CS118 Discussion 1B

DHCP, NAT, Routing, Project 2 Tutorial

Week 7

Zhiyi Zhang (zhiyi@cs.ucla.edu)

DHCP

DHCP: Dynamic Host Configuration Protocol

- DHCP server dynamically allocates the following information to a host
  - Host’s IP address, e.g., 192.168.0.10
  - IP mask for the local network, e.g., 255.255.255.0
  - Default router’s IP address (Gateway router), e.g., 192.168.0.1
  - IP address and name for DNS caching resolver, e.g., 8.8.8.8 (Google’s public DNS resolver)
- Why it’s dynamic
  - Once a host goes offline, its address can be re-assigned to another host

More on DHCP

- Protocol overview
  - Over UDP
  - UDP Broadcast for DHCP discovery
  - UDP Broadcast -> IP broadcast -> Link layer broadcast (Broadcast MAC address)
- Process
  - Host broadcasts a DHCP discovery message
  - DHCP server responds with a DHCP offer message (msg's source IP is DHCP server's IP)
  - Host request IP address with DHCP request message (unicast, no need to broadcast)
  - DHCP responds with a DHCP ack message
- Example
  - Check the animation on Chapter 4 lecture slide 46

NAT

NAT: Network Address Translation

- A short term solution to shortage of IP address
  - Use private IP address within the network
  - Use public IP address when communicating with an external host
- Core idea
  - IP:port mapping
    - When sending pkt out
    - Source address: 192.168.0.10: 100 (Private Addr) => 128.97.27.37: 98098 (Public Addr)
- Side effects
  - Security
    - It’s hard for an external host to reach an inner host, e.g., attackers ping your smart home controller
  - Unreachability
    - It’s hard for you to reach your devices behind NAT, e.g., you ping your smart home controller

Address Translation Problems

- Check the example on Chapter 4 slide 61
- When sending packet out:
  - Destination address: won’t change
  - Source address: A: x -> B: y
- When receiving packet in:
  - Destination address: B:y -> A:x
  - Source address: won’t change

IPv6 and IPv4

IPv6

- Long term solution to shortage of IP addresses
- IPv6 address: 128 bits (32 bits)
- Address space: 2^128 addresses
- Fixed header length
  - No variable-length options
- No more in-network fragmentation
  - A IPv6 packet can only be fragmented by the sender
- Removed checksum
- Flow label to identify packets in the same flow

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Header  |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                         Source Address                        +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                      Destination Address                      +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IPv6 Header Format (RFC 2460)

Link State and Distance Vector

Routing Information

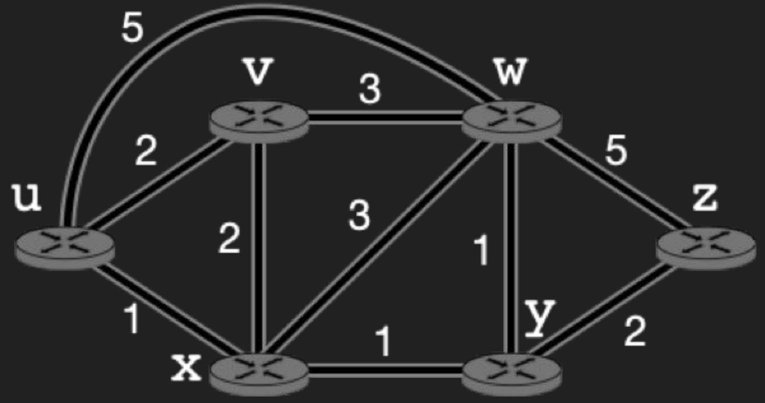
- Global information: Each router obtains the complete topology of the network
  - Link state
- Decentralized information: Each router obtains the information of which direct neighbor the packet should be forwarded based on their destination
  - Distance vector

Link State

- Dijkstra Algorithm
  - Input: Graph topology, Links with weight (cost)
  - Output: Least cost paths from the source to all the other nodes
  - Iteration
    - After k iterations, the source knows least cost path to k destinations
  - Complexity
    - Each iteration checks all the nodes that are not in N'
      - n + (n - 1) + (n - 2) + … + 1, therefore O(n(n+1)/2) = O(n^2)

Learn by Example

- Calculating routing info for u
- Fulfill the table below



N’	To v	To w	To x	To y	To z
u	2, u	5, u	**1, u**	NULL	NULL
ux	2, u	4, x		**2, x**	NULL
uxy	**2, u**	3, y			4, y
uxyv		**3, y**			4, y
uxyvw					**4, y**

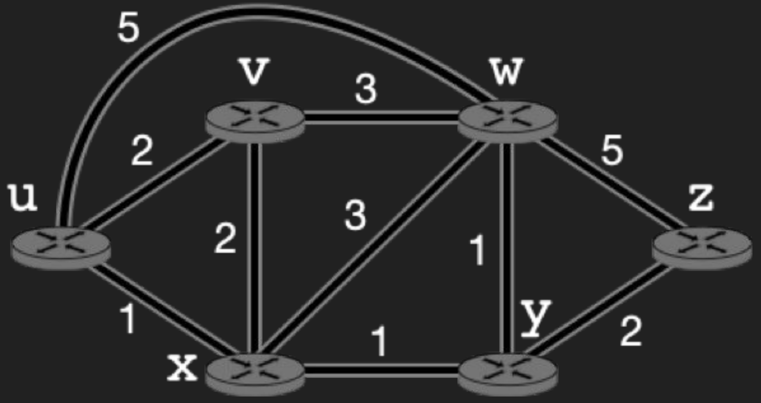
Known Issues in Original Link State

- An example
  - When cost is represented by the directional traffic carried by the link
  - Routers may keep finding new routes and never ends up
  - Check animation on slide 15 in Chapter 5

Distance Vector

- Bellman-Ford equation for dynamic programming
  - $d_x(y)$ is the cost of least cost path from x to y
  - $d_x(y) = \min\{ (c(x, v) + d_v(y))$ for each neighbor v of x $\}$

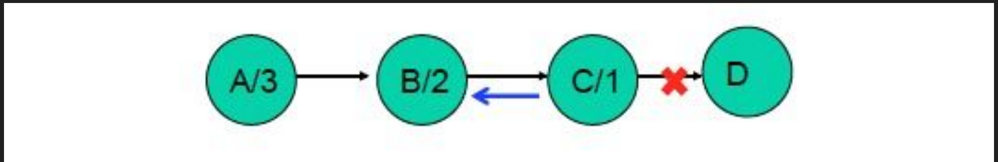
Learn by Example



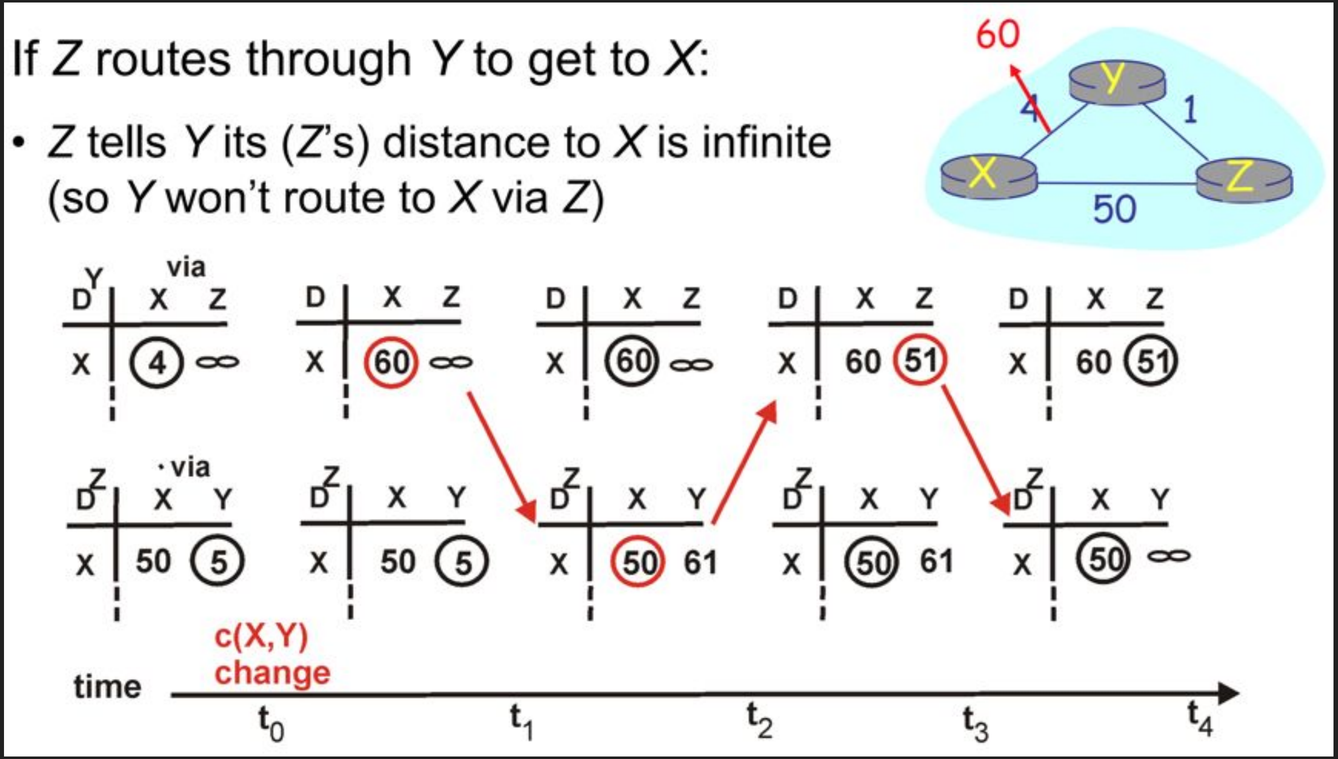
- Calculating routing info for u to z
- d_u(z) = min(
  - c(z, w) + d_w(z): from w to z
  - c(z, v) + d_v(z): from v to z
  - c(z, x) + d_x(z): from x to z
-)
- To know d_w(z), d_v(z), d_x(z), we need to recursively calculate routing info for w, v, x to z respectively.
  - d_w(z) = min(5, c(w,y) + d_y(z), c(w,x) + d_x(z), c(w,v) + d_v(z))
    - = min(5, 1 + 2, ?, ?) = 3
  - Similarly d_x(z) = 3, d_v(z) = 5
- Therefore, d_u(z) = min(5 + 3, 2 + 5, 1 + 3) = 4, next hop should be x

Known Issues in Original Distance Vector

- Count to infinity
  - When C-D broke, B will tell C that B can reach D with cost 2
  - Then c will tell B that C can reach D with cost 3
  - ...
- A possible solution: B doesn't tell C that B can reach D if B reaches D through c: this is called **split horizon**
- Another possible solution: B tells C that B's distance to D is infinite (16): this is called **split horizon with poison reverse**

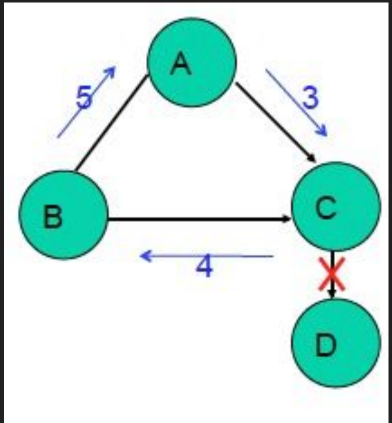


Example of Split Horizon with Poison Reverse



Known Issues in Original Distance Vector (cont'd)

- Problem solved?
- No
    - A and B tell each other their reachability to D
    - When C-D broke, C first report its distance to D is infinite.
    - A will tell C that A can reach D through B with cost = 3
    - Then C will tell B that C now can reach D with cost = 4
    - Then B will tell A that B can reach D through C with cost = 5
    - ...



A summary

- Link State
  - O(nE) messages sent, algorithm complexity is O(n^2)
  - Implemented by OSPF (Open Shortest Path First)
- Distance Vector
  - Exchange among direct neighbors
  - Implemented by RIP (Routing Information Protocol)

Project 2 Tutorial

TAs prepared the slides. Will post to CCLE soon.

- https://docs.google.com/presentation/d/14F6lW6MbpRBJkXhj9t_IH_dcNlJinyEBysAX3b55vwM

UDP Skeleton

- I will code some starting programs in real time