ICS 351: Today's plan

- network design exercise
- RIP
- distance-vector routing

Network Design Exercise

- three routers, A, B, and C, each connected to the other two
- router A is also connected to the wider internet
- router A connects to three other subnetworks:
- o X, with at least 10 static addresses and 20 dynamic addresses,
- o Y, with at least 50 dynamic addresses,
- o and Z, with at least 2 static addresses.
- router B connects to two other subnetworks:
- o H, with at least 5 static addresses but a good chance of future growth, and
- o K, with at least 30 dynamic addresses and 30 static addresses
- router C connects to two other subnetworks:
- o P, with at least 5 static addresses
- o Q, with at least 5 dynamic addresses
- You have been given the IP address range 10.11.12.0/24
- assign IP addresses for this network

RIPv2

- Routing Information Protocol, version 2 (v2 supports network masks)
- RIP generally used within a single Autonomous System (AS), i.e. within an organization
- this makes it an IGP, Interior Gateway Protocol
- reliably finds shortest paths in networks that don't change very often
- links may have a "metric" (cost, distance) associated with them, or "1" may be used for each link, but in any case metrics should be static
- limited to networks with a maximum metric of 15 between any two nodes
- router must be configured to know which interfaces to run RIP on
- defined in RFC 2453, http://tools.ietf.org/html/rfc2453

distance-vector routing

- Bellman-Ford, applying Bellman's equation in the Ford and Fulkerson algorithm
- the basic idea is that each router sends its routing table to all other routers on the same network
- when receiving such a message (routing update), a router must update its own routing table:

distance-vector algorithm

- when receiving a routing update, a router must update its own routing table:
- o any new route is simply added, with the next hop (gateway) being the router that sent the routing update, and the metric being the metric received plus the metric of the link over which it was received
- o any existing route is replaced if the new route has better metric (after adding the link metric)
- o any existing route is replaced if the routing update comes from the router listed as next hop
- routes through a given next hop G are *timed out* if no routing updates are received from G within a certain period (about 6 times longer than the 30s routing update time)
- if a route is deleted, it is actually marked as having an infinite cost (for RIP, infinity is 16)
- also, when sending a routing update to neighbor G, routes through G are "poisoned" by giving them infinite metric
- This last is **split horizon with poisoned reverse**

RIPv2 details

- routing tables sent to multicast address
- split horizon is required, poisoned reverse is optional
- whenever the routing table changes (e.g., as a result of an interface changing, or of receiving a routing update), a triggered update is sent out, perhaps with only the new route(s)
- triggered updates should be limited to about 1 every 5 seconds
- regular updates are sent every 30 seconds

counting to infinity

- suppose that a router R loses one of its connections (to network N with metric m), but does not send a triggered update, and does not do split horizon
- then, its neighbor G will tell it about a route to N with metric m+2
- R will use a route to N with metric m+3, next hop G
- at the next update, G will know R has a route to N with metric m+3, so will have a route with metric m+4, next hop R
- then R updates its route to N to have metric m+5
- then G updates its route to N to have metric m+6
- and so on
- this only stops when one of the routers reaches 16 (infinity)
- it could potentially take a long time to delete bad routes