Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network layer

- delivers segments from sending to receiving host
  - sender encapsulates segments into datagrams
  - Receiver de-encapsulates and delivers segments to transport layer
- network layer in every host, every router
- router examines IP header field in every passing datagram (exception: routers running MPLS)
Key Network-Layer Functions

- **forwarding**: move a packet from router’s input interface to an appropriate output interface
  - global address space and a packet format

- **routing**: determine route taken by packets from source to destination
  - routing protocols (intra-AS and inter-AS) where AS is acronym for “Autonomous System”
  - every AS runs the same inter-AS protocol

Interplay between routing and forwarding

- A routing protocol is a distributed algorithm
- Recent development: a Software Defined Network (SDN) uses a central controller (i.e. server) to compute routes
Virtual-circuit networks need 3rd function

- Before datagrams can flow, end hosts and routers between them establish a virtual circuit
  - Routers maintain state info
  - Earlier networks designed initially to compete with IP:
    - ATM, frame relay, X.25 (from old to very old)
  - MPLS protocol designed more recently to provide virtual circuits to enhance IP routing (typically within the same AS)
- Today, such virtual circuits serve as virtual links in Internet

Network layer service models:

<table>
<thead>
<tr>
<th>Network Architecture</th>
<th>Service Model</th>
<th>Guarantees?</th>
<th>Congestion feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Bandwidth</td>
<td>loss</td>
</tr>
<tr>
<td>Internet</td>
<td>best effort</td>
<td>none</td>
<td>no</td>
</tr>
<tr>
<td>ATM</td>
<td>CBR</td>
<td>constant rate</td>
<td>yes</td>
</tr>
<tr>
<td>ATM</td>
<td>VBR</td>
<td>guaranteed rate</td>
<td>yes</td>
</tr>
<tr>
<td>ATM</td>
<td>ABR</td>
<td>guaranteed minimum</td>
<td>no</td>
</tr>
<tr>
<td>ATM</td>
<td>UBR</td>
<td>none</td>
<td>no</td>
</tr>
</tbody>
</table>

3/21/2016

Network Layer (SSL) 4-5

3/21/2016

Network Layer (SSL) 4-6
Origins of datagram and VC

Internet (datagram)
- data exchange between computers
  - “elastic” service, no strict timing requirement
- many link types
  - different characteristics
  - uniform service difficult
- “smart” end systems (computers)
  - can adapt, perform control, error recovery
  - simplicity inside network, complexity at “edge”

ATM (VC)
- evolved from telephony
- human conversation:
  - strict timing, reliability requirements
  - need for guaranteed services
- “dumb” end systems
  - telephones
  - complexity inside network

Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing
Datagram networks
- no network-level concept of "connection" or "flow"
- each packet forwarded independently using destination host address
  - packets between same source-dest pair may take different paths

Forwarding table
4 billion possible entries

<table>
<thead>
<tr>
<th>Destination Address Range</th>
<th>Link Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>001000 00010111 00010000</td>
<td>0</td>
</tr>
<tr>
<td>00000000</td>
<td></td>
</tr>
<tr>
<td>11001000 00010111 00010111</td>
<td>0</td>
</tr>
<tr>
<td>00000000</td>
<td></td>
</tr>
<tr>
<td>11001000 00010110 00010110</td>
<td>1</td>
</tr>
<tr>
<td>00011111</td>
<td></td>
</tr>
<tr>
<td>11001000 00010110 00011000</td>
<td>2</td>
</tr>
<tr>
<td>00111111</td>
<td></td>
</tr>
<tr>
<td>11001000 00010111 00011011</td>
<td>otherwise</td>
</tr>
<tr>
<td>00000000</td>
<td></td>
</tr>
<tr>
<td>11001000 00010111 00011111</td>
<td></td>
</tr>
</tbody>
</table>

3/21/2016
Network Layer (SSL) 4-9

3/21/2016
Network Layer (SSL) 4-10
**Longest prefix match**

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Link Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>11001000 00010111 00010</code></td>
<td>0</td>
</tr>
<tr>
<td><code>11001000 00010111 00011000</code></td>
<td>1</td>
</tr>
<tr>
<td><code>11001000 00010111 00011</code></td>
<td>2</td>
</tr>
<tr>
<td>otherwise</td>
<td>3</td>
</tr>
</tbody>
</table>

**Examples**

- **DA:** `11001000 00010111 00011000 10101001`  
  Which interface?  

- **DA:** `11001000 00010111 00011000 10101010`  
  Which interface?  

A forwarding table in an Internet core router has more than 400,000 IP prefixes *(from 2014 data)*

---

**Virtual circuits: signaling protocols**

- Used to set up, maintain, tear down VC  
- Not used in Internet’s network layer (but may be used underneath the IP layer to provide a virtual link)

1. **Initiate call**  
2. **Incoming call**  
3. **Accept call**  
4. **Call connected**  
5. **Data flow begins**  
6. **Receive data**

[Diagram of a virtual circuit with layers: application, transport, network, data link, physical]
Virtual circuit (VC)

- call setup, teardown for each call before data can flow
- each packet carries a VC identifier which
  - is fixed length and short
  - only needs to be unique for a link
  - is carried in an additional header inserted between link and network layer headers

- every router on source-dest path maintains state information for each passing VC
  - incoming and outgoing VC identifiers,
  - resources allocated to VC (bandwidth, buffers)

VC Forwarding table

Forwarding table in northwest router:

<table>
<thead>
<tr>
<th>Incoming interface</th>
<th>Incoming VC #</th>
<th>Outgoing interface</th>
<th>Outgoing VC #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>2</td>
<td>63</td>
<td>1</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>2</td>
<td>17</td>
</tr>
<tr>
<td>1</td>
<td>97</td>
<td>3</td>
<td>87</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- Forwarding is fast because short fixed-length VC numbers are used vs. IP forwarding table with variable-length prefixes
- May have additional state information about service guarantees
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Router architecture overview

Two key router functions:
- run routing protocols (RIP, OSPF, BGP)
- forward datagrams from incoming to outgoing link

Network Layer (SSL) 4-15

Network Layer 4-16
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

The Internet Network layer

Host, router network layer functions:

Transport layer: TCP, UDP

- Routing protocols
  - Path selection
  - RIP, OSPF, BGP

IP protocol
- Addressing conventions
- Datagram format
- Packet handling conventions

ICMP protocol
- Error reporting
- Router "signaling"

Forwarding table

Link layer

Physical layer

3/21/2016
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

IP datagram format

- IP protocol version number
- Header length
- "Type" of data
- Max number remaining hops (decremented at each router)
- Upper layer protocol to deliver payload to
- 32-bit source IP address
- 32-bit destination IP address
- Options (if any)
- Data (variable length, typically a TCP or UDP segment)
- Total datagram length (bytes)
- Time to live
- Upper layer checksum
- Flags
- Fragment offset
- Type of service
- 16-bit identifier
- Head length

Image 3/21/2016

Network Layer (SSL) 4-19

Image 3/21/2016

Network Layer (SSL) 4-20
IP Fragmentation & Reassembly

- MTU (max. transfer size)
  - Different link types, different MTUs
  - Support MTU of at least 576 bytes

- Too large IP datagram “fragmented” within net
  - Reassembled only at final destination
  - IP header bits used to identify, order related fragments

Example:

- 3980 bytes of data
- 4000 byte datagram
- MTU = 1500 bytes

One large datagram becomes several smaller datagrams

- 1480 bytes in data field
  - Offset = 1480/8

3/21/2016
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

3/21/2016

IP Addressing: introduction

- **IP address**: 32-bit identifier for an interface
- **interface**: connection between host/router and physical link (wired or wireless)
  - a router typically has multiple interfaces
  - a host typically has one interface

Dotted decimal notation

\[
223.1.1.1 = 11011111 \ 00000001 \ 00000001 \ 00000001
\]

3/21/2016
Subnets

- **IP address:**
  - subnet part (high order bits)
  - host part (low order bits)

- **What's a subnet?**
  - device interfaces with same subnet part of IP address
  - can physically reach each other without a router

Subnets

**Recipe**

- To determine the subnets, detach each interface from its host or router, creating islands of isolated networks. Each isolated network is a subnet.

Note: There are virtual LANs (VLANs) - see Chapter 5
Subnets
How many?

IP addressing: CIDR

Classful addressing (now obsolete): fixed-length subnet portion of 8, 16 or 24 bits

CIDR: Classless InterDomain Routing
- subnet portion of address of variable length
- address format: a.b.c.d/x, where x is # bits in subnet portion of address

```
11001000  00010111 00010000  00000000
```

```
200.23.16.0/23
```
**IP addresses: how to get one?**

**Q:** How does host get IP address?

- hard-coded by system admin in a file
- **DHCP:** Dynamic Host Configuration Protocol: dynamically get address from a server
  - “plug-and-play”

**DHCP client-server scenario**

A router may act as a relay agent
DHCP client-server scenario

DHCP server: 223.1.2.5

DHCP discover

src: 0.0.0.0, 68
dest: 255.255.255.255, 67
yiaddr: 0.0.0.0
tx ID: 654

DHCP offer

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddr: 223.1.2.4
tx ID: 654
Lifetime: 3600 secs

DHCP request

src: 0.0.0.0, 68
dest: 255.255.255.255, 67
yiaddr: 223.1.2.4
tx ID: 655
Lifetime: 3600 secs

DHCP ack

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddr: 223.1.2.4
tx ID: 655
Lifetime: 3600 secs

3/21/2016

Discover & offer messages are optional

DHCP: more than IP address

DHCP can return more than just an allocated IP address on subnet:

- address of first-hop router for client
- name and IP address of DNS server
- network mask (indicating subnet portion of address)

3/21/2016
IP addresses: how to get them?

- **ICANN** (Internet Corporation for Assigned Names and Numbers)/IANA (Internet Assigned Numbers Authority)
  - allocates **IP addresses** (IANA IPv4 address exhaustion on 1/31/2011)
  - oversees DNS
    - root name servers, top level domain names
    - domain name registrars, resolves disputes

- **Regional, national, and local Internet registries, and ISPs**
  - End-user organization can be assigned IP address space from one of the above

3/21/2016

Network Layer (SSL) 4-33

---

IP address prefix: how to get one?

**A:** Typically, a customer network gets allocated a portion of its provider ISP's address space

<table>
<thead>
<tr>
<th>ISP's block</th>
<th>Organization 0</th>
<th>Organization 1</th>
<th>Organization 2</th>
<th>...</th>
<th>Organization 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>11001000_00010111_00010000_00000000</td>
<td>200.23.16.0/20</td>
<td>200.23.16.0/23</td>
<td>200.23.18.0/23</td>
<td>200.23.20.0/23</td>
<td>200.23.30.0/23</td>
</tr>
</tbody>
</table>

3/21/2016

Network Layer (SSL) 4-34
Hierarchical addressing: route aggregation

allows efficient advertisement of routing information

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

"Send me anything with address beginning 200.23.16.0/20"

"Send me anything with address beginning 199.31.0.0/16"

Fly-By-Night-ISP

ISPs-R-Us

Internet

Network Layer (SSL) 4-35
3/21/2016

Hierarchical addressing: more specific routes

ISPs-R-Us has a more specific route to Organization 1
Hole(s) in a block of addresses <- reason for longest prefix match

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

"Send me anything with address beginning 200.23.16.0/20"

"Send me anything with address beginning 199.31.0.0/16 or 200.23.18.0/23"

Fly-By-Night-ISP

ISPs-R-Us

Internet

Network Layer (SSL) 4-36
3/21/2016
NAT: Network Address Translation

Motivation: local network uses just one IP address as far as outside world is concerned

- can change addresses of devices in local network without notifying outside world
- can change ISP without changing addresses of devices in local network
- devices inside local net not explicitly addressable/visible by outside world (a security plus).
**NAT: Network Address Translation**

1. **host 10.0.0.1** sends datagram with port number 3345

   **NAT translation table**
   
<table>
<thead>
<tr>
<th>WAN side addr</th>
<th>LAN side addr</th>
</tr>
</thead>
<tbody>
<tr>
<td>138.76.29.7, 5001</td>
<td>10.0.0.1, 3345</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

2. **NAT router** changes datagram's source addr and port number

   S: 138.76.29.7, 5001
   D: 128.119.40.186, 80

3. **Reply arrives for 138.76.29.7, 5001**

4. **NAT router** changes datagram's dest addr and port number to 10.0.0.1, 3345

---

**16-bit port-number field:**
- 60,000+ simultaneous connections with a single IP address

**NAT is controversial:**
- routers should only process up to layer 3
  - violates "end-to-end argument"
- NAT possibility must be taken into account by app designers, e.g., IPsec, P2P applications
- address shortage should instead be solved by IPv6

3/21/2016
**NAT traversal problem**

- **client wants to connect to server with address 10.0.0.1**
  - only one externally visible IP address: 138.76.29.7

- **configure NAT to forward incoming connection requests at given port to server**
  - e.g., (123.76.29.7, port 2500) always forwarded to 10.0.0.1 port 2500

---

**solution 3: relaying (used in original Skype)**

- relay bridges packets between two connections

  1. connection to relay initiated by host behind NAT
  2. connection to relay initiated by client
  3. relaying established

- Both hosts may be behind NATs
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

3/21/2016

ICMP: Internet Control Message Protocol

- "above" IP in network layer
  - ICMP msgs carried in IP datagrams
  - error reporting: unreachable host, network, port, protocol
  - echo request/reply (used by ping)
- ICMP message
type, code plus first 8 bytes of IP datagram causing error

<table>
<thead>
<tr>
<th>Type</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>echo reply (ping)</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>dest. network unreachable</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>dest host unreachable</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>dest protocol unreachable</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>dest port unreachable</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>dest network unknown</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>dest host unknown</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>source quench (congestion control - not used)</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>echo request (ping)</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>route advertisement</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>router discovery</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>TTL expired</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>bad IP header</td>
</tr>
</tbody>
</table>

3/21/2016
Traceroute uses ICMP messages

- Source sends series of UDP segments to dest
  - First has TTL = 1
  - Second has TTL=2, ..., each with unlikely port number
- When nth datagram arrives to nth router:
  - Router discards datagram and
  - Sends to source a "TTL expired" message with name of router & IP address
- When "TTL expired" message arrives, source calculates RTT
- Traceroute does this 3 times for each TTL value

Stopping criterion
- Such a UDP segment arrives at destination host
- Destination returns msg "dest port unreachable" packet
- Upon receipt of this msg, source stops.

3/21/2016

Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing
IPv6

- **Initial motivation:** 32-bit address space soon to be completely allocated.
- **Additional motivation:**
  - simpler header format to speed up processing/forwarding
  - header change to facilitate QoS
- **IPv6 datagram format:**
  - fixed-length 40 byte header
  - no fragmentation allowed

IPv6 Header (Cont)

- **Priority:** identify priority of datagrams within flow or in different apps
- **Flow Label:** identify datagrams in same “flow.” (concept of “flow” not defined).
- **Next header:** identify upper layer protocol for data

```
ver  pri  flow label
payload len  next hdr  hop limit
source address (128 bits)
destination address (128 bits)
data
```

3/21/2016

Network Layer (SSL) 4-47

Network Layer (SSL) 4-48
Other Changes from IPv4

- **Checksum**: removed entirely to reduce processing time at each hop
- **Options**: allowed, but outside of header, indicated by “Next Header” field
- **ICMPv6**: new version of ICMP
  - additional message types, e.g. “Packet Too Big”
  - including *multicast group management* functions

Transition From IPv4 To IPv6

- Not all routers can be upgraded simultaneously
  - no “flag day”
  - How will the network operate with mixed IPv4 and IPv6 routers?

- **Tunneling**: IPv6 carried as payload in IPv4 datagram among IPv4 routers
Tunneling

Logical view:

Physical view:

Routers B and E have dual stacks. In this example, B encapsulates v6 packet in v4 packet. Later E extracts v6 packet from v4 packet.
**Concept - Tunnel as a virtual link**

Many possibilities:
- IPv6 in IPv4 tunnel (previous example)
- IPv4 in IPv6 tunnel
- IPv4 in IPv4 tunnel
- IPv4 in MPLS tunnel

...
Graph abstraction

Graph: $G = (N, E)$

- $N =$ set of routers = { u, v, w, x, y, z }
- $E =$ set of links ={ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) }

Remark: Graph abstraction is also useful in other network contexts
Example: P2P, where N is set of peers and E is set of TCP connections

Network Layer (SSL) 4-55

Graph abstraction: link costs

- $c(x,x') =$ cost of link $(x,x')$
- Cost could be 1, or inversely related to bandwidth, or inversely related to "congestion", etc.

Cost of path $(x_1, x_2, x_3, ..., x_p) = c(x_1,x_2) + c(x_2,x_3) + ... + c(x_{p-1},x_p)$

Routing protocol tries to find least-cost path

Comments:
- full-duplex links may not have symmetric costs
- cost of path computation is ad hoc if the link cost metric is not additive

Network Layer (SSL) 4-56
Routing Algorithm classification

Global or decentralized information?

Global info:
- all routers have complete topology, link costs
- link state protocols

Decentralized info:
- router knows physically-connected neighbors, link costs to neighbors
- distance vector protocols

Static or dynamic?

- Static - update only after topology change
- Dynamic
  - periodic update
  - in response to link cost changes
  - may result in route flaps

Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing
A Link-State Routing protocol

- net topology, link costs known to every node
  - accomplished via link state broadcast
  - all nodes have same info

Dijkstra’s algorithm

- computes least cost paths from one node (“source”) to all other nodes in a graph
  - iterative: after k iterations, source knows least-cost paths to k destinations
  - yields forwarding table for source node

Link State Broadcast

Flooding

- Source node of “link state” sends packet to all neighbors
- Intermediate node resends to neighbors except where packet arrived
- Many duplicates which must be recognized by nodes
Distance Vector Algorithm basis

Bellman-Ford Equation (dynamic programming)

Define
\[ d_x(y) := \text{cost of least-cost path from } x \text{ to } y \]

Then

\[ d_x(y) = \min_v \{ c(x,v) + d_v(y) \} \]

where \( \min \) is taken over all neighbors \( v \) of \( x \)
Bellman-Ford example

Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

B-F equation says:

$$d_u(z) = \min \{ c(u,v) + d_v(z), c(u,x) + d_x(z), c(u,w) + d_w(z) \}$$

$$= \min \{ 2 + 5, 1 + 3, 5 + 3 \} = 4$$

The node that achieves minimum is next hop in shortest path ➜ put it in forwarding table

Distance Vectors Protocol (1)

- Node $x$
  - knows cost to each neighbor $v$: $c(x,v)$
  - sends its own distance vector (DV) estimate $[D_x(y): y \in N]$ to its neighbors periodically
    where $D_x(y)$ denotes estimate of least cost from $x$ to $y$

- From each neighbor $v$, $x$ receives $[D_v(y): y \in N]$
Distance Vector Protocol (2)

- When a node \( x \) receives a new DV estimate from a neighbor, it updates its own DV estimate using B-F equation:

\[
D_x(y) \leftarrow \min_v \{ c(x,v) + D_v(y) \} \quad \text{for each node } y \in N
\]

If the \( v \) that achieves least cost to \( y \) is new, node \( x \) updates its forwarding table and DV.

- Eventually, assuming that link costs and topology do not change, the estimate \( D_x(y) \) converges to the actual least cost \( d_x(y) \) for all \( x, y \).

Distance Vector Protocol - summary

Distributed, asynchronous, iterative

**Initially**, \( D_x(y) = c(x, y) \) if \( x \) and \( y \) are direct neighbors; otherwise, \( D_x(y) = \infty \)

Each node:

- **waits** for a change in local link cost or a msg from a neighbor
- **recomputes** estimates
- if DV estimate for any dest has changed, updates its own state and **notifies** its neighbors

3/21/2016
\[
D_x(y) = \min(c(x,y) + D_y(y), c(x,z) + D_z(y))
\]
\[
D_x(z) = \min(c(x,y) + D_y(z), c(x,z) + D_z(z))
\]

Each row in a table is a distance vector.

Assume synchronous operations for this example - nodes y and z also received their DV updates.

Network Layer (SSL) 4-67

No more change
**Distance Vector: good news travels fast**

- **y** detects a lower link cost to **x**, updates its **DV**, and sends new **DV** to node **z**.
- **z** receives **y**'s updated **DV**, updates its own **DV**, and sends new **DV** to its neighbors.
- Later, **y** receives **z**'s updated **DV**. **y**'s least cost does not change.
- A similar interaction between nodes **x** and **z**.
- The **DV** protocol converges quickly.

---

**Distance Vector: “count to infinity” problem**

- **Link cost increase:**
  - **Y** still has *stale* information saying that it can go to **X** via **Z** in 6
  - 44 iterations (msg exchanges between **y** and **z**) before protocol stabilizes
- **Poisoned reverse:**
  - If **Z** routes through **Y** to get to **X**:
    - **Z** tells **Y** its (Z’s) distance to **X** is infinite (so **Y** won't route to **X** via **Z**)
  - will this completely solve count to infinity problem?
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Hierarchical network

Our routing study thus far
- all routers identical
- network "flat"
  ... not true in practice

scale: hundreds of millions destinations:
- can’t store all dest’s in forwarding tables
- Link State and Distance Vector do not scale

administrative autonomy
- internet is a network of networks
- each network admin may want to control routing in its own network
Hierarchical Routing

- Aggregate routers into regions, autonomous systems (AS)

- Routers in same AS run same routing protocol
  - intra-AS routing protocol
  - Routers in different ASes can run different intra-AS routing protocols

Gateway router
- Has direct link to a router in another AS

Inter-AS routing

- Suppose router in AS1 receives datagram destined outside of AS1:
  - Router should forward packet to a neighbor AS, but which one?

AS1 must:
1. Learn which dests are reachable through AS2, which through AS3
2. Propagate this reachability info to all routers in AS1

Job of inter-AS routing is performed by border gateway routers
Interconnected ASes

- forwarding table configured by both intra- and inter-AS routing protocols
  - intra-AS protocol sets entries for internal dests
  - inter-AS & intra-AS protocols set entries for external dests

Example: Setting forwarding table in router 1d

- AS1 learns from inter-AS protocol that subnet x is reachable via AS3 (gateway 1c) but not via AS2.
- inter-AS protocol propagates reachability info to all internal routers.
- For subnet x, router 1d determines from intra-AS routing info that its interface I is on the least cost path to 1c
  - installs forwarding table entry (x,I)

Note: both inter-AS and intra-AS are used
Example: Choosing among multiple ASes

- now suppose AS1 learns from inter-AS protocol that subnet \( x \) is reachable from AS3 and from AS2.
- to configure forwarding table, router 1d needs to know which gateway it should forward packets for dest \( x \)
- this is also a task of inter-AS routing protocol

Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing
**Intra-AS Routing**

- also known as Interior Gateway Protocols (IGP)
- most common Intra-AS routing protocols:
  - **RIP**: Routing Information Protocol
  - **OSPF**: Open Shortest Path First
  - **EIGRP** (Cisco) - distance vector with “loop-freedom”

---

**RIP (Routing Information Protocol)**

- distance vector algorithm
- included in BSD-UNIX Distribution in 1982
- distance metric: # of hops (max = 15 hops)

<table>
<thead>
<tr>
<th>Destination</th>
<th>Hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>1</td>
</tr>
<tr>
<td>v</td>
<td>2</td>
</tr>
<tr>
<td>w</td>
<td>2</td>
</tr>
<tr>
<td>x</td>
<td>3</td>
</tr>
<tr>
<td>y</td>
<td>3</td>
</tr>
<tr>
<td>z</td>
<td>2</td>
</tr>
</tbody>
</table>

From router A to subnets:

---

3/21/2016

Network Layer (SSL) 4-79
RIP advertisements

- **Distance vectors**: exchanged with neighbors every 30 sec via Response Message (also called advertisement)

- Each advertisement: list of up to 25 destination subnets within AS

RIP Table processing

- RIP routing tables managed by **application-level** process called **routed** (daemon)

- Advertisements sent in **UDP packets**, periodically repeated

![Diagram of network layer connections](image)
OSPF (Open Shortest Path First)

- "open": publicly available
- uses Link State algorithm
- OSPF advertisement carries one entry per neighbor router
- advertisements disseminated to entire AS (via flooding)
  - carried in OSPF messages directly over IP (rather than TCP or UDP)
- security: all OSPF messages authenticated
- ...

Note: IS-IS routing protocol: nearly identical to OSPF

Hierarchical OSPF

3/21/2016
Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Internet inter-AS routing: BGP

- BGP (Border Gateway Protocol): the de facto standard
- an AS advertises its existence to rest of Internet: “I am here” and
  1. obtains reachability information from neighboring ASes (using eBGP)
  2. propagates reachability information to all internal routers of the AS (using iBGP)
  3. determines “good” routes to other ASes based on reachability information and policy.
BGP basics

- **eBGP session**: two BGP routers ("peers") exchange messages over semi-permanent TCP connections
  - advertising paths to various destination network prefixes ("path vector" protocol)
- When AS3 advertises a prefix to AS1, AS3 promises it will forward datagrams towards that prefix
  - AS3 can aggregate prefixes in its advertisement

BGP basics: distributing path information

- Using eBGP session between 3a and 1c, AS3 sends prefix reachability info to AS1.
  - 1c can then use iBGP to distribute new prefix info to all routers in AS1
- When a router learns of a new prefix, it creates entry for prefix in its forwarding table.
  - In this example, router 1b can then re-advertise new reachability info to AS2 over 1b-to-2a eBGP session
Path attributes & BGP routes

- advertised prefix includes BGP attributes.
  - prefix + attributes = "route"
- two important attributes:
  - AS-PATH: contains ASes through which prefix advertisement has passed; e.g., AS 67, AS 17
  - NEXT-HOP: the router interface (IP address) that begins the AS path
    - there may be multiple links from current AS to next-hop-AS
- when a gateway router receives route advertisement, it uses the AS’s import policy to accept or decline.

BGP route selection

- router may learn more than 1 route to some prefix. Router must select one route
- Criteria
  1. local preference value attribute (policy decision)
     - Import rule: customer routes are preferred over peer routes, which are preferred over provider routes
  2. shortest AS-PATH
  ... 
  6. closest NEXT-HOP router: hot potato routing
     - (additional criteria) ...

3/21/2016
BGP routing policy example

- A, B, C are provider networks
- X, W, Y are customers (of provider networks)
- X is dual-homed: attached to two networks
  - X does not want to route from B via itself to C
  - So X will not advertise to B that it has a route to C

3/21/2016

BGP routing policy example (2)

- A advertises path AW to B
- B advertises path BAW to X
- Should B advertise path BAW to C?
  - No - B gets no "revenue" for routing CBAW since neither C nor A nor W is a customer of B
  - B wants to route only to/from its customers
- Export rule: peer/provider route advertised to customers only; customer route advertised to all neighbor ASes

3/21/2016
Why different Intra- and Inter-AS routing?

Scale:
- hierarchical routing reduces table size, also update traffic

Policy:
- Inter-AS: admin wants control over how its traffic is routed, who routes through its net.
- Intra-AS: single admin, so no policy decisions needed

Performance:
- Intra-AS: can focus on performance
- Inter-AS: policy dominates performance

Chapter 4: Network Layer

- 4.1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What’s inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing