Computer Networks
ICS 651

- distance-vector routing
- RIP
- BGP
- link-state routing
- OSPF
- Internet Control Message Protocol, ICMP

Challenge: Build Routing Tables Automatically

- automatically construct the routing tables for each router
- each router is configured (manually) with the IP address for each interface
- each router can send a message to the other endpoint of a link, and listen for replies, to find out who it is connected to
- on a broadcast network a router can broadcast or multicast a message, and all other routers on the network will reply
- each neighboring router is connected via a link, which may be shared with other routers (in case of a broadcast network) or may be dedicated (point-to-point links)
- given that each router has information about its own links to neighboring routers, how does this information get to all the other routers in the network?

Distributed Routing Algorithms

- Distance Vector:
  - I know how to reach my neighbors
  - I tell my neighbors they can reach my neighbors through me
  - I tell my neighbors they can reach my neighbors' neighbors through me
  - recurse until everyone is reachable
- Link State:
  - distribute each router's link state to all routers
  - each router independently builds a map of the entire network, and uses it for routing

Distance Vector Algorithm -- Generating Information

- routing table has:
  - destination (perhaps with address mask)
  - interface
  - metric/distance (in hops)
  - next hop (IP address)
- I send my entire table (destinations, masks and distances -- no need to send the next hop or the interface) to each neighbor, both periodically, and whenever it changes
- the message that is sent has, for each entry:
  - destination, with address mask
  - metric/distance (in hops)

Distance Vector Algorithm -- Processing Information

- when receiving a routing information message from router R on interface IF, look at each entry (IP/mask, d):
  - set d' = d + 1
  - if IP/mask is not in the table, add (IP/mask, IF, d', R) to the routing table
  - if IP/mask already is in the table with interface IF'' and distance d'', then
    - if either d'' > d' or IF'' = IF, then replace the routing table entry with (IP/mask, IF, d', R)
    - otherwise, ignore this entry
- In-class discussion: what happens if the new IP matches an existing IP but with a different mask?

Distance Vector Example

- My routing table:

Destination	Distance	Port	Gateway (next hop)
A	4	eth0	Q
B	2	tty0	R
X	5	eth1	S

- message from neighbor R on port tty0: (A, 2), (B, 3), (X, 5)
- New routing table:

Destination	Distance	Port	Gateway (next hop)
A	3	tty0	R
B	4	tty0	R
X	5	eth1	S

Issues with Distance Vector

- My routing table:

Destination	Distance	Port	Gateway (next hop)
A	3	tty0	R
B	4	tty0	R
X	5	eth1	S

- suppose the link to router R goes down
- the routes to A and B are unusable and can be deleted
- neighbor S advertises routes to A and B with a cost of 4 and 5, so those are added to the routing table
- unfortunately, neighbor S was simply sending back the routes it heard from this router
- there may be a higher-cost route to A or B, but this will be found only slowly (counting-to-infinity)

Resolving the issue of routers sending back routes no longer valid

- have a small value of infinity (16 in RIP)
- resend tables whenever they change, to get faster counting-to-infinity
- do not send to neighbor N routes that have N as the next hop (split horizon)
- or, send those routes, but with infinite metric (split horizon with poisoned reverse)

- counting to infinity can still happen if more than two routers are richly connected

More Issues with Distance Vector

- There is no way in the algorithm to delete routes
- Solution: routes time out when they are not refreshed within a certain time
- Problem: unless all routers time out simultaneously, that route may still be alive in a neighboring router, which may advertise it back to us (unless we use split horizon)
- this again leads to counting to infinity, but very slowly!

Routing Information Protocol

- DV modified with split horizon and poisoned reverse
- distance vector algorithm, using hop count as metric
- Intradomain routing only (e.g. within hawaii.edu)
- RIP v2 allows specification of netmask
- RIPng (RFC 2080) supports IPv6
- routers exchange messages every 30 seconds, and also when their routing tables change

Border Gateway Protocol

- current version is BGP-4
  - since 2006
  - RFC 4271
- Internet is an arbitrarily connected set of autonomous systems
- a Stub AS has single connection to Internet
- a Multihomed AS has multiple connections but refuses to carry transit traffic
- a Transit AS will carry transit traffic
- try to find a route, don't try too hard to find the optimal route
- one router in AS is the BGP speaker, advertises reachable networks
- explicit cancellation: withdrawn routes

- not DV or LS: instead, the protocol distributes paths to destinations, which makes it a kind of “path vector” protocol

BGP Policy

- RIP is very easy to configure, because there are few options
- OSPF (discussed under link-state) usually works on networks subdivided into areas, so some configuration is necessary
- BGP allows the network administrator to specify which transit AS is preferred over others, so can require a lot of configuration
- policy preferences allow selecting or preferring paths that do not cross expensive links and have the least security concerns
- Autonomous Systems truly are autonomous, so each administrator selects the policies that are best for this AS

Link-State Routing Algorithm

- each node keeps a table of its neighbors, with destination and metric: the **link-state table**
- each node **broadcasts** its link-state to all other routers in the network
- the link state is broadcast whenever it changes, and also from time to time
- after receiving a broadcast from all other routers, each node can figure out the network topology
- the node uses this private **map** to route packets
  - e.g. use Dijkstra's shortest path algorithm to build its routing table
- every node must use an equivalent algorithm, or the source must determine the route
  - e.g. source routing can be used

Comparison of Link-State and Distance-Vector

- LS must exchange more information overall, since each router gets all information about the network
- in LS, each host must compute a route based on the resulting graph
- with DV, computation of the route is automatic, and only the necessary information is exchanged
- LS requires some notion of the "age" of information, so subsequent broadcasts override earlier broadcasts
- a reliable DV also needs a way to make information obsolete

Broadcast: a Flooding Algorithm

1. keep track of all packets received
2. on receipt of a packet from interface i:
   - if it has been received before (see step 1), discard it.
   - otherwise, forward it on all interfaces except i
3. this scheme is the one used by hubs in a 10-Base T ethernet4. what problems does this scheme have?

Broadcast: an improved Flooding Algorithm

1.keep track of all packets received

2.on receipt of a packet from interface i:

- if it has been received before (see step 1), discard it.
- otherwise, forward it on all interfaces except i

3.this scheme is the one used by hubs in a 10-Base T ethernet4. what problems does this scheme have?

improvement:

- each sender numbers all broadcast packets sequentially
- each node only needs to remember the most recent sequence number from each sender
- have to be careful about sequence number wrap-around

Other Routing Protocols

- MPLS, Multi-Protocol Label Switching, adds to each packet a small header which includes an integer label.
  - routers in a transit AS can be configured to recognize this label and forward the packet, unmodified, to a specific interface and next hop.
  - this Label Switching can be implemented in hardware.
  - label switching must be set up by software that computes a path through the network for each pair of source (ingress) and destination (egress) routers, assigns a different label to each path, and distributes to each router the labels and the relevant details of each path.
  - MPLS can support any protocol, not just IP
- IS-IS is a link-state protocol, typically working at the layer below IP, and so, like MPLS, more protocol-agnostic than OSPF or RIP
- EIGRP is a Distance-Vector protocol, but sends updates instead of the entire routing table. EIGRP was proprietary to Cisco until 2013, when (most of) the protocol was published as RFC 7868

Software Defined Networking

- routing is an inefficient peer-to-peer system
- can we centralize it?
- challenge: the central controller uses the network to reach the router it wants to configure
- advantages: less overhead if only sending the data that must be sent, faster response to changes, consistent configuration, easier management

Summary: Routing Protocols

- Interior Gateway Protocols, or IGPs, route packets within an autonomous system
- Exterior Gateway Protocols, or EGPs, route packets between autonomous systems
- there is only one common EGP, which is BGP. All the other routing protocols we have discussed are IGPs
- in general, each autonomous system will be running just one IGP, and may have one or more BGP routers that are used as gateways to other autonomous systems

Summary: IP Routing

- summarize, summarize, summarize
- use hierarchy whenever possible:
- route to networks, not hosts
- OSPF: route to areas, not networks
- BGP: route to autonomous systems
- routing protocols can evolve faster than the underlying IP transport
- Interior Gateway Protocols (IGP) are the most automated: RIP for small networks, OSPF and IS-IS for larger ones

Internet Control Message Protocol

- ICMP
- the Internet is complex
- how do we find out what is going wrong?
- send a packet "there and back": ICMP echo, ping
- send an ICMP error packet whenever we drop a (non ICMP error) packet
- ICMP: RFC 792

ICMP Echo

- Echo packet (type 8) or Echo Reply (type 0)
- checksum covers entire packet
- identifier (typically process ID on sender machine)
- sequence number (typically 1, 2, 3...)
- arbitrary data follows (could be large)
- for ping, data typically holds binary date and time (8 bytes or 12 bytes)

Other ICMP Types

[3] Destination Unreachable (network, host, protocol, prohibited...)

[11] Time Exceeded (in transit, during reassembly)

[5] Redirect: use this other router for this destination

[9] Router advertisement

[10] Router solicitation

[4] Source Quench

[12] Parameter Problem

IP Path MTU Discovery

- send IPv6 packets, or IPv4 packets with DF (Don’t Fragment) set
- cannot send more than local network MTU
- if a router must drop a packet that exceeds the MTU of the outgoing interface, it can send a "destination unreachable/fragmentation needed" ICMP message, or an ICMPv6 “packet too big” message
- this ICMP message carries the MTU
- if there is no ICMP message, sender can do a binary search (on common MTU sizes) to find an MTU that works
- however, the path MTU can change!
- since ICMP message may be dropped, also needs other ways to detect dropped packets
- slow, time-consuming, error-prone...