The Network Layer
Forwarding, Routing, and Addressing
(Part II)
Lecture 23
6.082 Spring 2007
May 4, 2007
Hari Balakrishnan

Layering in the Internet

End-to-End Layer
Everything else!
Reliability, integrity,
ordering, jitter ctrl,
congestion response, ...

Forwarding & routing
(and addressing)
Framing, coding, [limited]
rxmits, channel access
Modulation/demodulation
Network Layer Functions

- **Main goal**: ensure best-effort end-to-end connectivity
- **Addressing**: How to name nodes?
- **Forwarding**: What should each switch do to each packet?
  - Switch design ideas (today)
- **Routing**: How to build routing tables to ensure that forwarding is correct?
  - Link-state protocols
  - Vector routing protocols (today)

Vector Routing Protocols

- **Distributed route computation**
  - Each node sends periodic *route advertisements* about *its* best path to all destinations to neighbors
  - Upon receiving info from neighbor, update best path info if necessary

- What information must advertisement contain?
  - At least distance (cost) to help neighbors update their best path info
  - And possibly more info
Distance Vector Routing

- Periodically announce a vector of \( \langle \text{destination:distance} \rangle \) pairs
- For each advertisement, run “integration step”
  - Incorporate route if distance smaller than stored distance
  - “Bellman-Ford” algorithm

Distance Vector: Pros and Cons

- + Very simple protocol
- + Works well for small networks
- - Works only on small network
- - May have routing loops
- - May “count to infinity”
- Partial solutions:
  - “Split horizon”: Don’t advertise route to node that sent you best route
  - “Poison reverse”: Advertise a cost of “infinity” to node that sent you the best route
- A more complete solution uses a path vector
Path Vector Routing

For each advertisement, run “integration step”
- E.g., pick shortest, cheapest, quickest, etc.
- Ignore advertisements with own address in path vector
Path Vector Pseudocode (Shortest-path Routing)

path ← NULL; // path is current path to destination

procedure n.ADVERTISE() // ADVERTISE is called every T seconds
    for each link do send(n | path); // prepend n to path

procedure n.INTEGRATE(p) // just heard a route p to destination
    if n is in p then return;
    else if ( path == NULL OR
                first_hop(p) == first(path) OR
                length(p) < length(path) )
        path ← p;

procedure n.TIMER() // called periodically to check nbhr liveness
    if haveNotHeardFrom(first(path)) then path ← NULL

Network Layer Functions

• Addressing (next week)

• Forwarding
  • Datagram forwarding using header info
  • Switch design

• Routing
  • Link-state
  • Distance vector & path vector
Basic Forwarding Functions

Early Routers: Bus Backplane

- All links connected via a shared bus
- Simple design
- Bus bottleneck as speeds scale up
Modern Routers: Crossbar Fabric

In each time slot, can “match” each input to at most one output, and vice versa.

What a High Performance Router Looks Like

Cisco GSR 12416
- Capacity: 160Gb/s
- Power: 4.2 kW
- Weight: 256 kg

Juniper T640
- Capacity: 640Gb/s
- Power: 6.5 kW
- Weight: 256 kg
What limits router capacity?

Approximate power consumption per rack

Slide from Nick McKeown (Stanford)
Router linecard

OC192c linecard (10 Gbits/s)

- 30M gates
- 2.5Gbits of memory
- 2-300W
- 1m²
- $25k cost, $100k price.

40-55% of power in chip-to-chip serial links

Slide from Nick McKeown (Stanford)

Summary

- The network layer is the “glue layer” that achieves network-wide reachability
- Forwarding: What a switch does on each packet
- Routing: How the switches build tables to help forwarding
- Next week: Reliable data delivery and network scalability