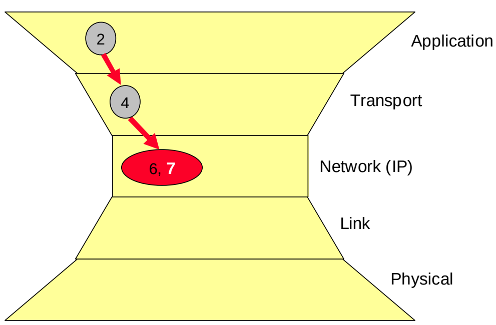
EECS 122:
Introduction to Computer Networks
Interdomain Routing

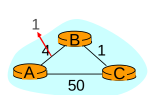
Computer Science Division
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
Berkeley, CA 94720-1776

Today's Lecture



Distance Vector: Link Cost Changes

```
loop:
 wait (until A sees a link cost change to neighbor V
       or until A receives update from neighbor V)
 if (D(A,V) changes by d)
   for all destinations Y through V do
     D(A,Y) =  D(A,Y) + d
 else if (update D(V, Y) received from V)
   D(A,Y) = D(A,V) + D(V, Y);
 if (there is a new minimum for destination Y)
   send D(A, Y) to all neighbors
forever
```



Node	D	C	N	D	C	N	D	C	N	D	C	N
Node B	A	4	A	A	1	A	A	1	A	A	1	A
	C	1	C	C	1	C	C	1	C	C	1	C
Node C	A	5	B	A	5	B	A	2	B	A	2	B
	B	1	B	B	1	B	B	1	B	B	1	B

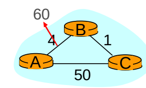
"good news travels fast"

time

Link cost changes here **Algorithm terminates**

Distance Vector: Count to Infinity Problem

```
loop:
 wait (until A sees a link cost change to neighbor V
       or until A receives update from neighbor V)
 if (D(A,V) changes by d)
   for all destinations Y through V do
     D(A,Y) =  D(A,Y) + d ;
 else if (update D(V, Y) received from V)
   D(A,Y) = D(A,V) + D(V, Y);
 if (there is a new minimum for destination Y)
   send D(A, Y) to all neighbors
forever
```



Node	D	C	N	D	C	N	D	C	N	D	C	N
Node B	A	4	A	A	6	C	A	6	C	A	8	C
	C	1	C	C	1	C	C	1	C	C	1	C
Node C	A	5	B	A	5	B	A	7	B	A	7	B
	B	1	B	B	1	B	B	1	B	B	1	B

…

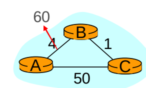
"bad news travels slowly"

time

Link cost changes here; recall from slide 24 that B also maintains shortest distance to A through C, which is 6. Thus D(B, A) becomes 6 !

Distance Vector: Poisoned Reverse

- If C routes through B to get to A:
  - C tells B its (C's) distance to A is infinite (so B won't route to A via C)
  - Will this completely solve count to infinity problem?



Node	D	C	N	D	C	N	D	C	N	D	C	N	D	C	N
Node B	A	4	A	A	60	A	A	60	A	A	51	C	A	51	C
	C	1	B	C	1	B	C	1	B	C	1	B	C	1	B
Node C	A	5	B	A	5	B	A	50	A	A	50	A	A	50	A
	B	1	B	B	1	B	B	1	B	B	1	B	B	1	B

time

Link cost changes here; B updates D(B, A) = 60 as C has advertised D(C, A) = ∞

Algorithm terminates

Link State vs. Distance Vector

Per-node message complexity
- LS: O(e) messages
  - e: number of edges
- DV: O(d) messages, many times
  - d is node's degree

Complexity/Convergence
- LS: $O(n^2)$ computation
- DV: convergence time varies
  - may be routing loops
  - count-to-infinity problem

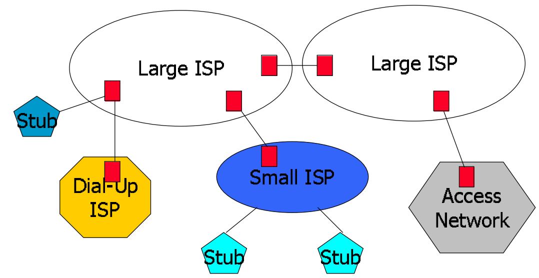
Robustness: what happens if router malfunctions?
- LS:
  - node can advertise incorrect *link* cost
  - each node computes only its *own* table
- DV:
  - node can advertise incorrect *path* cost
  - each node's table used by others; error propagate through network

Are We Done?

- We now know how to route scalably
- What more is there to do?

EECS F05 7

Network Structure



The Internet contains a large number of diverse networks

EECS F05 10

Issues We Haven't Addressed

- Scaling
  - Router table size
- Structure
  - Autonomy
  - Policy

EECS F05 8

Autonomous Systems (AS)

- Internet is not a single network!
- The Internet is a collection of networks, each controlled by different administrations
- An autonomous system (AS) is a network under a single administrative control

EECS F05 11

Scaling

- Every router must be able to forward based on *any* destination IP address
  - Given address, it needs to know "next hop" (table)
- Naive: Have an entry for each address
  - There would be 10^8 entries!
- Better: Have an entry for a range of addresses
  - But can't do this if addresses are assigned randomly!
- Addresses allocation is a big deal

EECS F05 9

Implications

- ASs want to choose own local routing algorithm
  - Intra-domain routing algorithm, e.g., link state (OSPF), distance vector
- ASs want to choose own nonlocal routing policy
  - Inter-domain routing: BGP de facto standard

EECS F05 12

Interconnection

- IP unifies network technologies
  - Allows any network to communicate with another
- BGP unifies network organizations
  - Ties them into a global Internet

Outline

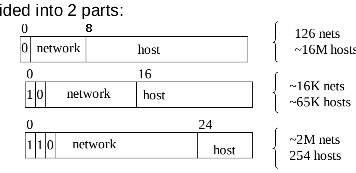
- Addressing
- BGP

Assigning Addresses (Ideally)

- Host: gets IP address from its organization or ISP
- Organization: gets IP address block from ISP
- ISP: gets address block from routing registry:
  - ARIN: American Registry for Internet Numbers
  - RIPE: Reseaux IP Europeens
  - APNIC: Asia Pacific Network Information Center
- Each AS is assigned a 16-bit number (65536 total)
  - Currently 10,000 AS's

Original Addressing Scheme

- Class-based addressing schemes:
  - 32 bits divided into 2 parts:
  - Class A
  - Class B
  - Class C



Original Vision:

- Route on network number
- All nodes with same net # are directly connected

Classless Interdomain Routing (CIDR)

Introduced to solve two problems:

- Exhaustion of IP address space
- Size and growth rate of routing table

#1: Address Space Exhaustion

- Example: an organization needs 500 addresses.
  - A single class C address not enough (254 hosts).
  - Instead a class B address is allocated. (~65K hosts)
  - That's overkill, a huge waste!
- CIDR: networks assigned on arbitrary bit boundaries.
  - Requires explicit masks to be passed in routing protocols
  - Masks: identify the "network" portion of the address
- CIDR solution for example above: organization is allocated a single /23 address (equivalent of 2 class C's).

CIDR Addressing

- Suppose fifty computers in a network are assigned IP addresses 128.23.9.0 - 128.23.9.49
  - They share the **prefix** 128.23.9
- Range: 01111111 00001111 00001001 00000000 to
  01111111 00001111 00001001 00110001
  - How to write 01111111 00001111 00001001 00XX XXXX ?
- Convention: 128.23.9.0/26
  - There are 32-26=6 bits for the 50 computers
  - 2^6 = 64 addresses
- Maximal waste: 50%

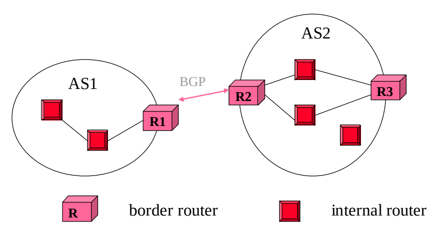
Border Gateway Protocol

ignore the details
pay attention to the "why"

More Formally

- Specify a range of addresses by a prefix: X/Y
  - The common prefix is the first Y bits of X.
  - X: The first address in the range has prefix X
  - Y: 2^{32-Y} addresses in the range
- Example 128.5.10/23
  - Common prefix is 23 bits:
  - 01000000 00000101 0000101
  - Number of addresses: 2^9 = 512
- Prefix aggregation
  - Combine two address ranges
  - 128.5.10/24 and 128.5.11/24 gives 128.5.10/23
- Routers match to longest prefix

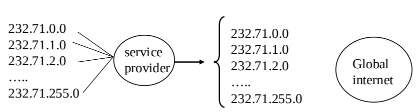
Who speaks BGP?



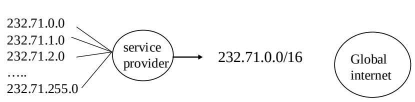
- Two types of routers
  - Border router (Edge), Internal router (Core)

Problem #2: Routing Table Size

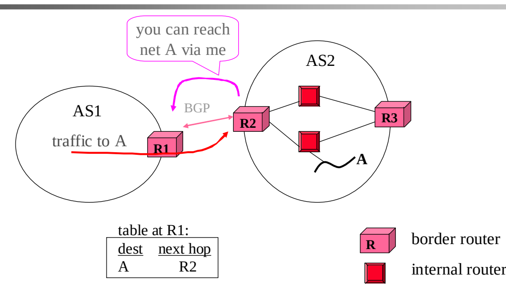
Without CIDR:



With CIDR:

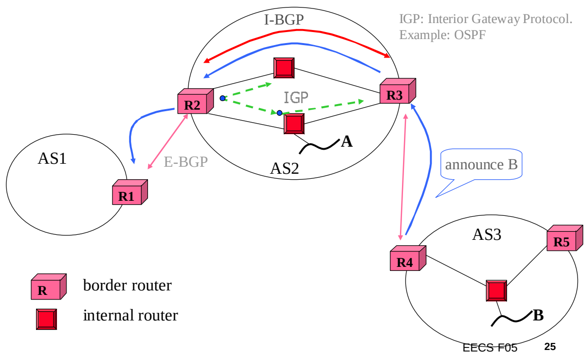


Purpose of BGP



Share connectivity information across ASes

I-BGP and E-BGP



Issues

- What basic routing algorithm should BGP use?
- How are the routes advertised?
- How are routing policies implemented?
  - Policy routing: not always shortest path

Choice of Routing Algorithm

- Constraints:
  - Scaling
  - Autonomy (policy and privacy)
- Link-state?
  - Requires sharing of complete network informatin
  - Information exchanges doesn't scale
  - All policies exposed
- Distance Vector?
  - Scales and retains privacy
  - Can't implement policy
  - Can't avoid loops if shortest paths not taken

Path Vector Protocol

- Distance vector algorithm with extra information
  - For each route, store the complete path (ASs)
  - No extra computation, just extra storage
- Advantages:
  - Can make policy choices based on set of ASs in path
  - Can easily avoid loops

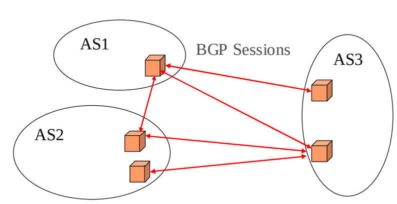
BGP Routing Table

```
ner-routes>show ip bgp
BGP table version is 6128791, local router ID is 4.2.34.165
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
* i3.0.0.0          4.0.6.142             1000     50      0 701 80 i
* i4.0.0.0          4.24.1.35                0    100      0 i
* i12.3.21.0/23     192.205.32.153           0     50      0 7018 4264 6468 ?
* e128.32.0.0/16    192.205.32.153           0     50      0 7018 4264 6468 25 e
```

- Every route advertisement contains the entire AS path
- Can implement policies for choosing best route
- Can detect loops at an AS level

Advertising Routes

- One router can participate in many BGP sessions.
- *Initially* … node advertises ALL routes it wants neighbor to know (could be > 50K routes)
- *Ongoing* … only inform neighbor of changes



Basic Messages in BGP

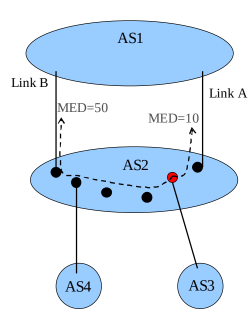
- *Open*:
  - Establishes BGP session (uses TCP port #179)
  - BGP uses TCP
- *Notification*:
  - Report unusual conditions
- *Update*:
  - Inform neighbor of new routes that become active
  - Inform neighbor of old routes that become inactive
- *Keepalive*:
  - Inform neighbor that connection is still viable

Routes Have Attributes

- When a route is “advertised” it is described in terms of attributes:
  - next hop, AS-path, etc.
  - We will discuss: Origin, MED, Local Preference
- Origin:
  - Who originated the announcement? Where was a prefix *injected* into BGP?
  - IGP, EGP or Incomplete (often used for static routes)

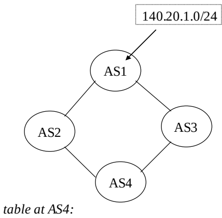
Multi-Exit Discriminator (MED)

- When AS’s interconnected via 2 or more links
- AS announcing prefix sets MED (AS2 in picture)
- AS receiving prefix uses MED to select link
- A way to specify how close a prefix is to the link it is announced on



Local Preference

- Used to indicate preference among multiple paths for the same prefix *anywhere* in the Internet.
- The higher the value the more preferred
- Exchanged between IBGP peers only. Local to the AS.
- Often used to select a specific exit point for a particular destination



BGP table at AS4:

Destination	AS Path	Local Pref
140.20.1.0/24	**AS3 AS1**	**300**
140.20.1.0/24	**AS2 AS1**	**100**

Choosing Best Route

- Choose route with highest **LOCAL_PREF**
  - Preference-based routing
- Multiple choices: select route with shortest **hop-count**
- Multiple choices for same neighboring AS: choose path with min MED value
- Choose route based on lowest origin type
  - IGP < EGP < INCOMPLETE
- Among IGP paths, choose one with lowest cost
- Finally use router ID to break the tie.

Is Reachability Guaranteed?

- In normal routing, if graph is connected then reachability is assured
- With policy routing, not always

BGP and Performance

- BGP designed for policy not performance
  - Hot Potato routing common but suboptimal
  - 20% of internet paths inflated by at least 5 router hops
- Susceptible to router misconfiguration
  - Blackholes: announce a route you cannot reach
- Incompatible policies
  - Solutions to limit the set of allowable policies