

GENI Lab 1

Due: 3:30pm Sept 04, 2019

0. Lab Configurations

- This lab assumes that you've accomplished GENI Lab 0. We are going to use the resources that we reserved in that lab.
- Open one SSH terminal to the client node and one to the server node.

1. Basic Experiments

1.1. Send IP Traffic

The first experiment that we will run is to verify the IP connectivity between our hosts.

a. Check the interfaces of your nodes. In each terminal type:

```
sudo ifconfig
```

You should see at least two interfaces:

- The **control interface**. This is the interface you use to access the node, e.g. ssh into your host. The control interface is mainly used for control traffic, i.e. traffic for controlling the node and the experiment, e.g. the `sudo ifconfig` above. The IP address is likely to be something like 172.17.x.x.
- The **data interface**. This is the interface that is used for sending experimental traffic. This is the interface that connects to the other hosts of your experiment through GENI. The links between these interfaces are the ones that allow you to run network experiments. The data interface is the one that has an IP address and mask that match what you configured before you reserved your resources (Remember the "Auto IP" option that we chose?). The IP address is likely to be something like 10.10.1.x.
- There might be another interface for self loopback.

b. Fill in the [lab1-worksheet](#), noting the interface name (i/f) and IP address of the control and of the data interfaces for each node.

c. From the client, ping the server data plane interface.

From the terminal window that is logged into the client type:

```
ping <server data IP addr> -c 5
```

d. From the server, ping the client data plane interface.

From the terminal window that is logged into the server type:

```
ping <client data IP addr> -c 5
```

1.2. Install and use `iperf`

a. Install the `iperf` software on both nodes:

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

b. On both nodes run the `hash` command to ensure the new `iperf` command is accessible from the command line:

```
hash
```

c. Start an `iperf` server on the server node:

```
iperf -s
```

d. On the client node, run an `iperf` client via the data plane:

```
iperf -c <server data IP addr>
```

What is the bandwidth of this link? Fill in the [lab1-worksheet](#).

e. Still on the client node, run an `iperf` client via the control plane:

```
iperf -c <server control IP addr>
```

What is the bandwidth of this link? Fill in the [lab1-worksheet](#).

f. Type CTRL-C on the server node to stop the `iperf` server.

1.3. Bring down the server's data interface

a. From the client node, start pinging the server **data plane** interface:

```
ping <server data IP addr>
```

b. From the server node, bring down the data plane interface:

```
sudo ifconfig <server data interface name> down
```

Note: It is the server data *interface name* rather than the *IP address* that you want to use above, e.g. **eth1**.

After you bring down the data interface, the pings should indicate that the destination is unreachable or 100% packet loss.

c. From the server node, bring the data plane interface back on:

```
sudo ifconfig <server data interface name> up
```

After you bring up the data interface, notice that the ping command on the client node resumes.

Fill in the **lab1-worksheet**.

2. What to Turn in?

Submit the following file:

- lab1-worksheet.docx