Lecture 15: Interdomain Routing

CSE 123: Computer Networks
Alex C. Snoeren
Overview

• Mitigating loops in DV
  ◆ Split horizon and poison reverse

• Autonomous Systems
  ◆ Each network on the Internet has its own goals

• Path-vector Routing
  ◆ Allows scalable, informed route selection
Handling Link Failure

- A marks distance to E as $\infty$, and tells B
- E marks distance to A as $\infty$, and tells B and D
- B and D recompute routes and tell C, E, and E
- etc... until converge

CSE 123 – Lecture 15: Interdomain Routing
Problem: Counting to Infinity

Distance to C

CSE 123 – Lecture 15: Interdomain Routing
Why so High?

- Updates don’t contain enough information
  - $B$ accepts $A$’s path to $C$ that is *implicitly* through $B$!

- Can’t totally order “bad news” (a link has gone down) above “good news” (a link is available)

- Aside: this also causes delays in convergence even when it doesn’t count to infinity
Mitigation Strategies

- **Hold downs**
  - As metric increases, delay propagating information
  - Limitation: Delays convergence

- **Loop avoidance**
  - Full path information in route advertisement
  - Explicit queries for loops

- **Split horizon**
  - Never advertise a destination through its next hop
    » A doesn’t advertise C to B
  - Poison reverse: Send negative information when advertising a destination through its next hop
    » A advertises C to B with a metric of $\infty$
    » Limitation: Only works for “loop”s of size 2
If Z routes through Y to get to X:

- Z tells Y its (Z’s) distance to X is infinite (so Y won’t route to X via Z)
Split Horizon Limitations

- A tells B & C that D is unreachable
- B computes new route through C
  - Tells C that D is unreachable (poison reverse)
  - Tells A it has path of cost 3 (split horizon doesn’t apply)
- A computes new route through B
  - A tells C that D is now reachable
- Etc…
RIP: Routing Information Protocol
- 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
- Rarely used today

EIGRP: proprietary Cisco protocol
- Ensures loop-freedom (DUAL algorithm)
- Only communicates changes (no regular broadcast)
- Combine multiple metrics into a single metric (BW, delay, reliability, load)
Distance Vector shortest-path routing
- Each node sends list of its shortest distance to each destination to its neighbors
- Neighbors update their lists; iterate

Weak at adapting to changes out of the box
- Problems include loops and count to infinity
Routing so far...

- Shortest-path routing
  - Metric-based, using link weights
  - Routers share a common view of path “goodness”

- As such, commonly used *inside* an organization
  - EIGRP and OSPF are mostly used as *intradomain* protocols

- But the Internet is a “network of networks”
  - How to stitch the many networks together?
  - When networks may not have common goals
  - … and may not want to share information
The Internet is Complicated

- Inter-domain versus intra-domain routing

You at school —

Large organization

Backbone service provider

“Consumer” ISP

Small corporation

Peering point

You at home

“Consumer” ISP

Large corporation

“Consumer” ISP

Peering point

“Consumer” ISP

CSE 123 – Lecture 15: Interdomain Routing
Original ARPAnet had single routing protocol
- Dynamic DV scheme, replaced with static metric LS algorithm

New networks came on the scene
- NSFnet, CSnet, DDN, etc...
- The total number of nodes was growing exponentially
- With their own routing protocols (RIP, Hello, ISIS)
- And their own rules (e.g. NSF AUP)

New requirements
- Huge scale: millions of routers
- Varying routing metrics
- Need to express business realities (policies)
Shortest Path Doesn’t Work

- All nodes need common notion of link costs
- Incompatible with commercial relationships
● Separate routing inside a domain from routing between domains
  ◆ Inside a domain use traditional interior gateway protocols (RIP, OSPF, etc)
    » You’ve seen these already

  ◆ Between domains use Exterior Gateway Protocols (EGPs)
    » Only exchange reachability information (not specific metrics)
    » Decide what to do based on local policy

● What properties do we need from a domain?
Autonomous Systems

- Internet is divided into **Autonomous Systems**
  - Distinct regions of administrative control
  - Routers/links managed by a single “institution”
  - Service provider, company, university, …

- Hierarchy of Autonomous Systems
  - Large, “tier-1” provider with a nationwide backbone
  - Medium-sized regional provider with smaller backbone
  - Small network run by a single company or university

- Interaction between Autonomous Systems
  - Internal topology is not shared between ASes
  - … but, neighboring ASes interact to coordinate routing
- **Border routers** summarize and advertise their routes to external neighbors and vice-versa
  - Border routers apply *policy*

- Internal routers can use notion of default routes

- Core is *default-free*; routers must have a route to all networks in the world

- But what routing protocol?
Issues with Link-state

- Topology information is flooded
  - High bandwidth and storage overhead
  - Forces nodes to divulge sensitive information

- Entire path computed locally per node
  - High processing overhead in a large network

- Minimizes some notion of total distance
  - Works only if policy is shared and uniform

- Typically used only inside an AS
  - E.g., OSPF and IS-IS
Advantages
- Hides details of the network topology
- Nodes determine only “next hop” toward the destination

Disadvantages
- Minimizes some notion of total distance, which is difficult in an interdomain setting
- Slow convergence due to the counting-to-infinity problem (“bad news travels slowly”)

Idea: extend the notion of a distance vector
- To make it easier to detect loops
Path-vector Routing

- Extension of distance-vector routing
  - Support flexible routing policies
  - Avoid count-to-infinity problem
- Key idea: advertise the entire path
  - Distance vector: send *distance metric* per destination
  - Path vector: send the *entire path* for each destination

![Diagram of Path-vector Routing]

CSE 123 – Lecture 15: Interdomain Routing
For next time...

- Read Ch 4.1.2 in P&D
- Project 2 out Monday