Iowa State University
Department of Electrical and Computer Engineering
Cpr E 489: Computer Networking and Data Communications
Lab Experiment #6
GENI Experiment: IPv4 Routing Basics
(Total Points: 100)

Objective

In this GENI lab, you will get familiar with the static routing protocol and manually updating the routing tables.

Pre-Lab

Complete the steps a-f of the Procedure section and remind yourself how to SSH into the nodes as in the "Introduction to GENI" lab.

1) Go to https://portal.geni.net/ press the **Use GENI** button and from the **Drop-Down** menu select your institution.
2) Join a project through GENI portal and create a slice (use **LastName-lab6** as the name)
3) If needed, generate and use a new private SSH key. (The one from the "Introduction to GENI" lab will still work)

Lab Expectations

Work through the lab and let the TA know if you have any questions. After the lab, write up a lab report with your partner. Be sure to

- **summarize what you learned in a few paragraphs. (20 points)**
- **include your answers for all questions, with screenshots**. **(75 points)**
- **cleanup. (5 points)**

Procedure:

a) Create a slice and name it: **LastName-lab6**.

b) On the slice page, click on your slice and then click on the **Add Resources** button placed at the top part of the screen.

c) In the **Choose RSpec** section, choose **File** and use the enclosed **lab06_rspec.xml** file.

d) You will need to choose an aggregate where you want this topology to be instantiated. Click on the Site 1 box and a panel on the left side of the canvas will appear. Choose any aggregate with **InstaGENI** in its name (except Starlight).

e) Click on the **Reserve Resources** button on the bottom left part of the screen.

f) Wait while your resources are being reserved. This will take several minutes so be patient. The node statuses on the Home->Slice->Details screen will turn green (READY) to signify that your resources are ready.

g) **Take a screenshot** of your slice when all the nodes are ready and include it in your report. **(5 points)**

h) Setup the routing:

1) The downloaded network is connected as indicated in Figure 1.

2) Open a terminal.

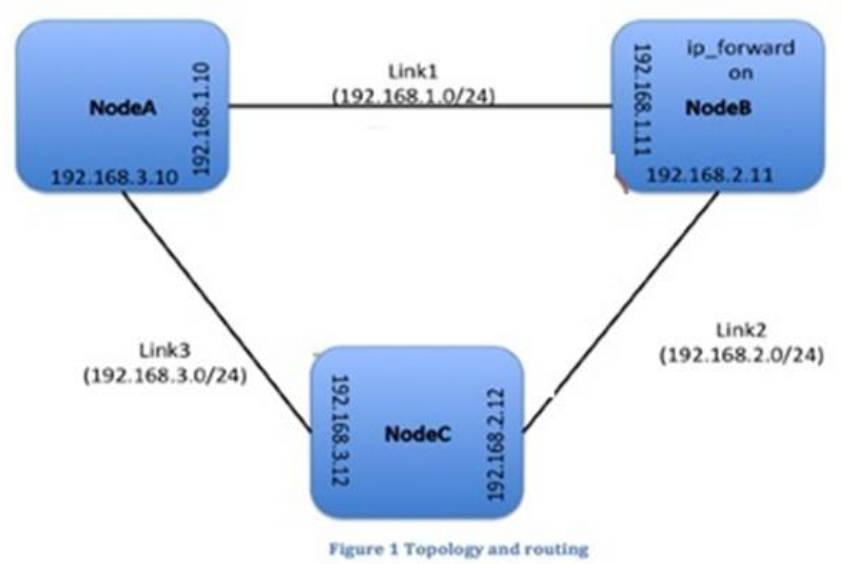


Figure 1 Topology and routing

3) Run *ssh* for all three Nodes and enter your passphrase when prompted.

```
ssh -i <private key location> <username>@<hostname> -p <port>
```

i. To get the information you need, you can load your slice in the portal, and from the Home tab locate your slice and click on it to load it.

ii. Under the topology canvas, click the Details button. This page should have all the information you need.

4) Execute the **route** command in each of the three nodes to show their respective routing tables. **Include a screenshot of the tables in your lab report. (10 points)**

5) From node A, try to ping the other addresses of nodes B and C (two IP addresses for each node). **Include a screenshot of the ping outputs and explain the results. (10 points)**

6) **What happens when you traceroute from A to IP address 192.168.2.12 before you set up the static routes? Why? Include a screenshot of the traceroute output in your lab report. (10 points)**

7) If you get a message like "-sh: 7: traceroute: not found" when trying to execute traceroute, use 'sudo apt-get install traceroute' to install traceroute.

Setting Up Static Routes

- The following command will add the destination subnet **destiny** (A.B.C.0) to the local routing table and use IP address **thisway** (W.X.Y.Z) accessible via interface **intf** as the gateway:
  `sudo route add -net` **`destiny`** `netmask 255.255.255.0 gw` **`thisway intf`**
  More specifically, in the above command:
  - **destiny** is the subnet address (A.B.C.0) that will indicate where to send packets that have a matching destination IP address (e.g., if you want to direct traffic for the subnet 192.168.254.0/24, then you would set the destination to 192.168.254.0 with netmask 255.255.255.0 accounting for the /24. Note: netmask 255.255.255.0 should be sufficient for the purposes of this lab);
  - **intf** is the name of the interface on this computer from which traffic will be sent out; and

- **thisway** is the IP address of the interface on the gateway that will receive the traffic that is sent (W.X.Y.Z).

Note: The gateway IP address **thisway** must be an address that the interface **intf** can access directly (i.e., it is not further routed before getting to **destiny**). It is also not the IP address of the interface on the current computer.

- To delete this entry in the table, simply replace **add** with **del**.
  `sudo route del -net destiny netmask 255.255.255.0 gw thisway intf`

- Now, modify the routing tables to allow for node A to reach the IP addresses that you could not reach in step 6. **Include a screenshot of the routing table of node A (10 points).**

- **Take a screenshot of node A successfully pinging and using traceroute on 192.168.2.12 (10 points).**

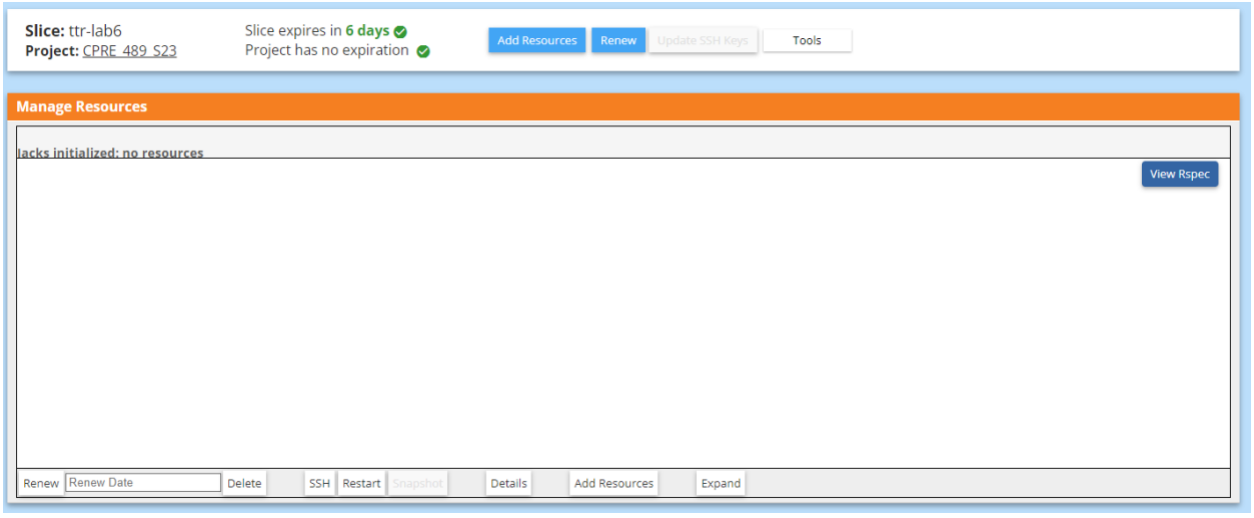
- Setup up more static route(s) so that every node can access every interface in the system. **Take screenshots of routing tables of nodes B and C as well (10 points).**

- Show a traceroute from node B using traceroute on the four interfaces it does not own. **Take a screenshot of each traceroute output. (10 points).** For example:

```
user@node-b$ traceroute 192.168.1.10
[OUTPUT]
user@node-b$ traceroute 192.168.3.10
[OUTPUT]
user@node-b$ traceroute 192.168.3.12
[OUTPUT]
user@node-b$ traceroute 192.168.2.12
[OUTPUT]
```

Cleanup

After you are done with the experiment, you should always release your resources so that other experimenters can use the resources. To cleanup your slice, press the **Delete** button at the bottom of the Manage Resources panel on the Slice page. Wait a few moments for all the resources to be released and you will have an empty canvas again. Notice that your slice is still there. There is no way to delete a slice. It will be removed automatically after its expiration date but remember that a slice is just an empty container, which doesn't take up any resources. **Include a screenshot of your slice with no resources on the Manage Resources panel (see example below). (5 points)**



Tips

- 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 IP forwarding of packets on a node, you may need to execute the following command:
  ```
  sudo sh -c 'echo 1 > /proc/sys/net/ipv4/ip_forward'
  ```
- The routing table matches the destination address to the newest, valid entry.
- Ping and traceroute use ICMP which gets sent to the receiver, processed, and sent back. The response also needs to know how to reach its destination.
- A useful tool to debug the packet flow is **tcpdump**.