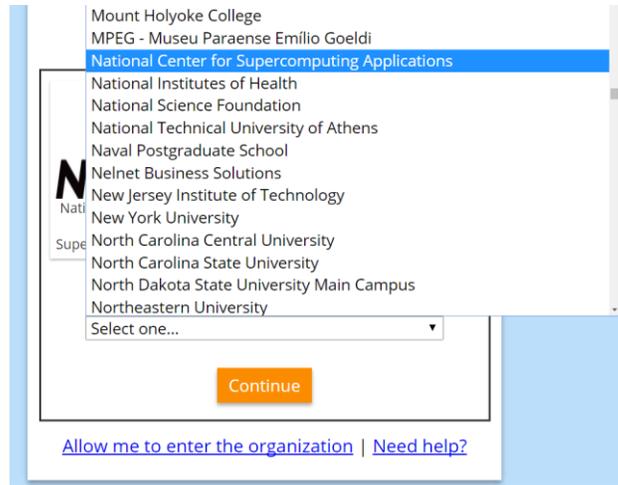


Figure 3.

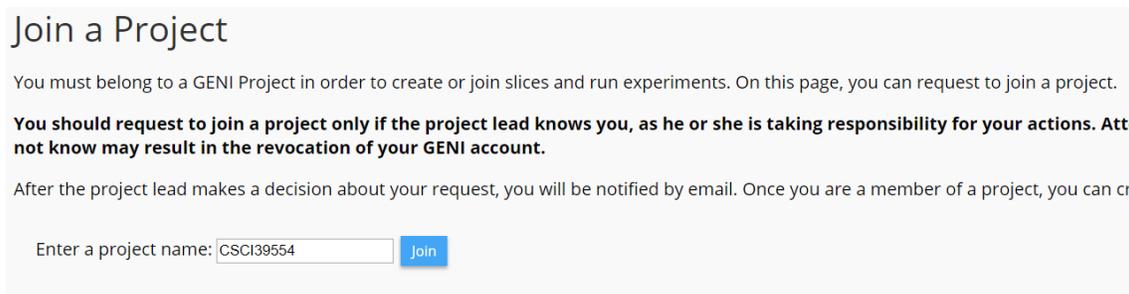
Next, select “National Center for Supercomputing Applications” and hit “Continue” (**Figure 4**). Use the username and password you’ve created to log in, accept their agreements to activate GENI account. Then click on “Use GENI”, you will be redirected to GENI site.



**Figure 4.**

#### ***2.4. Associate your GENI account with the course project***

In order to use the portal to reserve GENI infrastructure resources for your lab exercises, you must be the member of a GENI project. In your ‘Home’ page, click on “Join a Project” button. Then enter the keyword ‘CSCI39554’ and select ‘Join’ (**Figure 5**). Next, send the request and wait for the acceptance.

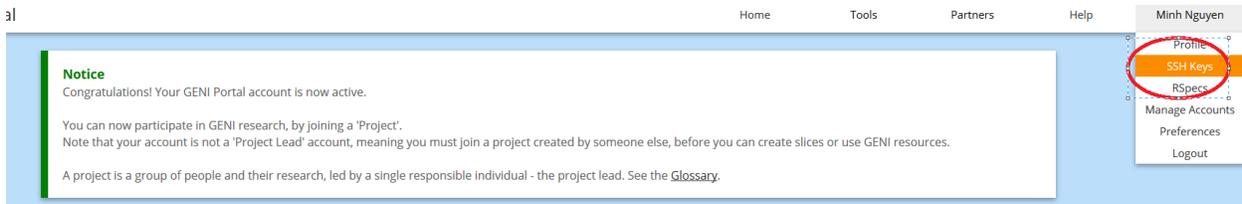


**Figure 5.**

### **3. Keypair Generation**

Access to the resources in GENI through the portal is provided through a SSH keypair (a pair of private – public key). **Public key** is stored on the remote end (i.e. VMs on GENI in this case); a local end (i.e. your machine) can only access to these VMs if it has the matching **private key**.

To generate a keypair, click on “SSH Keys” section on “Your name” menu (top right, **Figure 6**).

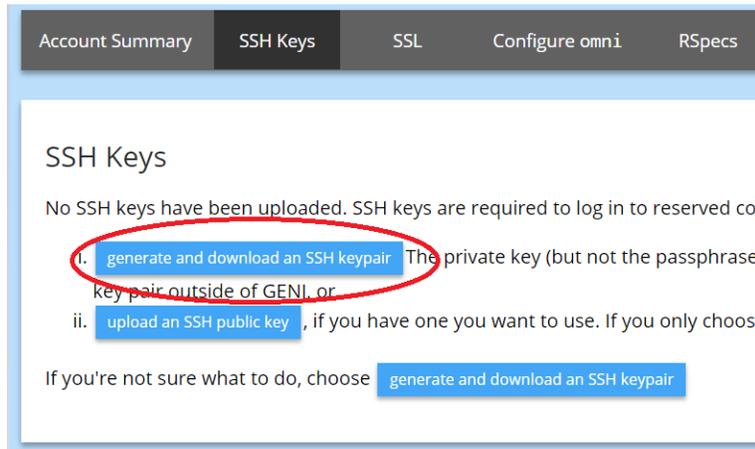


**Figure 6.**

Next, choose “generate and download an SSH keypair” (**Figure 7**), enter passphrase and hit “Generate SSH private key”.

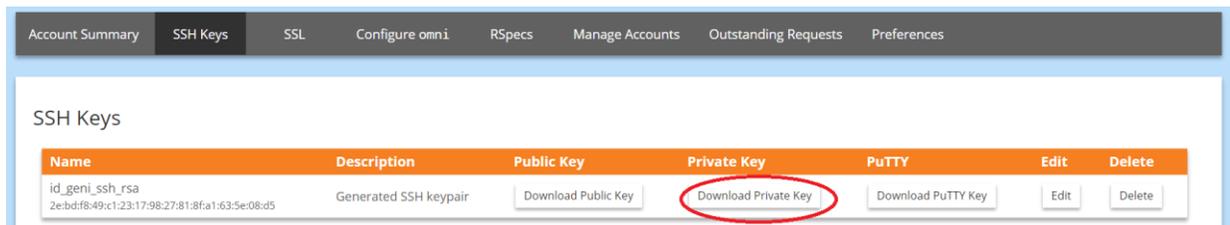
**Note:**

1. *Passphrase should be strong to avoid being hijacked (at least should be a combination of letters and numbers; capital letters and special characters are recommended).*
2. *Please remember your passphrase as it will be used for the entire course.*



**Figure 7.**

After finish generating new keypair, download your private key (**Figure 8**). The file name should be ‘*id\_genie\_ssh\_rsa*’ and it contains the passphrase you have just created.



**Figure 8.**

**Note:**

1. Please keep this private key in safe place as you are going to use it for the entire course.
2. For Linux and Mac users, you guys need to change the permission for the key:

```
chmod 0600 /<PATH-TO-YOUR-KEY-DIRECTORY>/id_geni_ssh_rsa
```

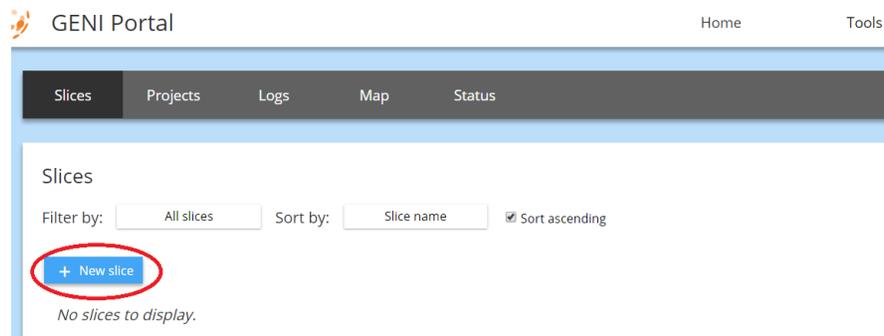
For example:

```
chmod 0600 /home/geni_keys/id_geni_ssh_rsa/
```

## 4. Experiment Setup

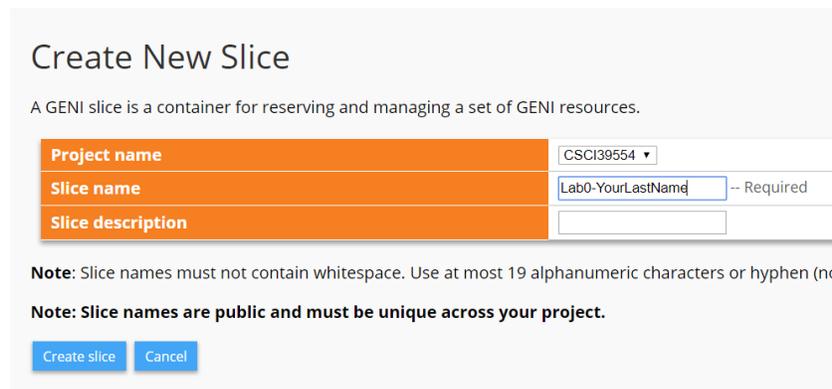
### 4.1. Create and renew a new slice

Go back to home tab, click on “New Slice” (**Figure 9**).



**Figure 9.**

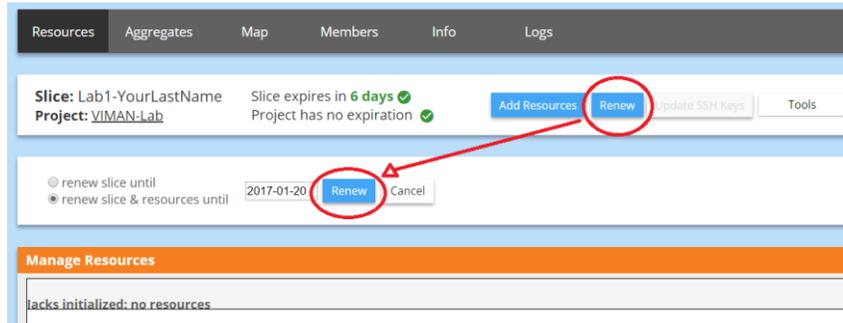
Choose project “CSCI39554”, name your slice “Lab0-YourLastName”, and hit “Create slice” (**Figure 10**).

The image shows a screenshot of the 'Create New Slice' form. The title is 'Create New Slice'. Below the title is a descriptive sentence: 'A GENI slice is a container for reserving and managing a set of GENI resources.' The form has three input fields: 'Project name' with a dropdown menu showing 'CSCI39554', 'Slice name' with a text input field containing 'Lab0-YourLastName' and a '-- Required' label, and 'Slice description' with an empty text input field. Below the form are two buttons: 'Create slice' and 'Cancel'. There are two notes below the form: 'Note: Slice names must not contain whitespace. Use at most 19 alphanumeric characters or hyphen (no' and 'Note: Slice names are public and must be unique across your project.'

**Figure 10.**

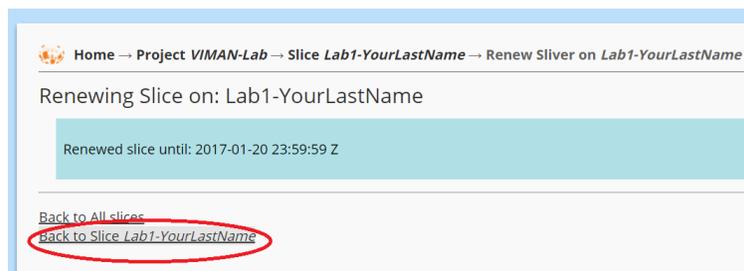
As you can see, the expiration of new slice is six days by default (meaning it will be deleted automatically after six days). Therefore, you need to renew it until after the lab report submission deadline. To do so, hit the “Renew” option and enter the date you wish to renew (**Figure 11**).

***Note:** You cannot delete slices manually, it will be automatically removed after its expiration date. You also cannot shorten the lifetime of slices after renewing them.*



**Figure 11.**

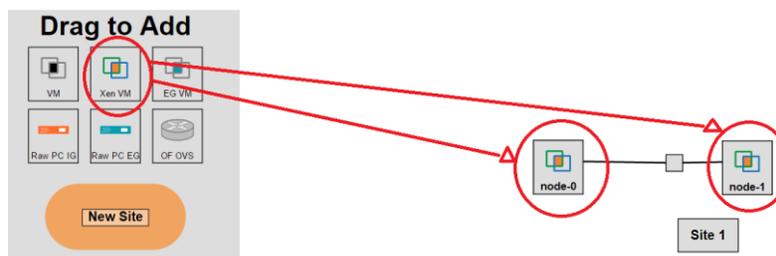
Wait until it says “Renewed slice...” and then click on “Back to Slice...” to go back (**Figure 12**).



**Figure 12.**

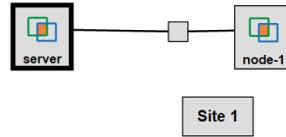
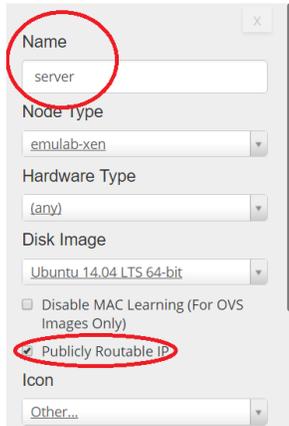
#### 4.2. Add resources

Firstly, click on “Add Resources”, you will be redirected to Jacks Graphic User Interface (Jacks GUI). Next, drag two “Xen VM” nodes from node menu and drop them on the white background on the right (**Figure 13**). Then create a link between to node by moving the mouse near one of two nodes and hold left mouse to drag the link to the other one.



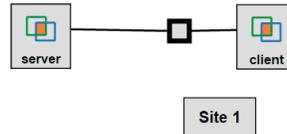
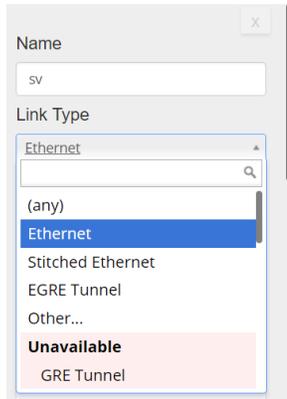
**Figure 13.**

Secondly, click on 1 node, rename it to “server” and check on “Publicly Routable IP”; for the ‘Disk Image’ option, choose Ubuntu 14.04 LTS (**Figure 14**). Similarly, click on the other one, rename it to “client”, check on “Publicly Routable IP”, and choose Ubuntu 14.04.



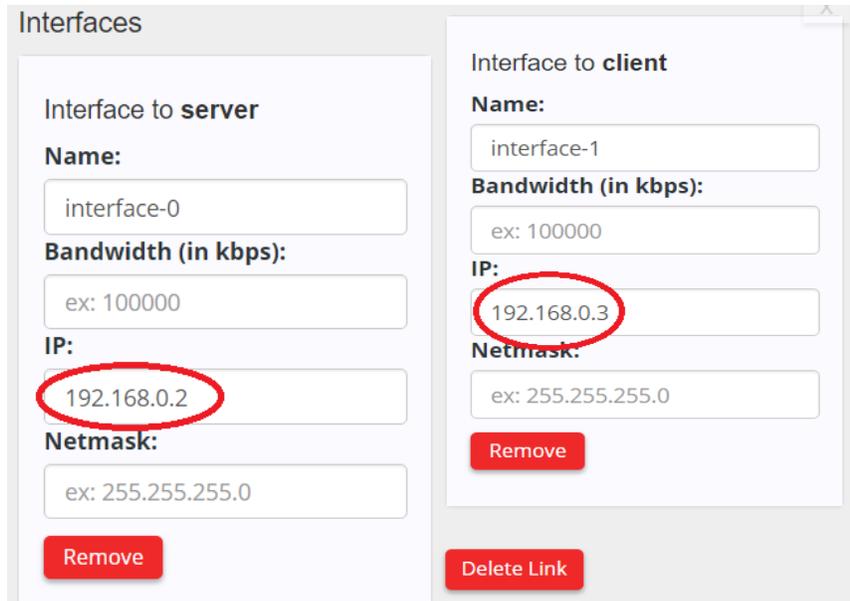
**Figure 14.**

Next, click on the small square in the middle of the link, rename it to “sv” (server - client) and change the link type to “Ethernet” (Layer 3 communication, see **Figure 15**).



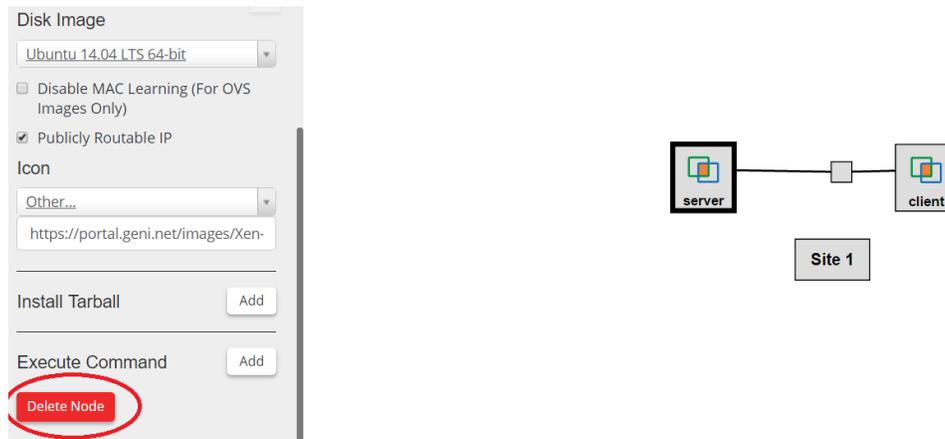
**Figure 15.**

Also on the link menu, scroll down to see the “Interfaces” section. Set the “Interface to **server**” to IP “192.168.0.2” and “Interface to **client**” to IP “192.168.0.3” (**Figure 16**).



**Figure 16.**

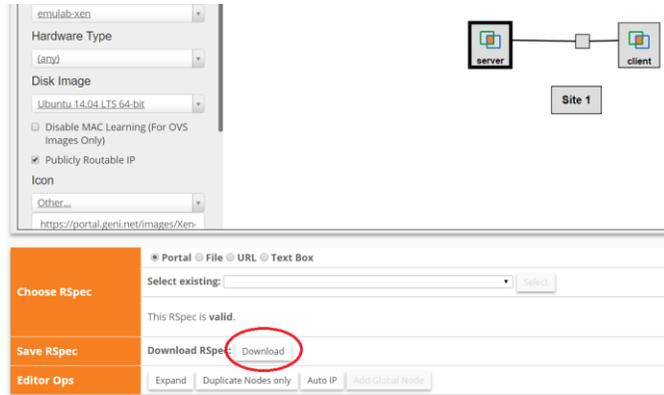
***Note:** If you want to delete a node or link, just click on that node or link and scroll all the way down until you see the red “Delete” button (Figure 17).*



**Figure 17.**

### 4.3. Save source code (RSpec)

All of the resources you have been adding so far is stored under a source code form called RSpec. You might want to save this source code for future experiment, so you don't have to create the whole experiment again later. To do so, scroll the page down until you see “Download RSpec” option (**Figure 18**).



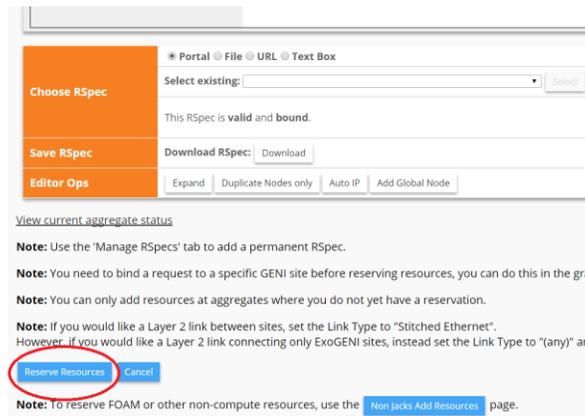
**Figure 18.**

**Note:**

1. *The downloaded file comes with '.xml' extension by default. However, '.xml' is usually considered unsafe by email servers, this means it will be blocked when you send it via email. Therefore, you should change its extension to '.rspec'.*
2. *To load the topology from RSpec.*

**4.4. Choose site (aggregate) and reserve resources**

Click on “Site 1” box and choose one of these sites: MAX, NPS, NYU, NYSERNet, Rutgers, or Missouri. Then scroll down until you see “Reserve Resources” button and click it (**Figure 19**).



**Figure 19.**

You can see it's running (**Figure 20**).

Add Resources to GENI Slice *Lab1-YourLastName* (Results)

Total run time: **6 seconds**  
 Status: **Running**

[Results](#) | [Detailed Progress](#) | [Request RSpec](#) | [Manifest RSpec](#) | [Send Problem Report](#) | [Advanced](#)

Results

Resources requested from RSpec:

Pending... (See 'Detailed Progress' tab for more information.)

[Back to All slices](#)  
[Back to Slice \*Lab1-YourLastName\*](#)

**Figure 20.**

Wait for it to be finished (up to 2 minutes) to make sure there is no error (**Figure 21**). Theoretically you can choose any InstaGENI site, but some site might not work properly at some point (due to the lack of resources). So if you run into problems like “Not enough...” or “Error creating container experiment”, just give it another shot with other sites.

Add Resources to GENI Slice *LabU-YourLastName* (Results)

Total run time: **24 seconds**  
 Status: **Finished**

[Results](#) | [Detailed Progress](#) | [Request RSpec](#) | [Manifest RSpec](#) | [Send Problem Report](#) | [Advanced](#)

Results

Resources requested from RSpec:

Note that the results are current as of the finish time. Your resource allocation may have changed after this time if resources expired or were the most up-to-date results about your slice's current allocated resources.

**Node #1 (at Ohio Metro Data Center InstaGENI):**

Status	Client ID	Component ID	Expiration	Type	Hostname
Unknown	router-A	pc1		emulab-xen	router-A.LabU-YourLastName.ch-geni-net.instageni.metrodatacenter.com
<b>Login</b>					
ssh_minbguy@pc1.instageni.metrodatacenter.com_p_25445					
<b>Interfaces</b>			<b>MAC</b>		
<b>Layer 3</b>			<b>Layer 3</b>		
interface-5	pc1:eth2	02c591c7cf61	ipv4:	0.0.0.0	
interface-0	pc1:eth2	02a96ac0f2db	ipv4:	10.10.100.1	

**Node #2 (at Ohio Metro Data Center InstaGENI):**

**Figure 21.**

Now, scroll all the way down to the bottom of the page and hit “Back to Slice...” option to go back your slice. You have to wait until both two nodes are up (the color change from grey to green, see **Figure 22**). This step will take several minutes (up to 20).

**Manage Resources**

Resources on Illinois InstaGENI are ready. [View Report](#)

client — server

[Renew](#) | [Renew Date](#) | [Delete](#) | [SSH](#) | [Restart](#) | [Snapshot](#) | [Details](#) | [Add Resources](#) | [Expand](#)

**Figure 22: Both two nodes are green.**

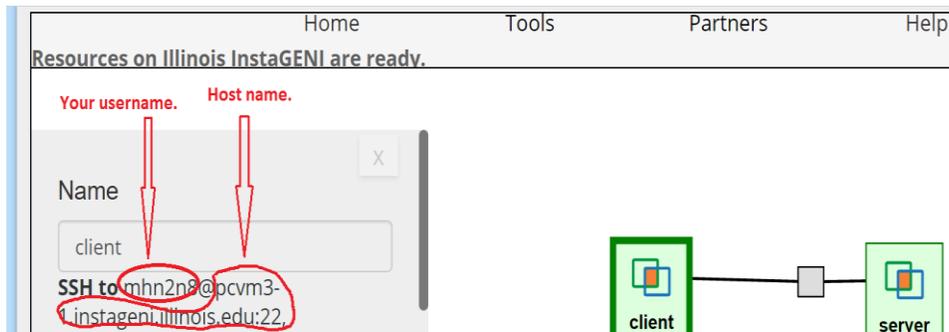
**Note:**

1. Sometimes, some nodes might not be able to boot up (they remain gray and don't turn green). If you run into this issue, you have to delete the resources (see [section 7](#) on how to delete resources) and re-add them; consider choosing other site.
2. As being said, some sites might not work properly at some point. Therefore, the best option is saving RSpec file before choosing a site.

## 5. Secure Shell (SSH)

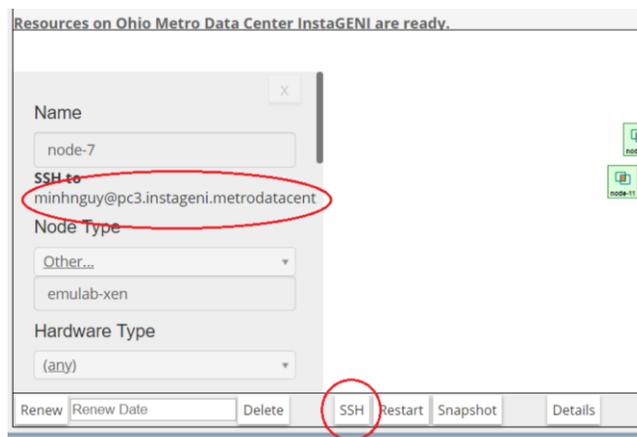
SSH, also known as Secure Socket Shell, is a network protocol that provides you with a secure way to access a remote computer.

Now, if you click on an active node, you can see the *username* and *hostname* to SSH to that node (under the form *username@hostname*, see [Figure 23](#), excluding the port number ‘:22’).



**Figure 23: The hostname does not include ‘:22’ part, that is SSH default port number.**

Sometimes, you might run into an issue like [Figure 24](#) (you can only see a part of SSH info).



**Figure 24.**

All you need to do is click on SSH option right below the topology (**Figure 24**), a new tab will pop up with full SSH info (**Figure 25**).



**Figure 25.**

***Note:** If you reserve the resources first (subsection 4.2 to 4.4) and generate the keypair later (section 3), you will not be able to use it for SSH. Therefore, if you re-generate the keypair, you have to update the SSH keypair (Figure). Worst case scenario, you might have to delete current resources (see section 7 on how to delete resources) and re-add new ones.*

### 5.1. For Linux and Mac

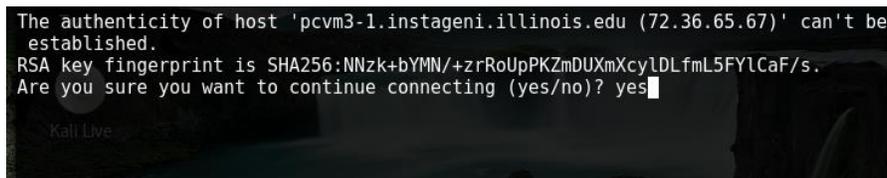
Open a terminal and type in the following command. This is to SSH to “username@hostname” node using the private key that you downloaded previously.

```
ssh -i /<PATH-TO-YOUR-KEY-DIRECTORY>/id_geni_ssh_rsa username@hostname
```

For example:

```
ssh -i /home/geni_keys/id_geni_ssh_rsa/ mhn2n8@pcvm3-1.instageni.illinois.edu
```

For the first time only, you will be asked confirming the connection, type in “yes” and press ‘Enter’ (**Figure 26**), then enter the passphrase when prompted (**Figure 27**).



**Figure 26.**

```

The authenticity of host 'pcvm3-1.instageni.illinois.edu (72.36.65.67)' can't be
established.
RSA key fingerprint is SHA256:NNzk+bYMN/+zrRoUpPKZmDUXmXcyLDLfmL5FYlCaF/s.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'pcvm3-1.instageni.illinois.edu,72.36.65.67' (RSA) to
the list of known hosts.
Enter passphrase for key '/root/Downloads/id_geni_ssh_rsa':
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-33-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

mhn2n8@client:~$

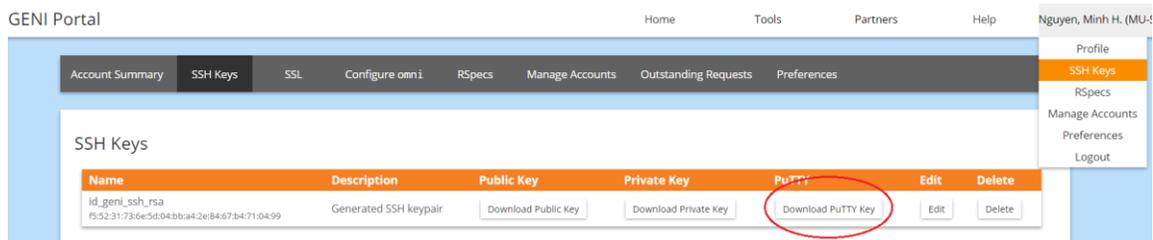
```

**Figure 27: Connection was established successfully.**

### 5.2. For PC (using PuTTY)

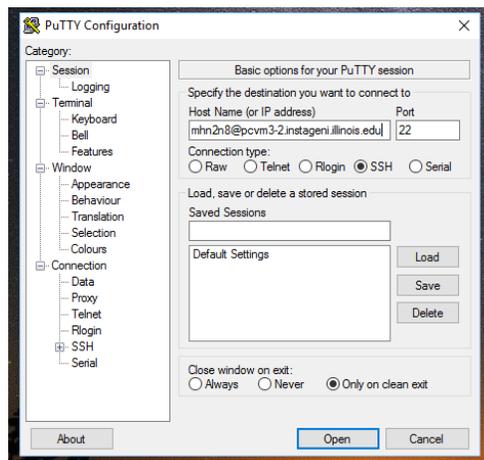
PuTTY is a popular open-source network file transfer application, you can download PuTTY [here](#). PuTTY uses a special key to establish the connection. To download it, go to “SSH Keys” section on “Your name” menu (top right, see **Figure 28**) and select “Download PuTTY key”. Then, enter the passphrase and download it. The file name should be ‘*id\_geni\_ssh\_rsa.ppk*’.

**Note:** Please keep PuTTY key in safe place since you are going to use it for the entire course.



**Figure 28.**

Open PuTTY and enter ‘*username@hostname*’ of the node you want to access to (**Figure 29**).



**Figure 29.**

Then, double click on “SSH” option below (Figure 30) and select “Auth” sub-option.

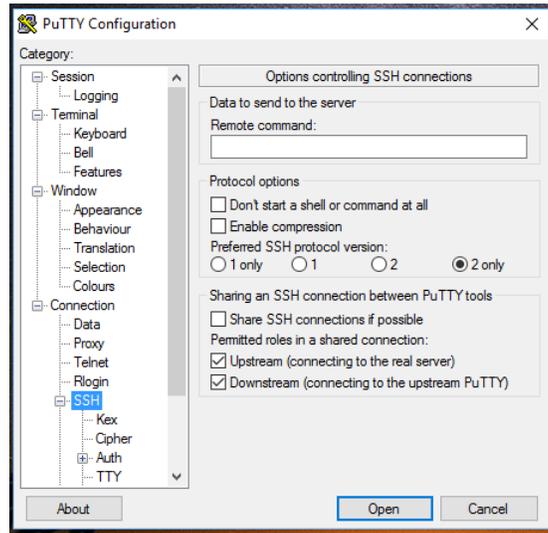


Figure 30.

In “Auth” section, select “Browse...” and point your PuTTY key, then hit “Open” (Figure 31). Click “Yes” when prompted (first time only) and enter the passphrase (Figure 32).

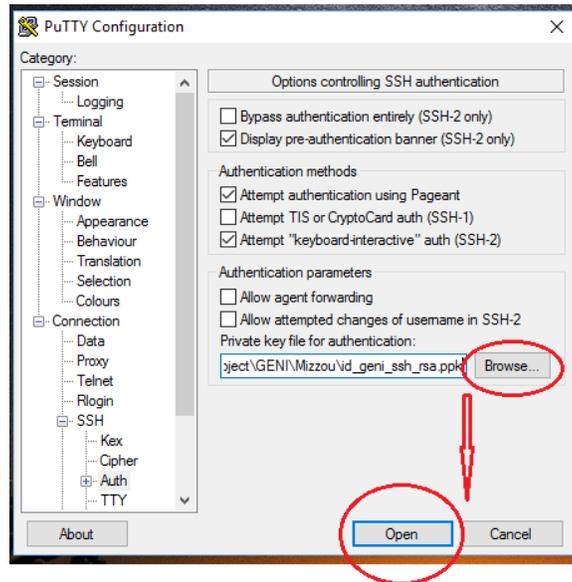


Figure 31.

```
mhn2n8@server: ~
Using username "mhn2n8".
Authenticating with public key "imported-openssh-key"
Passphrase for key "imported-openssh-key":
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-33-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

mhn2n8@server:~$
mhn2n8@server:~$
mhn2n8@server:~$
```

**Figure 32: Connection established successfully.**

**Note:**

1. Unlike PuTTY, most SSH applications on PC nowadays use private key to establish connection. However, PuTTY still remains popular due to its built-in useful utilities.
2. You can convert a regular private key into a PuTTY key. To do that, you need an application so-called PuTTYgen. For more information, please refer to [this](#).
3. Thus, although using PuTTY for SSH, you should still keep the private key.

## 6. Bandwidth Test Experiment

**Note:** Due to some font conflict, please do NOT copy and paste the contents from this instruction, just type them in.

SSH to ‘server’ node and type in the following command to update the system (**Figure 33**).

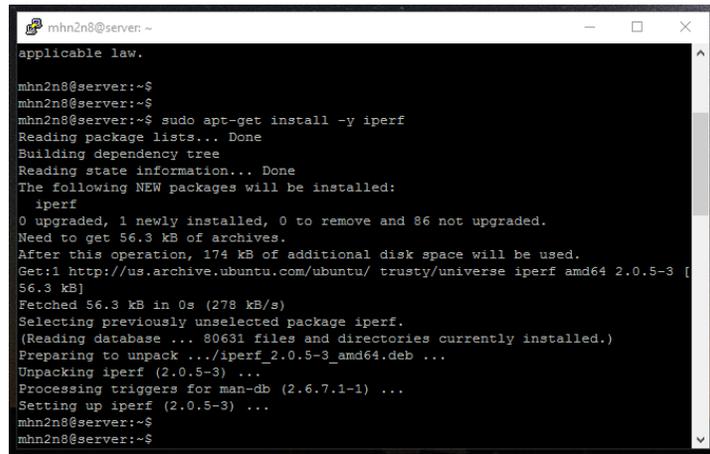
```
sudo apt-get update
```

```
mhn2n8@server: ~
mhn2n8@server:~$
mhn2n8@server:~$ sudo apt-get update
Ign http://us.archive.ubuntu.com trusty InRelease
Get:1 http://us.archive.ubuntu.com trusty-updates InRelease [65.9 kB]
Hit http://us.archive.ubuntu.com trusty Release.gpg
Hit http://us.archive.ubuntu.com trusty Release
Get:2 http://us.archive.ubuntu.com trusty-updates/main Sources [388 kB]
Get:3 http://us.archive.ubuntu.com trusty-updates/restricted Sources [5,888 B]
Get:4 http://us.archive.ubuntu.com trusty-updates/universe Sources [171 kB]
Get:5 http://us.archive.ubuntu.com trusty-updates/main amd64 Packages [938 kB]
Get:6 http://us.archive.ubuntu.com trusty-updates/restricted amd64 Packages [16.4 kB]
Get:7 http://us.archive.ubuntu.com trusty-updates/universe amd64 Packages [392 kB]
Get:8 http://us.archive.ubuntu.com trusty-updates/main 1386 Packages [896 kB]
Get:9 http://us.archive.ubuntu.com trusty-updates/restricted 1386 Packages [16.2 kB]
Get:10 http://us.archive.ubuntu.com trusty-updates/universe 1386 Packages [394 kB]
Get:11 http://us.archive.ubuntu.com trusty-updates/main Translation-en [460 kB]
Get:12 http://us.archive.ubuntu.com trusty-updates/restricted Translation-en [3,842 B]
Get:13 http://us.archive.ubuntu.com trusty-updates/universe Translation-en [208
```

**Figure 33.**

Then type in this command to install bandwidth test tool namely ‘iperf’ (**Figure 34**).

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



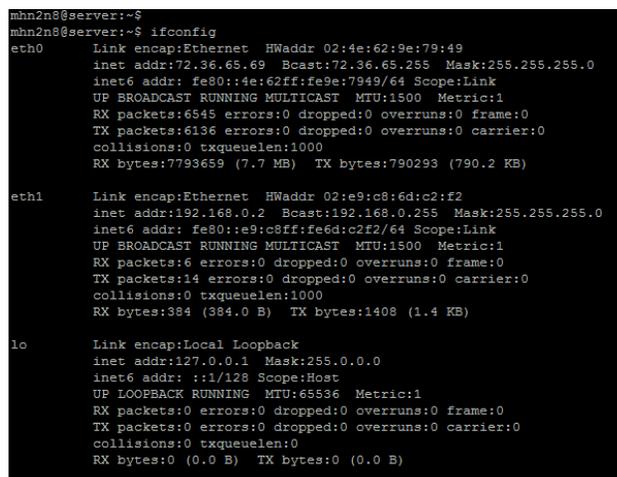
```
mhn2n8@server:~$ sudo apt-get install -y iperf
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  iperf
0 upgraded, 1 newly installed, 0 to remove and 86 not upgraded.
Need to get 56.3 kB of archives.
After this operation, 174 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu/ trusty/universe iperf amd64 2.0.5-3 [
56.3 kB]
Fetched 56.3 kB in 0s (278 kB/s)
Selecting previously unselected package iperf.
(Reading database ... 80631 files and directories currently installed.)
Preparing to unpack .../iperf_2.0.5-3_amd64.deb ...
Unpacking iperf (2.0.5-3) ...
Processing triggers for man-db (2.6.7.1-1) ...
Setting up iperf (2.0.5-3) ...
mhn2n8@server:~$
```

**Figure 34.**

Similarly, do the same thing on ‘client’ node.

**Note:** If you type in “ifconfig”, you can see two Ethernet interfaces (**Figure 35**).

- ‘eth0’ has the public IP for SSH, meaning the ‘hostname’ is mapped to this IP via Domain Name Service (DNS). Hence, in this case, you can type ‘username@72.36.65.69’ instead of ‘username@hostname’.
- ‘eth1’ contains the local IP ‘192.168.0.2’ that we’ve set up earlier in this lab.



```
mhn2n8@server:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 02:4e:62:9e:79:49
          inet addr:72.36.65.69  Bcast:72.36.65.255  Mask:255.255.255.0
          inet6 addr: fe80::4e:62ff:fe9e:7949/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6545 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6136 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7793659 (7.7 MB)  TX bytes:790293 (790.2 KB)

eth1      Link encap:Ethernet  HWaddr 02:e9:c8:6d:c2:f2
          inet addr:192.168.0.2  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::e9:c8ff:fe6d:c2f2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:384 (384.0 B)  TX bytes:1408 (1.4 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

**Figure 35.**

Now, on ‘server’ node, type:

```
iperf -s
```

On ‘client’ node, type:

```
iperf -c 192.168.0.2
```

The result can be seen in **Figure 36 and 37**. The bandwidth via a local link between ‘server’ and ‘client’ is about 94 Mb/s or 11.5 MB/s.

```
mhn2n8@server:~$
mhn2n8@server:~$
mhn2n8@server:~$ iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 4] local 192.168.0.2 port 5001 connected with 192.168.0.3 port 39763
[ ID] Interval      Transfer    Bandwidth
[ 4]  0.0-10.1 sec  113 MBytes  93.8 Mbits/sec
```

**Figure 36: Outcome on ‘server’ node.**

```
mhn2n8@client:~$ iperf -c 192.168.0.2
-----
Client connecting to 192.168.0.2, TCP port 5001
TCP window size: 85.0 KByte (default)
-----
[ 3] local 192.168.0.3 port 39763 connected with 192.168.0.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3]  0.0-10.0 sec  113 MBytes  95.1 Mbits/sec
mhn2n8@client:~$
```

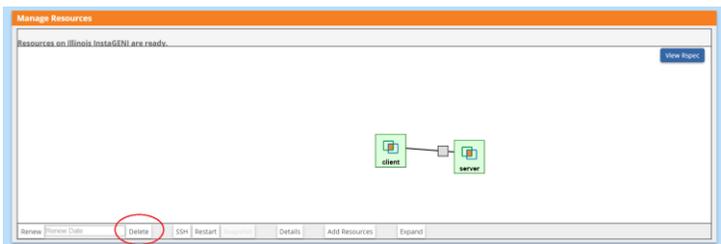
**Figure 37: Outcome on ‘client’ node.**

**Note:**

- 1. *Your output might be different from mine.*
- 2. *‘iperf -s’ on ‘server’ node will keep running. To quit it, press ‘Ctrl + C’.*

**7. Delete Resources**

When you are done with the lab, please delete your resources in the GENI Portal, to free them for other experimenters, sharing is caring. To do so, simply click on “Delete” button (**Figure 38**).



**Figure 38.**