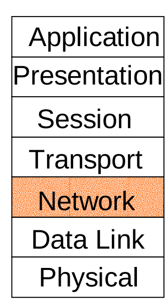
CSE/EE 461
Distance Vector Routing

Last Time

- Introduction to the Network layer
  - Internetworks
  - Datagram and virtual circuit services
  - Internet Protocol (IP) packet format
- The Network layer
  - Provides end-to-end data delivery between networks
  - Issues of scale and heterogeneity

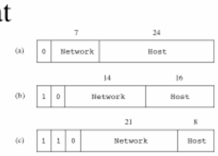


What is an Internet Address?

0–3	4–7	8–15	16–18	19–31
Version	HLen	TOS	Length	
Identifier for Fragments			Flags	Fragment Offset
TTL		Protocol	Checksum	
Source Address				
Destination Address				
Options (variable)				Pad (variable)
Data				

Global Addresses

- Properties
  - globally unique
  - hierarchical: network + host
- Format



- Dot notation
  - 10.3.2.4
  - 128.96.33.81
  - 192.12.69.77

1. Small number of large networks
2. Modest # of medium sized networks
3. Many small networks

CLASS	*SIZE*	*NUMBER*
A	2G	126
B		
C	254	2M

Original Rationale: Beware the Routing Tables
1. You don't care about most networks.
2. The few networks you do care about, you care about them a lot.
3. Not many routing table entries get you "closer" to a lot of the hosts

Datagram Forwarding

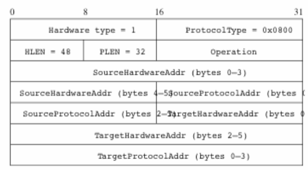
- Strategy
  - every datagram contains destination's address
  - if directly connected to destination network, then forward to host
  - if not directly connected to destination network, then forward to some router
  - forwarding table maps network number into next hop
  - each host has a default router
  - each router maintains a forwarding table
- Example (router R2)

Network Number	Next Hop
1	R3
2	R1
3	interface 1
4	interface 0

Address Translation

- Map IP addresses into physical addresses
  - destination host
  - next hop router
- Techniques
  - encode physical address in host part of IP address
  - table-based
- ARP
  - table of IP to physical address bindings
  - broadcast request if IP address not in table
  - target machine responds with its physical address
  - table entries are discarded if not refreshed

ARP Packets

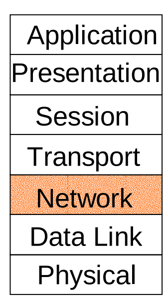


- HardwareType: type of physical network (e.g., Ethernet)
- ProtocolType: type of higher layer protocol (e.g., IP)
- HLEN & PLEN: length of physical and protocol addresses
- Operation: request or response
- Source/Target Physical/Protocol addresses

- Notes
  - table entries timeout in about 10 minutes
  - update table with source when you are the target
  - update table table if already have an entry
  - do not refresh table entries upon reference

This Time

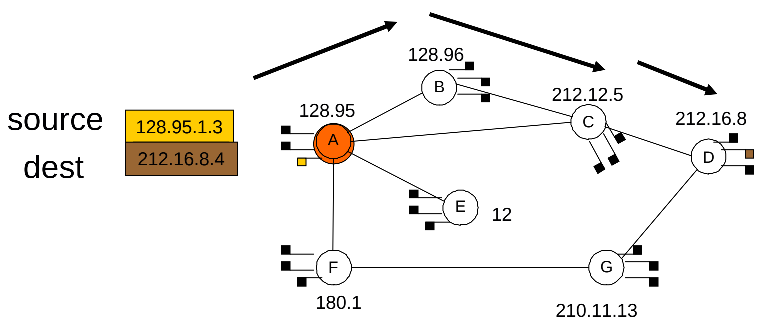
- Focus
  - How do we calculate routes for packets?
  - Routing is a network layer function
- Routing Algorithms
  - Distance Vector routing (RIP)



Forwarding and Routing

- Forwarding is the process that each router goes through for every packet to send it on its way
  - Involves local decisions
  - "For this packet, which link should I sent it out?"
- Routing is the process that all routers go through to calculate the routing tables
  - Involves global decisions
  - "For this link, which networks can I get to?"

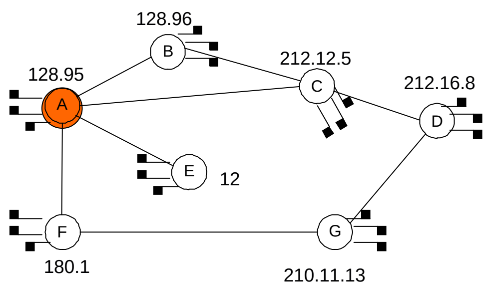
Forwarding



Routing Table Lets You Forward

- The routing table at A, for example, lists at a minimum the next hops for the different destinations

Dest	Next Hop
B	B
C	C
D	C
E	E
F	E
G	F



Kinds of Routing Schemes

- Many routing schemes have been proposed/explored!

- Distributed or centralized
- Hop-by-hop or source-based
- Deterministic or stochastic
- Single or multi-path
- Static or dynamic route selection

- The Internet is more left than right.

Routing Questions

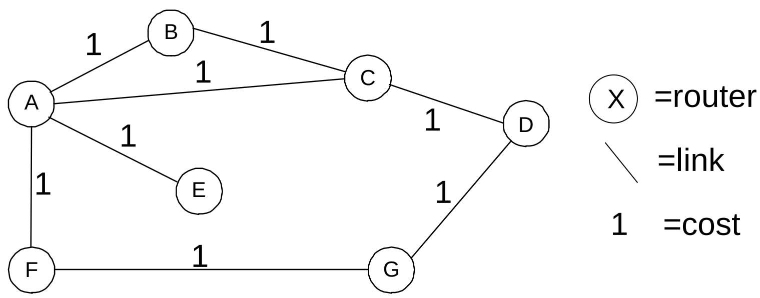
- How to choose best path?
  - Defining "best" is slippery
- How to scale to millions of users?
  - Minimize # control messages and routing table size per router
- How to adapt to failures or changes?
  - Node and link failures, plus message loss
  - We will use distributed algorithms

Some Pitfalls

- Using global knowledge is challenging
  - Hard to collect
  - Can be out-of-date
  - Needs to summarize in a locally-relevant way
- Inconsistencies in local/global knowledge can cause
  - Loops
    - black holes
  - Oscillations, esp. when adapting to load

Network as a Graph

- Routing is essentially a problem in graph theory



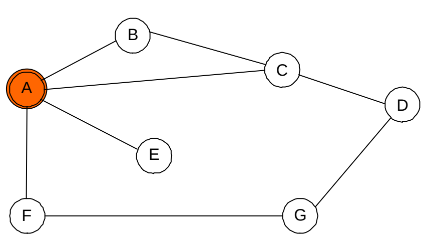
Distance Vector Routing

- Assume:
  - Each router knows only address/cost of neighbors
- Goal:
  - Calculate routing table of next hop information for each destination at each router
- Idea:
  - Tell neighbors about learned distances to all destinations

DV Algorithm

- Each router maintains a vector of costs to all destinations as well as routing table
  - Initialize neighbors with known cost, others with infinity
- Periodically send copy of distance vector to neighbors
  - On reception of a vector, if neighbors path to a destination plus neighbor cost is better, then switch to better path
    - update cost in vector and next hop in routing table
- Assuming no changes, will converge to shortest paths
  - But what happens if there are changes?

DV Example – Initial Table at A

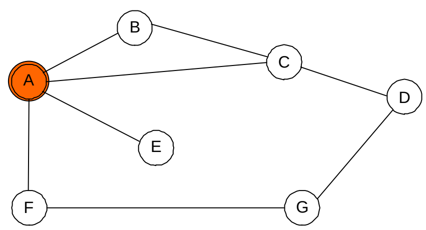


Dest	Cost	Next
B	1	B
C	1	C
D	∞	-
E	1	E
F	1	F
G	∞	-

The Distance Vector

DV Example – Final Table at A

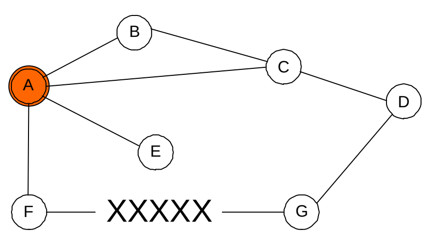
- Reached in a single iteration … simple example



Dest	Cost	Next
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

Adapting to Change

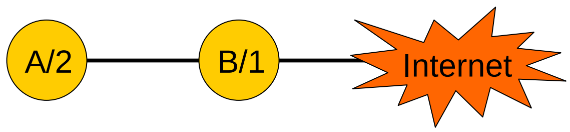
- One scenario: Suppose link between F and G fails
  1. F notices failure, sets its cost to G to infinity and tells A
  2. A sets its cost to G to infinity too, since it learned it from F
  3. A learns route from C with cost 2 and adopts it



Dest	Cost	Next
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	3	C

Count To Infinity Problem

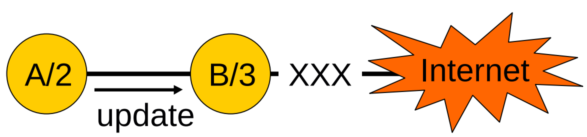
- Simple example
  - Costs in nodes are to reach Internet (really, “get to another routing domain”)



- Now link between B and Internet fails …

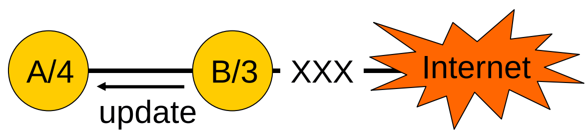
Count To Infinity Problem

- B hears of a route to the Internet via A with cost 2
- So B switches to the “better” (but wrong!) route



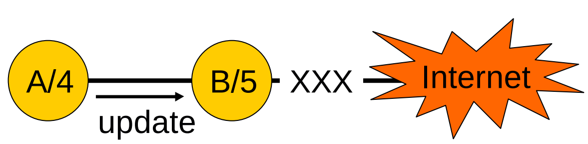
Count To Infinity Problem

- A hears from B and increases its cost



Count To Infinity Problem

- B hears from A and (surprise) increases its cost
- Cycle continues and we "count to infinity"



- Packets caught in the crossfire loop between A and B
  - Data and routing packets get lost

Split Horizon

- Solves trivial count-to-infinity problem
- Router never advertises the cost of a destination back to to its next hop – that's where it learned it from!
- Poison reverse: go even further – advertise back infinity to keep the receiver from routing "right" when they should go "left"
  - In other words, A's DV message to X sets distance(Y)=infinity if A's next hop for Y is X.
- However, DV protocols still subject to the same problem with more complicated topologies (eg, 3 way loops with lost messages)
  - Many enhancements suggested

Routing Information Protocol (RIP)

- DV protocol with hop count as metric
  - Infinity value is 16 hops; limits network size
  - Includes split horizon with poison reverse
- Routers send vectors every 30 seconds
  - With triggered updates for link failures
  - Time-out in 180 seconds to detect failures
- RIPv1 specified in RFC1058
  - www.ietf.org/rfc/rfc1058.txt
- RIPv2 (adds authentication etc.) in RFC1388
  - www.ietf.org/rfc/rfc1388.txt

Key Concepts

- Routing is a global process, forwarding is local one
- The Distance Vector algorithm and RIP
  - Simple and distributed exchange of shortest paths.
  - Weak at adapting to changes (loops, count to infinity)