**CS 455 Introduction to Computer Networks**

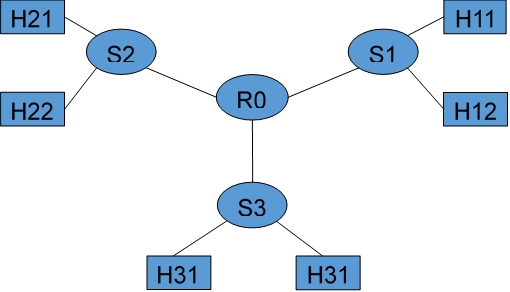
**Course project (counts as 10% of your final course grade)**

**(due: 12/3)**

In this project, we will be writing a program for layer 3 routers by using python and raw socket.

**Mininet Setup**

After setting up the environment, you can create a virtual network with six hosts, three switches, and one router just run the python template file “createNet.py”.



* $ sudo python createNet.py

The default IP and MAC addresses for the router and hosts are as follows:

IP

* h11 192.168.1.100
* h12 192.168.1.101
* …
* h21 192.168.2.100
* h22 192.168.2.101
* …
* h31 192.168.3.100
* h32 192.168.3.101
* …

router (r0)

* intf1 (to s1): 192.168.1.1 00.00.00.00.00.01
* intf2 (to s2): 192.168.2.1 00.00.00.00.00.02
* intf3 (to s3): 192.168.3.1 00.00.00.00.00.03

pull up a terminal on each of your new virtual network hosts, run this in the mininet shell:

* mininet> xterm r0 h11 h12 h21 h22 h31 h32 % display terminal windows

All the operations you perform within these windows are just like if you were using the main Linux VM. They work independently like different machines but share the same files on the machine. Here are some other commands may be useful.

* $ help % display commands
* $ exit % quit
* $ pingall % all the machine ping each other to test connectivity
* $ xterm h11 % display h11 a terminal window

**Project Description**

At a high level, we'll write a raw socket program in Python for a static router with no BGP or OSPF. It does not change the forwarding table dynamically but it can match on the masked IP prefix ranges, just like a router. According to [RFC 1812](http://www.faqs.org/rfcs/rfc1812.html#ixzz0oPra4P9C): An IP router can be distinguished from other sorts of packet switching devices in that a router examines the IP protocol header as part of the switching process. It generally removes the Link Layer header a message was received with, modifies the IP header, and replaces the Link Layer header for retransmission.

**Forwarding key points**

* Decrement TTL and re-compute checksum.
* Find the entry in the forwarding table has the longest prefix match.
* Check the ARP cache for the destination MAC address corresponding to the next-hop IP and modify the destination MAC address.

**In addition to IP forwarding**, your router has to be able to send and respond to ARP requests. If the router is functioning correctly, all of the following operations should work:

* The router should respond correctly to ICMP echo requests.
  + Pinging from any host to any other host
  + Pinging from any host to any of the router's interfaces
* The router should handle TCP/UDP packets sent to one of its interfaces.
  + Downloading a file using HTTP from any host
  + If the destination is not reachable, the router should respond with an ICMP network unreachable reply
* The router must correctly handle traceroutes through it.
  + Tracerouting from any host to any of the router’s interfaces
  + Tracerouting from any host to any other host
* The router should send ARP requests for unknown destination.
  + If the router does not know the MAC address of the destination, it will have to ARP the MAC address for that host
  + Maintain an ARP cache and handle routing accordingly

**Turn-in Instruction**

* You are encouraged to form a team for this project (up to 3 members)
* Submit all the python code files and a Readme document including
  + Name and the last 5 digits of student id for each team member
  + Instruction on how to run your program
  + What is the logic for packet forwarding in your program?
  + How does your program handle ICMP echo request?
  + Is there an ARP cache implemented in your program, how it works?
  + How does your program handle a packet forwarding if its next-hop MAC address corresponding to the next-hop IP is not in the ARP cache?

**Grading**

* Readme document (10%)
* Correctly forward packets (decrement TTL and re-compute checksum) (50%)
* Correctly handle ICMP request (ping and traceroute) (20%)
* Correctly handle ARP request and reply (static ARP cache has no credit) (20%)