ICS 351: Today's plan

- distance-vector routing game
- link-state routing
- OSPF

distance-vector routing game

- 1. prepare a list of all neighbors and the links to them, and the metric for each link

- 2. create your initial routing table with the directly connected networks

- 3. distribute a copy to every neighbor

- 4. add or remove incoming routes, timestamp

- 5. return to step 3 whenever the table changes

- 6. forward any packets you receive

link-state routing

- in distance-vector (Bellman-Ford, Ford-Fulkerson, RIP-style) routing, each router distributes its routing table to its neighbors

- an alternative is for each router to broadcast its neighborhood connectivity to all routers

- once a router has heard from every other router, it can puzzle the network together from the pieces

- once a router has a network map, it can find routes in the map (using a shortest path graph algorithm), and build its routing table accordingly

- in practice, this process runs incrementally

- the neighborhood connectivity describes the state (up or down, and perhaps bandwidth and load) of all the links to which a router is connected, so this algorithm is called Link-State

- in wireless ad-hoc networks, OLSR stands for Optimized Link-State Routing protocol

- in wired IP networks, OSPF stands for Open Shortest Path First

- IS-IS (Intermediate System to Intermediate System routing) also uses link-state routing

link-state routing game

- 1. prepare a list of all neighbors and the links to them (HELLO protocol)

- 2. make a copy for every router in the network

- 3. distribute a copy to every router in the network (by sending it to the neighbors, and letting them distribute it)

- 4. build the network map

- 5. find the shortest path to each router

- 6. build the routing table

Flooding Link State Advertisements

- each router is responsible for distributing copies of every link-state advertisement that it gets

- if there are loops, this means each router will get multiple copies of each LSA

- so LSAs are only forwarded if they are new

- LSAs are acknowledged to the sender, and LSAs are resent if there is no acknowledgement, so that the transmission is reliable

OSPF

- OSPF generally used within a single Autonomous System (AS), i.e. within an organization (IGP, Interior Gateway Protocol)

- reliably finds shortest paths quickly

- divides AS into areas, including a backbone area

- all areas are connected to the backbone area

- all routing information is disseminated over the backbone area

- routers in OSPF play different roles, for example a backbone router is connected to the backbone, an area border router is connected to more than one area (and is usually also a backbone router), and an internal router is only connected to routers in the same area

- every area has one Designated Router which receives then rebroadcasts link-state updates

- defined in RFC 2328 (and in RFC 5340 for IPv6)

RIP compared to OSPF

- both RIP and OSPF find optimal paths
- OSPF generally finds them much more quickly
- OSPF can use multiple metrics
- RIP generally sends less data (somewhat lower overhead)
- OSPF is more complex: more configurable, more code

More OSPF details

- each router has a list of all link states

- an algorithm such as Dijkstra's shortest path algorithm can be used to build a directed acyclic graph (DAG) with the router at the root, and all other networks reachable through the DAG

- multiple equal-cost paths can be used for each destination

- OSPF supports authentication among routers (null authentication is an option)

- link-state advertisements expire if they are too old

OSPF areas

- areas can be used in larger networks to minimize the amount of information exchanged among routers

- routers outside an area don't have all the link state information of routers inside the area

- areas form a 2-level hierarchy with the backbone at the root, and all the other areas below it