Named Data Networking

- Part I: Design/Setup
  - Step 1: Design Experiment
  - Step 2: Establish Environment
  - Step 3: Obtain Resources
- Part II: Execute
  - Step 4: Wait for resources to be ready
  - Step 5: Try out NDN application
- Part III: Finish
  - Step 6: Cleaning Up

1. Design the Experiment

In today's experiment you will run an experiment to learn some concepts underlying Named Data Networking (NDN). The NDN Forwarding Daemon (NFD) will be installed on the nodes in your network and you will run an application that fetches content by name.

2. Establish the Environment

2.1 Set up ssh keys

Follow the instructions to login to the GENI portal and set up your `ssh` keys provided in your pre-work. As part of this, you will have joined the project named: `GRW-Summer-Camp-UKentucky`

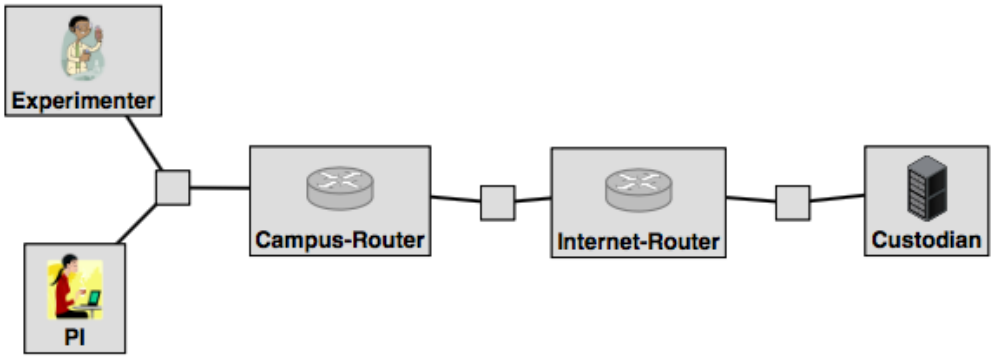
3. Obtain Resources

3.1 Create a slice

Once you have joined the project, create a slice using the `Create Slice` button, giving the slice a name of your choice. From now on that slice name will be referred to as `SLICENAME`.

3.2. Load a topology in Jacks

a. In the Portal, launch Jacks for your slice by clicking on `Add Resources`.
b. In this exercise, we will use an existing resource specification (RSpec) file that defines the topology depicted below. It has five virtual machines (VMs) in the same rack connected by layer 2 links. This scenario emulates the real NDN testbed



c. Under the `Choose RSpec` field, click on URL and then copy/paste the following URL into the field: https://raw.githubusercontent.com/GENI-NSF/geni-tutorials/master/LabOne/ndn/lab1-ndn-rspec.xml
then click on `Select`.
d. Now, click on `Site 1`, and select any `instaGENI` rack from the dropdown menu on the left. After you pick a site, scroll down and click on the `Reserve Resources` button. If this step finishes successfully (it may take up to a minute), then return to your slice page. Once all of the five VMs change colour, from grey to green, to you will be ready to move on to the next step. Note that this step can take some time, so please be patient.

4 Wait for resources to be ready

Now, click on `Site 1`, and select any `instaGENI` rack from the dropdown menu on the left. After you pick a site, scroll down and click on the `Reserve Resources` button. If this step finishes successfully (it may take up to a minute), then return to your slice page. Once all of the five VMs change colour, from grey to green, to you will be ready to move on to the next step. Note that this step can take some time (up to several minutes), so please be patient.

5 Trying out the NDN application

In this experiment, you will be able to see in-network caching in action. Our experiment consists of the

following nodes:

- A data source node, called Custodian that holds data in the namespace /nytimes
- A node, called Internet Router that forwards Interest and Data packets to and from the Custodian.
- A node, called Campus-Router that forwards Interest and Data packets to and from the university nodes.
- A principal investigator node, called PI and a experimenter node, called Experimenter that will send Interest requests to the Custodian via UDP tunnels.

Once the topology is up, logon to the Custodian node and run the script in /local/install_script.sh. This will restart the NFD daemon on this node.

```
$ cd /local      $ sudo ./install_script.sh
```

5.1 Run the NDN application on the entire topology

We are now ready to run our experiment. On the Custodian node, start the producer application. The producer application will listen for Interest requests of a namespace -n and reply with Data packets.

```
$ sudo python /local/producer.py -n /nytimes
```

ssh to the Experimenter node, and start the consumer application:

```
$ sudo python /local/consumer.py -u /nytimes/science
```

The Interest packet travels the entire topology, leaving breadcrumbs. The Data packet follows the breadcrumbs back to the consumer, leaving cached versions of the content. This is called in-network caching and it is one of the most important features in Information Centric Networking (ICN). You can check this phenomenon by running the same consumer application in the PI node. ssh to the PI node and start the consumer application:

```
$ sudo python /local/consumer.py -u /nytimes/science
```

This time your PI node gets the content back, but nothing happens on the Custodian because the requested content is cached in the Campus-Router node. Note that the data was retrieved much faster. You can repeat the experiment with different namespaces:

```
$ sudo python /local/consumer.py -u /nytimes/math
```

This time you see that the Interest request is served by the Custodian. Feel free to explore different namespaces.

6 Cleaning up

It is always good practice to release resources you hold so that they can reused by other experimenters. While there are many tools to accomplish this, we will demonstrate the clean up procedure from the Portal. Go to you slice page, and it will bring up the topology. Click on the

Delete button at the bottom, and then click ok to the Delete known slice resources? question.