Week 3 Section Notes: ARP and Lab 3
Overview

- High-level concepts from the week relevant to Lab 3
- VNS/Setting up Lab 3
- Where to start in Lab 3
- Advice and hints
- Questions
IP Addresses

• Each end host on the Internet is assigned an IP address.

• To send a packet to another host on the Internet, you must stamp the packet with the destination host's IP address*.
Question: Is this IP address fixed?
Question: Is this IP address fixed?

No. If I plug my computer into an Ethernet jack in Boston, I will receive a different IP than if I plug into Stanford's network.

Why?
Routing

- IP addresses contain information that routers use to determine how to forward a packet.
- Basic logic: a router that matches the second bit of an IP address is “closer” to the destination than a router that matches the first bit.
Simplified Example

Advertises for:
18.243.1.0/30
18.243.1.0
18.243.1.1
18.243.1.2
18.243.1.3

Default link:
0.0.0.0/0
Simplified Example

<table>
<thead>
<tr>
<th>IP Addr</th>
<th>Subnet</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.243.1.0</td>
<td>255.255.255.252</td>
<td>$\text{IP}_B$</td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>0.0.0.0</td>
<td>$\text{IP}_C$</td>
</tr>
</tbody>
</table>

Advertises for:
- 18.243.1.0/30
- 18.243.1.1
- 18.243.1.2
- 18.243.1.3

Default link:
- 0.0.0.0/0
Longest Prefix Match

- Bitwise and the packet's destination with all subnets in the routing table:
  \[ X_i = (\text{Packet Dest IP}) \& (\text{Subnet}_i) \]

- Forward to next-hop of entry \( n \) with largest IP address that equals \( X_n \).
**Simplified Example**

Advertises for:
- 18.243.1.0/30
  - 18.243.1.0
  - 18.243.1.1
  - 18.243.1.2
  - 18.243.1.3

Default link:
- 0.0.0.0/0

<table>
<thead>
<tr>
<th>IP Addr</th>
<th>Subnet</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.243.1.0</td>
<td>255.255.255.252</td>
<td>IP_B</td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>0.0.0.0</td>
<td>IP_C</td>
</tr>
</tbody>
</table>
How do routers know where the next-hop is ($IP_A$ and $IP_B$ in figure)?
Address Resolution Protocol

- Maps next-hop IP address to which link-layer address to matches that IP.

- Basics. When have next-hop IP address:
  - Check if know corresponding link-layer address.
  - If don't, broadcast on link layer interfaces an ARP request for the IP address.
  - Listen for ARP responses, and route accordingly.
Overview

- High-level concepts from the week relevant to Lab 3
- **VNS/Setting up Lab 3**
- Where to start in Lab 3
- Advice and hints
- Questions
VNS/Setting Up Lab 3

- Read ALL the instructions before beginning.
VNS/Setting Up Lab 3

- Read ALL the instructions before beginning.
- You must test and your labs on the **myth machines**.
- Copy the `auth_key` file from
  
  `/usr/class/cs144/lab3_10au/student_auths/<SUNET_ID@stanford.edu>/auth_key`
  
  to your lab directory
VNS/Setting Up Lab 3 ctd.

- Create a routing table file. Example is online.

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next hop</th>
<th>Mask</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.0</td>
<td>172.24.74.11</td>
<td>0.0.0.0</td>
<td>eth0</td>
</tr>
<tr>
<td>192.168.128.204</td>
<td>192.168.128.204</td>
<td>255.255.255.255</td>
<td>eth1</td>
</tr>
<tr>
<td>192.168.128.206</td>
<td>192.168.128.206</td>
<td>255.255.255.255</td>
<td>eth2</td>
</tr>
</tbody>
</table>
VNS/Setting Up Lab 3 ctd.

- Run the command

  ```
  ./sr -u <username> -T '1-router 2-server' -s vns-2.stanford.edu -r rtable.vrhost -l logname
  ```
Run the command

```
./sr -u <username> -T '1-router 2-server' -s vns-2.stanford.edu -r rtable.vrhost -l logname
```

If you see an error that says “unauthorized user” ...
VNS/Setting Up Lab 3 ctd.

• Run the command

  ./sr -u <username> -T '1-router 2-server' -s vns-2.stanford.edu -r rtable.vrhost -l logname

• If you see an error that says “unauthorized user” …
  blame the TA.
Overview

- High-level concepts from the week relevant to Lab 3
- VNS/Setting up Lab 3
- **Where to start in Lab 3**
- Advice and hints
- Questions
Lab 3 File Overview

- sr_arpcache.h / sr_arpcache.c
  - Handles all the ARP messaging alluded to previously.

- sr_router.h / sr_router.c
  - Where you receive incoming packets.
sr_arpcache

- Pseudocode is included in the sr_arpcache.h file.
sr_arpcache ctd.

- When you receive a packet, you look for the link-layer address of the next-hop IP.
- If you don't find it, you:
sr_arpcache ctd.

- When you receive a packet, you look for the link-layer address of the next-hop IP.
- If you don't find it, you:
  - Queue the packet (call arpcache_queuereq).
sr_arpcache ctd.

- When you receive a packet, you look for the link-layer address of the next-hop IP.
- If you don't find it, you:
  - Queue the packet (call arpcache_queuereq).
  - Broadcast an ARP request.
sr_arpcache ctd.

- When you receive a packet, you look for the link-layer address of the next-hop IP.
- If you don't find it, you:
  - Queue the packet (call arpcache_queuereq).
  - Broadcast an ARP request.

You get to write this!!!
• Question 1: What layer is ARP?

• Question 2: When I send a packet on the Internet, does it always get to the destination?
• Question 1: What layer is ARP?
   
   *It's on the Link Layer.*

• Question 2: When I send a packet on the Internet, does it always get to the destination?
   
   *No! Remember Labs 1 & 2?*
sr_arpcache

- sr_arpcache.c has a function called sr_arpcache_sweepreqs:
  - Called (roughly) every second.
  - If you haven't received a response for your ARP request in more than a second, send another...
  - Unless, you've requested 5 times. Then send an ICMP error message to all sources.
sr_arpcache: Things you'll have to do.

- Broadcast ARP requests.
- Write sr_arpcache_sweepreqs.
- Build a data structure tracking outstanding ARP requests.
- Send ICMP error messages when can't find next-hop router.
- Handle ARP responses.
sr_router.c: sr_handlepacket

- Where all packets enter your router implementation.
- Passed in raw Ethernet frames that may represent:
  - ARP packets
  - ICMP packets
  - IP packets
sr_router.c: sr_handlepacket

• Where all packets enter your router implementation.
• Passed in raw Ethernet frames that may represent
  • ARP packets
  • ICMP packets
  • IP packets

You will eventually have to send/generate/modify all of these.
Overview

• High-level concepts from the week relevant to Lab 3
• VNS/Setting up Lab 3
• Where to start in Lab 3
• Advice and hints
• Questions
Advice and Hints: General

- Start the lab early.
- Read the documentation carefully before starting.
- Check out various packet formats linked to in the lab (ICMP, Ethernet, etc.).
- Come to office hours if you have questions.
Advice and Hints: Debugging

- Use Wireshark! Example.

```
./sr -u <username> -T '1-router 2-server' -s vns-2.stanford.edu -r rtable.vrhost -l logname
```

Enable logging, and open logs in wireshark.
But Wait! There's more!

• What we went over:
  • Basic concepts.
  • Setting up the lab.
  • Where to start.

• There's more:
  • Ping and traceroute.
  • FTP/HTTP
Extra Questions

- Who remembered when I said VNS gave you 8 IP addresses? How come there aren't that many in the figure?
Extra Questions

• What's a multihomed AS?
Extra Questions

• Why wouldn't a router always forward a packet to a router advertising a shorter path?
Extra Questions

• What is the difference between forwarding and routing?
Extra Questions

- What is Nagle's algorithm? Why would people use it?