(Pre) GENI Lab 8

Due: Although this is a pre-lab, there is something to submit. After watching all my videos, reading Section 0, and completing Section 1, make sure you submit the PDF and JPG files specified in the "What to Turn in" section at the bottom of this document by 3:30pm on Dec 2, 2019.

0. Introduction

0.1. Overview

In this lab, we are going to move down the network protocol hierarchy by another layer and arrive at the **Data Link Layer**.

An Ethernet switch or bridge relays Ethernet frames between devices connected to different ports. It thereby links multiple hosts (or smaller network segments each with multiple hosts) into a single connected network.

When a frame arrives at a switch or a bridge, the source address of the frame is noted.

- If this address is not already in the forwarding table, a new entry is added, including the address and the port on which the frame was received, and an ageing timer is started for that entry.
- If the address is already in the forwarding table, the ageing timer for that entry is restarted. If the port on which the frame was received is different (e.g. if the host has moved to a different port) then the entry is updated to reflect the new port.

The ageing timer is used to remove old entries from the forwarding table. When the "age" value for an entry in the table passes a given threshold, that entry is removed.

Then, the forwarding table is used to determine whether to forward, filter, or flood the frame, depending on its destination address:

- When the switch receives a frame that is destined for an address that is not in the table, the switch will flood the frame out of all ports other than the port on which it arrived.
- If the destination address is in the table, and is known to be reachable on the same port that the frame

was received on, the switch will filter (drop) the frame.
- Otherwise, the switch will forward the frame out of the port corresponding to its destination address in the forwarding table.

Switches and bridges improve performance by partitioning the collision domain, the section of network within which frames collide (destructively) with one another when they are sent simultaneously. Only one host in a collision domain may transmit at a time, in order to avoid collisions. Thus, the total network capacity is shared among all hosts in a collision domain. With a switch or bridge, each port on a switch becomes its own collision domain, and so each network segment can separately support the full network capacity.

0.2. Objectives

In this experimental demonstration of the basic operation of a layer 2 switch/bridge, we will see:

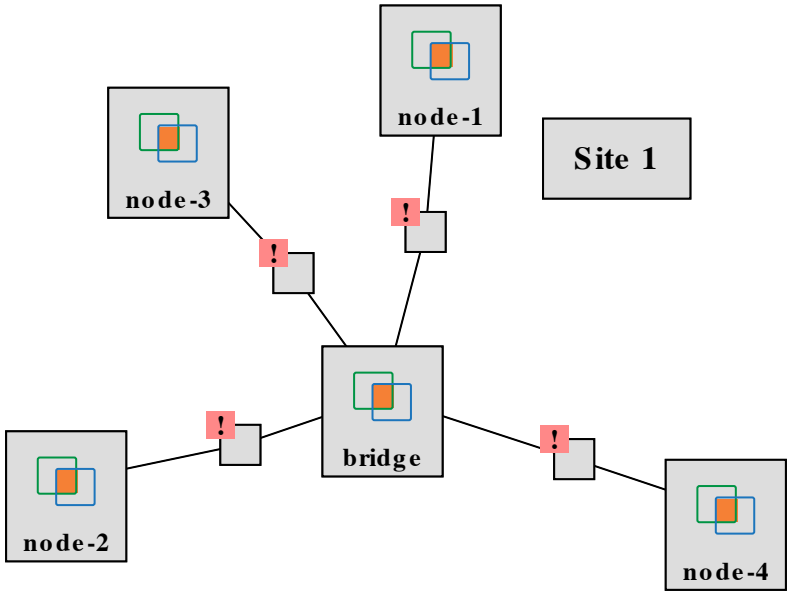
- How to set up a layer 2 bridge on Linux
- How a bridge or switch learns MAC addresses and updates its forwarding table, and how it forwards, filters, or floods a frame depending on the forwarding table
- How a bridge or switch reduces collisions by separating each port into a separate collision domain.

1. Pre-Lab Information

1.1. Lab Configurations

Create a new slice (please name the new slice "**lab8-your-initial**") and request resources by loading the RSpec from the following URL:https://git.io/vyslM.

Your topology may have some red warning indicators on it, like this:



This is not a problem - it's just a warning that some IP addresses are duplicated in the topology. In this case, that's intentional (all the bridge interfaces are assigned an IP address of "0.0.0.0" which results in them having no IPv4 address.) The "host" machines in the topology (node-1, node-2, node-3, node-4) have the IP address 10.0.0.X, where the X is the node number: 10.0.0.1, 10.0.0.2, 10.0.0.3, and 10.0.0.4.

Click on "Site 1" and choose an InstaGENI site to bind to, then reserve your resources. Wait for your nodes to boot up (they will turn green in the canvas display on your slice page in the GENI portal when they are ready).

> Warning: It took me several tries on three different sites to finally get all the resources reserved.

1.2. Set up the Bridge

Use SSH to log in to the "**bridge**" node in your topology, and install the bridge software with:

```
sudo apt-get update
sudo apt-get -y install bridge-utils
```

Since a bridge operates at Layer 2 or Data Link Layer, bridge interfaces should not have an IP address. On the bridge, flush the IP address from each of the four experiment/data interfaces (but be careful not to bring down the control interface):

```
sudo ip addr flush dev eth1
sudo ip addr flush dev eth2
sudo ip addr flush dev eth3
sudo ip addr flush dev eth4
```

Then, create a new bridge interface named `br0` with the command:

```
sudo brctl addbr br0
```

And add the four interfaces to the bridge:

```
sudo brctl addif br0 eth1
sudo brctl addif br0 eth2
sudo brctl addif br0 eth3
sudo brctl addif br0 eth4
```

Finally, we will bring "up" all of the interfaces:

```
sudo ifconfig eth1 up
sudo ifconfig eth2 up
sudo ifconfig eth3 up
sudo ifconfig eth4 up
sudo ifconfig br0 up
```

At this point, you should be able to list the bridge ports with:

```
brctl show br0
```

The output should look something like this:

```
bridge name     bridge id               STP enabled     interfaces
br0             8000.021e6b573974       no              eth1
                                                        eth2
                                                        eth3
                                                        eth4
```

You can also see the `MAC addresses` that the bridge is aware of, with

```
brctl showmacs br0
```

For now, since no traffic has passed through the bridge, it will only know the four MAC addresses

corresponding to the four local interfaces that are on this host (i.e. on the bridge):

```
port no    mac addr        is local?   ageing timer
  4    02:1e:6b:57:39:74   yes         0.00
  1    02:34:7e:04:7b:4e   yes         0.00
  2    02:97:e4:d3:aa:95   yes         0.00
  3    02:cb:96:8d:5b:1f   yes         0.00
```

but it won't know about any other nodes' addresses. (At this point, you can use `ifconfig` to find the `MAC address` of each interface, then match with the output of the command above to determine the "port number" (1, 2, 3, 4) associated with each interface.)

> There is a Google Drawing that you need to fill up with certain information about the connectivity of the network. You may scroll down to the bottom of this document to find out the details.

Verify that all of the end hosts in the topology can reach one another through the bridge. On **each** **of "node-1", "node-2", "node-3", and "node-4"**, run

```
ping -c 1 10.0.0.1
ping -c 1 10.0.0.2
ping -c 1 10.0.0.3
ping -c 1 10.0.0.4
```

Immediately afterwards, the bridge should be aware of every nodes' MAC address, and which port they are located on. On the **"bridge"** node, the output of

```
brctl showmacs br0
```

should show something like this:

```
port no    mac addr        is local?   ageing timer
  4    02:1e:6b:57:39:74   yes         0.00
  1    02:34:7e:04:7b:4e   yes         0.00
  1    02:5d:96:11:77:19   no          1.86
  2    02:97:e4:d3:aa:95   yes         0.00
  2    02:a0:6e:e0:58:93   no          1.89
  3    02:cb:96:8d:5b:1f   yes         0.00
  3    02:cd:b5:11:64:e2   no          1.89
  4    02:da:4f:51:ac:c9   no          1.86
```

Now, you can use `ifconfig` on each of the nodes to find out its MAC address and determine which bridge port (1,2,3,4) each one is connected to.

As time passes without further traffic passing through the bridge, the values in the "ageing timer" column will increase. Once an entry ages past 300 seconds, it will be removed from the forwarding table. So, after 5 minutes with no traffic, the output of

```
brctl showmacs br0
```

again shows only the MAC addresses local to the bridge.

(ACTION) Fill up a Google Drawing using the output of the `brctl showmacs br0` and `ifconfig` commands.

- Go to Google Drawing while signed into your Goolge account.

  Apparently, here I assume you have a Google account.

- Click on "File > Make a copy", Name the document `learning-switch-yourinitial` and choose "My Drive" for Folder.

  Apparently, "My Drive" here implies "Google Drive" therefore I assume you have used it before.

- Fill up the Google Drawing using the output of the `brctl showmacs br0` and `ifconfig` commands. **In the middle of the Google Drawing, insert a text box that reads "Pre Lab8, Your Full Name". No points will be assigned without such a text box.**

> Note: it is **very natural** if for example node 3 is connected to the bridge through port 1 (of the bridge), node 4 through port 2, etc., because the ports and interfaces on the bridge are automatically assigned, over which we don't have any control.

- Finally, click on "File > Download as > PDF Document (.pdf)

(ACTION) Take a screenshot of the bridge terminal and save it in bridge_yourinitial.jpg. Make sure it includes the following:

- The command `brctl showmacs br0` which shows only the local four ports and MAC addresses.
- The command `brctl showmacs br0` again which shows all eight ports and MAC addresses (local plus connected nodes).
- The command `brctl showmacs br0` the third time which shows only the local four ports and MAC addresses after the "ageing timers" have expired.

What to Turn in?

- learning-switch-yourinitial.pdf

- bridge_yourinitial.jpg

2. Lab Details

Te be continued.