Lecture 9: Naming

CSE 123: Computer Networks
Alex C. Snoeren

HW 2 due Now; Project 1 due Tuesday

Some material courtesy Mike Freedman
Lecture 9 Overview

- Finish up Route Aggregation
  - CIDR
  - Packet forwarding example

- User-friendly names (DNS)

- Discovering addresses (DHCP/ARP)

- End-to-end lookup and forwarding example
CIDR

- Classless Inter-Domain Routing (1993)
  - Networks described by variable-length prefix and length
  - Allows arbitrary allocation between network and host address

  - e.g. 10.95.1.2/8: 10 is network and remainder (95.1.2) is host

- Pro: Finer grained allocation; aggregation
- Con: More expensive lookup: longest prefix match

Network | Host
--- | ---
Prefix

Mask=# significant bits representing prefix

e.g. 10.95.1.2/8: 10 is network and remainder (95.1.2) is host
Route Aggregation

- Combine adjacent networks in forwarding tables
  - Helps keep forwarding table size down

```
Fly-By-Night-ISP

Organization 0
200.23.16.0/23

Organization 1
200.23.18.0/23

Organization 2
200.23.20.0/23

Organization 7
200.23.30.0/23

ISPs-R-Us

Internet
```

“Send me anything with addresses beginning 200.23.16.0/20”

“Send me anything with addresses beginning 199.31.0.0/16”
Most Specific Route

- But what if address range is not contiguous?

Organization 0
200.23.16.0/23

Organization 2
200.23.20.0/23

Organization 7
200.23.30.0/23

Organization 1
200.23.18.0/23

ISP-R-Us

Fly-By-Night-ISP

“Send me anything with addresses beginning 200.23.16.0/20”

“Send me anything with addresses beginning 199.31.0.0/16 or 200.23.18.0/23”

Internet
Forwarding table contains many prefix/length tuples
  - They *need not* be disjoint!
  - E.g. 200.23.16.0/20 and 200.23.18.0/23
  - What to do if a packet arrives for destination 200.23.18.1?
  - Need to find the longest prefix in the table which matches it (200.23.18.0/23)

Not a simple table, requires multiple memory lookups
  - Lots and lots of research done on this problem
  - Our own George Varghese is the master of this domain
PATRICIA Trie

- Straightforward way to look up LMP
  - Arrange route entries into a series of bit tests
  - Worst case = 32 bit tests
  - Problem: memory speed is a bottleneck

![PATRICIA Trie Diagram]

Bit to test – 0 = left child, 1 = right child

- 0
- 10
  - 128.2/16
  - default 0/0
- 16
  - 128.32/16
  - 128.32.130/24
  - 128.32.150/24
- 19
Forwarding example

- Packet to 10.1.1.3 arrives
- Path is R2 – R1 – H1 – H2
Forwarding example (2)

- Packet to 10.1.1.3
- Matches 10.1.0.0/23

Forwarding table at R2

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>loopback</td>
</tr>
<tr>
<td>Default or 0/0</td>
<td>10.1.0.1</td>
</tr>
<tr>
<td>10.1.8.0/24</td>
<td>interface1</td>
</tr>
<tr>
<td>10.1.2.0/23</td>
<td>interface2</td>
</tr>
<tr>
<td>10.1.0.0/23</td>
<td>10.1.2.2</td>
</tr>
</tbody>
</table>

CSE 123 – Lecture 9: Naming
Forwarding example (3)

- Packet to 10.1.1.3
- Matches 10.1.1.2/31
- Longest prefix match

Routing table at R1

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>loopback</td>
</tr>
<tr>
<td>Default or 0/0</td>
<td>10.1.2.1</td>
</tr>
<tr>
<td>10.1.0.0/24</td>
<td>interface1</td>
</tr>
<tr>
<td>10.1.1.0/24</td>
<td>interface2</td>
</tr>
<tr>
<td>10.1.2.0/23</td>
<td>interface3</td>
</tr>
<tr>
<td>10.1.1.2/31</td>
<td>10.1.1.2</td>
</tr>
</tbody>
</table>

CSE 123 – Lecture 9: Naming
Forwarding example (4)

- Packet to 10.1.1.3
- Direct route
  - Longest prefix match

Routing table at H1

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>loopback</td>
</tr>
<tr>
<td>Default or 0/0</td>
<td>10.1.1.1</td>
</tr>
<tr>
<td>10.1.1.0/24</td>
<td>interface1</td>
</tr>
<tr>
<td>10.1.1.3/31</td>
<td>interface2</td>
</tr>
</tbody>
</table>

CSE 123 – Lecture 9: Naming
Layers of Identifiers

- **Host name** (e.g., www.ucsd.edu)
  - Used by *humans* to specify host of interest
  - Unique, selected by host administrator
  - Hierarchical, variable-length string of alphanumerical characters

- **IP address** (e.g., 128.54.70.238)
  - Used by *routers* to forward packets
  - Unique, topologically meaningful locator
  - Hierarchical namespace of 32 bits

- **MAC address** (e.g., 58:B0:35:F2:3C:D9)
  - Used by *network adaptors* to identify interesting frames
  - Unique, hard-coded identifier burned into network adaptor
  - Flat name space (of 48 bits in Ethernet)
Host name: www.ucsd.edu
- **Domain**: registrar for each top-level domain (e.g., .edu)
- **Host name**: local administrator assigns to each host

IP addresses: 128.54.70.238
- **Prefixes**: ICANN, regional Internet registries, and ISPs
- **Hosts**: static configuration, or dynamic using DHCP

MAC addresses: 58:B0:35:F2:3C:D9
- **OIDs**: assigned to vendors by the IEEE
- **Adapters**: assigned by the vendor from its block
Mapping Between Identifiers

- **Domain Name System (DNS)**
  - Given a host name, provide the IP address
  - Given an IP address, provide the host name

- **Address Resolution Protocol (ARP)**
  - Given an IP address, provide the MAC address
  - To enable communication within the Local Area Network

- **Dynamic Host Configuration Protocol (DHCP)**
  - Automates host boot-up process
  - Given a MAC address, assign a unique IP address
  - … and tell host other stuff about the Local Area Network
Domain Name System (DNS)

- Distributed administrative control
  - Hierarchical name space divided into zones
  - Distributed over a collection of DNS servers

- Hierarchy of DNS servers
  - Root servers
  - Top-level domain (TLD) servers
  - Authoritative DNS servers

- Performing the translations
  - Local DNS servers
  - Resolver software
DNS: Distributed Database

Unnamed root

generic domains

com
edu
org

bar
west
east

foo
my

my.east.bar.edu

country domains

ac
dot
uk
dot
zw

ac
cam
usr

usr.cam.ac.uk

CSE 123 – Lecture 9: Naming
DNS Root Servers

- 13 root servers (see http://www.root-servers.org/)
  - Labeled A through M

  A Verisign, Dulles, VA
  C Cogent, Herndon, VA (also Los Angeles)
  D U Maryland College Park, MD
  G US DoD Vienna, VA
  H ARL Aberdeen, MD
  J Verisign, (11 locations)
  E NASA Mt View, CA
  F Internet Software C. Palo Alto, CA (and 17 other locations)
  K RIPE London (+ Amsterdam, Frankfurt)
  I Autonomica, Stockholm (plus 3 other locations)
  B USC-ISI Marina del Rey, CA
  L ICANN Los Angeles, CA
  M WIDE Tokyo

CSE 123 – Lecture 9: Naming
Using DNS

- Local DNS server ("default name server")
  - Usually near the end hosts who use it
  - Local hosts configured with local server (e.g., /etc/resolv.conf) or learn the server via DHCP

- Client application
  - Extract server name (e.g., from the URL)
  - Do gethostbyname() to trigger resolver code

- Server application
  - Extract client IP address from socket
  - Optional gethostbyaddr() to translate into name
Example

Host at cis.poly.edu wants IP address for gaia.cs.umass.edu!

requesting host

cis.poly.edu

CSE 123 – Lecture 9: Naming
DNS servers are replicated
- Name service available if at least one replica is up
- Queries can be load balanced between replicas

UDP used for queries
- Need reliability: must implement this on top of UDP
- Try alternate servers on timeout
- Exponential backoff when retrying same server

Cache responses to decrease load
- Both at end hosts and local servers
Whence come IP Addresses?

- You already have a bunch from the days when you called Jon Postel and asked for them (e.g. BBN)

- You get them from another provider
  - E.g. buy service from Sprint and get a /24 from one of their address blocks

- You get one directly from a routing registry
  - ARIN: North America, APNIC (Asia Pacific), RIPE (Europe), LACNIC (Latin America), etc.
  - Registries get address from IANA (Internet Assigned Numbers Authority)
How Do You And I Get One?

- Well from your provider!
- But how do you know what it is?
- Manual configuration
  - They tell you and you type that number into your computer (along with the default gateway, DNS server, etc.)
- Automated configuration
  - Dynamic Host Resolution Protocol (DHCP)
Bootstrapping Problem

- Host doesn’t have an IP address yet
  - So, host doesn’t know what source address to use

- Host doesn’t know who to ask for an IP address
  - So, host doesn’t know what destination address to use

- Solution: shout to discover a server who can help
  - Install a special server on the LAN to answer distress calls

CSE 123 – Lecture 9: Naming
DHCP

- Broadcast-based LAN protocol algorithm
  - Host broadcasts “DHCP discover” on LAN (e.g. Ethernet broadcast)
  - DHCP server responds with “DHCP offer” message
  - Host requests IP address: “DHCP request” message
  - DHCP server sends address: “DHCP ack” message w/IP address

- Easy to have fewer addresses than hosts (e.g. UCSD wireless) and to renumber network (use new addresses)

- What if host goes away (how to get address back?)
  - Address is a “lease” not a “grant”, has a timeout
  - Host may have different IP addresses at different times?
Address Resolution Protocol

- Every node maintains an ARP table
  - (IP address, MAC address) pair
- Consult the table when sending a packet
  - Map destination IP address to MAC address
  - Encapsulate and transmit the data packet
- What if the IP address is not in the table?
  - Broadcast: “Who has IP address x.x.x.x?”
  - Sender caches the result in its ARP table
Example: Sending to CNN
Basic Steps

1. Host A must learn the IP address of B via DNS
2. Host A uses gateway R to reach external hosts
3. Router R forwards IP packet to outgoing interface
4. Router R learns B’s MAC address and forwards frame
Host A Learns B’s IP Address

- Host A does a DNS query to learn B’s address
  - Suppose gethostbyname() returns 222.222.222.222
- Host A constructs an IP packet to send to B
  - Source 111.111.111.111, dest 222.222.222.222
Host A Learns B’s IP Address

- **IP packet**
  - From A: 111.111.111.111
  - To B: **222.222.222.222**

- **Ethernet frame**
  - From A: 74-29-9C-E8-FF-55
  - To gateway: ????
Host $A$ has a gateway router $R$
- Used to reach dests outside of 111.111.111.0/24
- Address 111.111.111.110 for $R$ learned via DHCP

But, what is the MAC address of the gateway?
Host A learns the MAC address of R’s interface
- ARP request: broadcast request for 111.111.111.110
- ARP response: R responds with E6-E9-00-17-BB-4B

Host A encapsulates the packet and sends to R
A Sends Packet Through R

- **IP packet**
  - From A: 111.111.111.111
  - To B: 222.222.222.222

- **Ethernet frame**
  - From A: 74-29-9C-E8-FF-55
  - To R: E6-E9-00-17-BB-4B

CSE 123 – Lecture 9: Naming
Router $R$’s adapter receives the packet
- $R$ extracts the IP packet destined to 222.222.222.222
Router $R$ consults its forwarding table
- Packet matches 222.222.222.0/24 via other interface
R Wants to Forward Packet

- **IP packet**
  - From A: 111.111.111.111
  - To B: 222.222.222.222

- **Ethernet frame**
  - From R: 1A-23-F9-CD-06-9B
  - To B: ???
Router $R$’s learns the MAC address of host $B$

- ARP request: broadcast request for 222.222.222.222
- ARP response: $B$ responds with 49-BD-D2-C7-56-2A

Router $R$ encapsulates the packet and sends to $B$
R Wants to Forward Packet

- IP packet
  - From A: 111.111.111.111
  - To B: 222.222.222.222

- Ethernet frame
  - From R: 1A-23-F9-CD-06-9B
  - To B: 49-BD-D2-C7-56-2A

CSE 123 – Lecture 9: Naming
Summary

- Domain Name System
  - Distributed, hierarchical database
  - Distributed collection of servers
  - Caching to improve performance

- IP to MAC Address mapping
  - Dynamic Host Configuration Protocol (DHCP)
  - Address Resolution Protocol (ARP)
For Next Time

- Read 5.1-2 in P&D

- Finish up Project 1!
  - Lose a letter grade for each day late.