Anthony Powell

anjopowe

CSCI-P438

November 3, 2014

*GENI Network Architecture & OpenFlow*

The GENI architecture, which is used by OpenFlow, is based around the GENI Rack Aggregate Manager (GRAM). On the software level of this architecture, the servers used are separated into two categories: controller node and compute node. Basically, there is one controller node per rack that serves as the master that manages multiple compute nodes corresponding to that controller. The compute nodes are provide the resources needed by various connections to these servers.

On the hardware side of things, there are four separate networks needed to be used, which are used through the multiple compute nodes on each rack. These four networks are as follows: control network, data network, external network, management network.

Each network allows for different processes and is used in a different way/controlled by a different protocol. The control network, to start, allows commands from OpenFlow and GRAM to be passed between nodes, but is not OpenFlow controlled. The data network, on the other hand, is OpenFlow controlled, and allows for interfaces with virtual machine connections. The management network allows VM’s to connect via SSH, and is also not controlled by OpenFlow. The external network is simply an extra network for connecting to the outside, or public, internet.

These networks are accessed through the interfaces on each of the servers from each rack. The controller nodes each have four interfaces, connecting to three of the four networks above (control, data, and management), as well as to the public network. The compute nodes connect to control, data, and management as well, but not to the public network. They also have an interface open to connect to an external network, but the use of this is optional.

OpenFlow itself is a API that is the first software-defined networking (SDN) standard. It allows users to control switches with OpenFlow protocols. It is based around three objects: switches (OpenFlow enabled and communicates with OpenFlow protocol), controller (service controlled by OpenFlow switch), and interfaces (network interfaces that connected to the switch and by the switch). An OpenFlow protocol works by first taking in incoming packet traffic, and essentially acts as a traffic controller, deciding what to do with and where to send each packet based on the scripts rules. Packets which don’t fit any rules go to the rack’s controller and a default action set by the user is enacted.