Scaling the Internet Globally
Chapter 4.3
Peterson and Davie

CS/ECPE 5516
Spring 2000
Outline for Tuesday, 27 March 2000

- **10 minutes** Checkpoint: How's class going?

- **15 minutes** Sub-lecture:
  - Subnets (chopping up class B address spaces)
  - Supernets (coalescing class C address spaces)

- **15 minutes** Group exercise:
  - Building your own global Internet

- **10 minutes** IP version 6
How's Class Going

■ How was the midterm?

■ How did project 1 go…
  ● To hard/easy?
  ● Educational?
  ● Suggestions?

■ How have class meetings been going?
  ● Has the class material been too easy, too hard, too slow, too fast, …?

■ Honor code
You studied routing in a homogeneous IP net

Does that scale to...
- Small nets
- World-wide nets
Let's Look at IP Addresses

- In 70's/80's, IP designers chose 2-level hierarchical addresses:

<table>
<thead>
<tr>
<th>Network number</th>
<th>Host number</th>
</tr>
</thead>
</table>

- Why was this a very bad thing by 2000?

  - Is the hierarchy or the 2 levels or both bad?
Explosion in # nets => Running out of IP address space!

Fundamental IP design choice:
- Class C addresses (2^{24} nets of <2^{8} hosts)
- Class B addresses (2^{16} nets of <2^{16} hosts)
- Every time a C or B net # is assigned, it kills a whole range of IP addresses

Inefficiency results:
- class C with 2 hosts (2/255 = 0.78% efficient)
- class B with 256 hosts (256/65535 = 0.39% efficient)
Very Bad Thing #1

- **Explosion in # nets => Routing doesn't scale!**

<table>
<thead>
<tr>
<th>Network number</th>
<th>Host number</th>
</tr>
</thead>
</table>

- **Recall the "beauty" of IP routing:**
  - Routing tables are based on # nets, not #hosts
  - But #nets is in tens of thousands today = big tables

- **Problems**
  - Routing tables do not scale
  - Route propagation protocols do not scale
Ways to Solve Problems

- Running out of IP addresses...
  - Assign multiple & contiguous class-C addresses to a new network
  - Temporary solution until IPv6 with 128-bit IP addresses

- Scaling routing...
  - Add 3rd level to IP hierarchy
  - Example:
    - Assign ISP one class B address
    - ISP subnets address space for $2^{16}$ small companies, each having up to $2^{10}$ hosts
    - Without subnetting, need 64 class B addresses!
Subnetting (To solve route scaling)

- Add another level to address/routing hierarchy: \textit{subnet}
- \textit{Subnet masks} define variable partition of host part
- Outside net w/ subnets, subnetting is invisible

<table>
<thead>
<tr>
<th>Network number</th>
<th>Host number</th>
</tr>
</thead>
</table>

Class B address

\begin{tabular}{c|c}
11111111111111111111111111 & 00000000 \\
\end{tabular}

Subnet mask (255.255.255.0)

<table>
<thead>
<tr>
<th>Network number</th>
<th>Subnet ID</th>
<th>Host ID</th>
</tr>
</thead>
</table>

Subnetted address
Subnet Example (Fig. 4.25)

Subnet #: 128.96.34.0

Subnet #: 128.96.34.128

Subnet #: 128.96.33.0
Subnet Example

Forwarding table at router R1:

<table>
<thead>
<tr>
<th>Subnet #</th>
<th>Mask</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.96.34.0</td>
<td>255.255.255.128</td>
<td>interface 0</td>
</tr>
<tr>
<td>128.96.34.128</td>
<td>255.255.255.128</td>
<td>interface 1</td>
</tr>
<tr>
<td>128.96.33.0</td>
<td>255.255.255.0</td>
<td>R2</td>
</tr>
</tbody>
</table>
Subnet Example (Fig. 4.25)

If (Dest & Mask) = Subnet-Number then use Next-Hop:
(Dest: 128.96.34.15 & Mask: 255.255.255.128) = 128.96.34.0, so use interface 0
Forwarding Algorithm

D = destination IP address
for each entry (SubnetNum, SubnetMask, NextHop)
   D1 = SubnetMask & D
   if D1 = SubnetNum
      if NextHop is an interface
         deliver datagram directly to D
      else
         deliver datagram to NextHop
   else
     deliver datagram to NextHop

- Use a default router if nothing matches
- Not necessary for all 1s in subnet mask to be contiguous
- Can put multiple subnets on one physical network—subdivides LAN so traffic is forced through router even though direct delivery possible
Supernetting (To solve running out)

- Supernetting is used to aggregate routes
  - Assign block of contiguous network numbers to nearby networks
- Called CIDR:
  - Classless Inter-Domain Routing
- Restrict block sizes to powers of 2
- Use a contiguous bit mask (CIDR mask) to identify block size
- All routers must understand CIDR addressing
  - Routing involves *longest prefix match*
## Group Exercise on 4.3.3

### Divide yourselves into groups; fill in table:

<table>
<thead>
<tr>
<th>How many backbones?</th>
<th>Internet in 1989</th>
<th>in 2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal of routing is ???</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distance vector or shortest-path run over all netids?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>True/false: every backbone router's table lists every network number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>True/false: economic incentive exists to do evil routing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Internet Structure

Internet in 1990

- NSFNET backbone
- BARRNET regional
- Berkeley
- PARC
- NCAR
- Westnet regional
- UNM
- UA
- MidNet regional
- UNL
- KU
- ISU
- Stanford
Internet Structure

Today: multiple "Autonomous Systems" (ASes)

1. Stub AS
2. MultiHomed AS
3. Transit AS
The Internet is composed of many autonomous systems (AS)
- Each AS is operated largely independently
- Policies and even protocols within an AS may vary
- Some standardization is needed, of course

Types of routing protocols
- Interior Gateway Protocol (IGP): a routing protocol operating within an AS
- Exterior Gateway Protocol (EGP) or Interdomain Routing Protocols: a protocol to route between ASes
See Stallings Fig. 16-10

See link on Web site
Popular **Interior Gateway Protocols**

- **RIP: Route Information Protocol**
  - developed for XNS
  - distributed with Unix
  - distance-vector algorithm
  - based on hop-count

- **OSPF: Open Shortest Path First**
  - recent Internet standard
  - uses link-state algorithm
  - supports load balancing
  - supports authentication
EKP: Exterior Gateway Protocol

- Overview
  - designed for tree-structured Internet
  - concerned with \textit{reachability}, not optimal routes

- Protocol messages
  - neighbor acquisition: one router requests that another be its peer; peers exchange reachability information
  - neighbor reachability: one router periodically tests if the another is still reachable; exchange HELLO/ACK messages; uses a k-out-of-n rule
  - routing updates: peers periodically exchange their routing tables (distance-vector)
BGP-4: Border Gateway Protocol

Each AS has:
- one or more border routers
- one BGP *speaker* that advertises:
  - local networks
  - other reachable networks (transit AS only)
  - gives *path* information

Path = "reach netids 128.173 & 192.4.3 via (AS1, AS2)"

Note that between ASes, they use entire AS as nodes in routing list like (AS1, AS2).
BGP Example

- **Speaker for AS2 advertises reachability to P and Q**
  - network 128.96, 192.4.153, 192.4.32, and 192.4.3, can be reached directly from AS2

- **Speaker for backbone advertises**
  - networks 128.96, 192.4.153, 192.4.32, and 192.4.3 can be reached along the path (AS1, AS2).

- **Speaker can cancel previously advertised paths**
Evil Routing Examples

- I'm a transit AS.  
  I get paid by # packets I transit, not latency

- So my optimal decision:  
  Anytime anyone sends me a packet, I immediately dump it back out on someone else's AS!

- So end-user routes could get longer!

- Or:  I'm a transit AS.  I compete with you.  I'll manipulate my advertisements to make you vanish/become swamped.

- Or:  I've hired an incompetent router programmer.  So I'll still make you vanish or become swamped!
IP Version 6

- Features
  - 128-bit addresses (classless)
  - multicast
  - real-time service
  - authentication and security
  - autoconfiguration
  - end-to-end fragmentation
  - protocol extensions

- Header
  - 40-byte “base” header
  - extension headers (fixed order, mostly fixed length)
    - fragmentation
    - source routing
    - authentication and security
    - other options