Lab 5: BGP routing

50.012 Networks

Hand-out: October 13
eDimension hand-in: October 20, 17pm

1 Objectives & Notes

- Discover Mininet, learn the basics
- Get more familiar with BGP and Zebra tools
- You can work together with another student, but please write and hand in the writeup individually.
- Note: the Mininet part REQUIRES Linux. You can do other parts on your own OS, but Mininet requires Linux.

2 Set up of your machine

2.1 Mininet

1. What is Mininet?
   - Mininet is a network simulator (written in Python), that allows you to define topology and actions in python.
   - The simulated hosts run your OS, and can execute your client/server applications easily.
   - You can also easily change link parameters such a loss, bandwidth, etc.
   - Mininet is mainly intended to experiment with *OpenFlow*, but is also convenient for this lab session. We might look at OpenFlow later in the term.
   - More details at `http://mininet.org`, in particular `http://mininet.org/walkthrough/`
   - If you like videos: some basic things of this sheet are also demo'ed in `https://www.youtube.com/watch?v=jmlgXaocwiE`
2. Installation and first test
   - Install mininet. This requires Ubuntu and Internet access. Mininet can run on many *nixes, but NOT on Windows or Mac
   - The lab machines should have mininet installed already. Nevertheless, it doesn't hurt to check.

```
sudo apt-get install mininet
```

- Start up mininet:

```
sudo mn -x
```

- This will by default spawn 4 new terminals - they belong to each node in a simulated network.

The network contains 2 hosts (h1,h2), one controller C0, and one switch S1. We only need the h1,h2 terminals for now.

- Using these terminals, you can run applications or perform ping operations on the simulated machines
  - To close running commands like **ping -c 100**, try CTRL-C
- Find out the IP addresses of h1 and h2
- To leave/close Mininet, try CTRL-D on the main command line

3 Experiments

3.1 Set up your machine

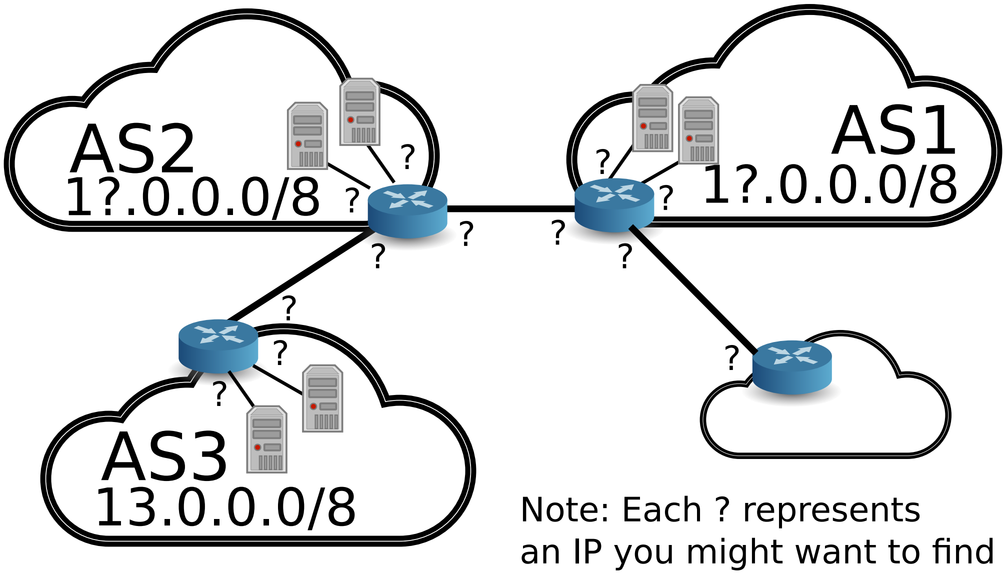
- Connect to SUTD wireless, disconnect the local wired network
- Download the lab5.zip file from eDimension
- Unpack zip file contents into a directory, e.g. ~/**lab5**/. **cd** into that directory.
- Install missing things and set up configuration:
  - Note: you will be asked to provide the password if you are not root already

```
./install.sh
```

- You should now be able to start **sudo python bgp.py**
  - This will start the mininet environment (without xterms this time)

3.2 Getting started

- After starting mininet with the above command, try to find out more about the current topology in mininet using `nodes` and `net` (also note `help`). You can also use ping, zenmap, or similar on the nodes.
  - Annotate the following figure with the AS's announced prefix (network) and the IP addresses of the routers' interfaces. Please note that the "small" AS is not up yet at this stage, so you can't interact with it so much.



- There are also a number of hosts in each network. They represent different smaller AS internal networks connected to internal interfaces of the BGP routers.
- You can connect to the bgp daemons running on the nodes by running something like this: **./connect.sh R1**. The password for the bgpd is **zebra**
  - The command line interface is belonging to **bgpd** (as provided by Zebra), a widely used BGP routing daemon.
  - Experiment around with the different offered commands. Using `show ip bgp`, you can list all IPv4 BGP routes.
- From R1, try to reach 13.0.1.1. Does it work? If no, why not?
- Modify the configuration on R3 to allow R1 to reach 13.0.1.1. Configuration files can be found in the `conf` folder, or try `route` on R3.

3.3 Observing BGP in action

- In mininet, start wireshark sessions on the individual routers, e.g. **R1 wireshark**
  - For some reason, we don't see the simulated interfaces on the hosts (reason unkown)
- Start a wireshark session on one of the routers (make sure to select the right interface), and also open a bgpd command line session to it with the connect script.
- Type "enable" in the bgpd session to enable admin mode (pw: zebra). Look at current routes with **show ip bgp**. Type **clear bgp external** to clear the exchanged routes.
- Watch the bgp traffic establishing the routes again in BGP.

3.4 Malicious BGP abuse

1. Introduction
   - Assume the following setting: a user from AS1 want to visit a website on 13.0.1.1. A malicious attacker wants to redirect the user to its own webserver instead.
   - The attacker has control over AS4, which is BGP-peering with AS1
   - How can the attacker reach his goal?
2. Understanding bgpd and zebra more
   - In **conf**/, you will find a range of configuration files:
     - bgpd-R1.conf and similar, that configure the bgpd setup of each router
     - zebra-R1.conf and similar, that configure the network setup of each router
   - Look at these files, and try to understand what is configured, and how.
   - To perform the attack, you will have to modify bgpd-R4.conf.
3. Performing the attack
   - Use the provided website script in a terminal like this: **./website.sh R1**
     - It will continously contact a webserver on 13.0.1.1 from R1 (if you fixed R3's config). Leave the script running in the background.
   - Open a wireshark session on R1, make sure to listen on all eth interfaces
   - Run **./start_rogue.sh** script in a terminal. Observe the continous website results. Observe the wireshark traffic.
   - If you successfully configured R4, the victim should now see the attack website. Make sure that this is the case.
   - Running **./stop_rogue.sh** will stop the attack again if needed.

4 What to Hand in

4.1 eDimension submission:

Please provide a writeup (in PDF format with your name) that includes the following information:

- The topology as you were able to derive it
  - IP addresses of all routers
  - Hosts/ IPs in the ASs
- What was it initially not possible to reach 13.0.1.1 from AS1? How did you find out/what did you do to fix this?
- Describe the BGP traffic you were able to observe during re-establishment of routes.
- Describe in detail what happened when you started the attack on BGP.

4.2 Checkoff:

- No checkoff required if you submitted your reply sheet