SIMPLE ROUTER – PROJECT 2

- Aakash Arayambeth
- TA – CSE 123 (FALL 2018)
- OH (Thu 5-6:30pm B250A)
Basic Forwarding Principles

• Remove IP datagram from Ethernet frame.
  ○ Could also be an ARP payload in Ethernet frame
• Verify checksum. If checksum is invalid, drop the packet.
• Inspect the packet’s Destination Address.
  ○ If destination address is not one of the router’s IP addresses:
    ■ Look up next-hop address by doing a Longest Prefix Match on the routing table using the packet’s Destination Address
    ■ If it does not exist, send ICMP host unreachable
    ■ Decrement TTL, update header checksum
  ○ If TTL == 0 after decrementing, send ICMP time exceeded
  ○ From next-hop IP address, determine outgoing interface and next-hop MAC address
    ■ If necessary, send ARP request to determine MAC address
  ○ Encapsulate IP datagram in Ethernet packet
  ○ Forward packet to outgoing interface
Contd..

• If the packet’s Destination address is one of the routers interfaces:
  ○ If it’s an ICMP echo request, generate an ICMP echo reply
  ○ Otherwise if it’s a TCP or UDP packet, generate an ICMP port unreachable (needed for traceroute to work)
Main Functions and Structures

• In sr_router.h:
  ○ struct sr_instance is the context of the router

• In sr_router.c:
  ○ sr_handlepacket is called for every packet that goes through the router—you have to fill it out

• sr_protocol.h contains convenience structs for accessing fields in packets
  ○ Note: only the basic ICMP header is provided. sr_protocol.h doesn’t include fields/structs for all the various ICMP packet types you’ll need so you’ll have to make your own.

• sr_if.h contains methods for getting information about the router’s interfaces

• ARP Cache in sr_arpcache.h
A Few thoughts:

• Organize your code
  • Sticking everything in sr_router.c will probably give you a headache
  • Make some new files (suggestions, not necessary):
    • sr_arp.c/h for handling/generating ARP packets
    • sr_icmp.c/h for handling/generating ICMP packets
    • sr_ip.c/h for to handle generating IP packets
    • Add the sources and headers to the Makefile
• Do one thing at a time
  • You NEED ARP to send anything at all. A good place to start.
  • If you just do the forwarding path without ICMP, you should be able to route packets to the app servers.
  • You can add ICMP support later on. (Again, just a suggestion)
QUESTIONS?