CSC 525:
Computer Networks

The Midterm Exam

- Next Wed 10/12, in class
- closed books/notes.
- Based on lecture slides and papers
  - Covers lectures/papers through last week
  - Not including multicast, no coding.
- You may bring one sheet of paper, US Letter size (8.5x11), double-sided, with any note on it, and a simple calculator.
  - Laptops, smartphones, smart watches etc. are not allowed to be used.

Type of Questions

- Design principles
- Protocol operations
- Comparison and evaluation

Design Principles

- Understand and be able to articulate them
- Understand the tradeoff between different design choices
- Be able to apply the principles to analyze simple design scenarios.

Protocol Operations

- Understand each protocol
  - Algorithm
  - Operations
- Given a scenario or an example, can describe how it works step by step

Comparison and Evaluation

- Compare protocols
  - e.g., RIP vs. OSPF vs. BGP
  - e.g., convergence time, scalability, stability, adaptability, memory usage, message overhead, etc.
- Compare different mechanisms
  - e.g., how is packet loss handled in RIP vs. OSPF.
- Every coin has two sides

Design Principles (I)

- Packet switching
  - versus circuit switching
  - Resiliency, quality of service, network utilization, etc.
- How to build a robust system?
  - Baran's main ideas: redundancy, distributed control, packet switching, and dynamic routing adaptive to changes.

Design Principles (II)

- The goals of original Internet design
  - Survivability, Types of services, Variety of networks
- How did the design achieve these goals?
  - Fate-sharing, IP datagram
- Soft-state vs. hard-state
- End-to-End argument

Unicast Routing Protocols

- Distance Vector: RIP
- Path Vector: BGP
- Link State: OSPF
- Routing states: routing tables and their contents
- Algorithm: how to process an update and what to announce to neighbors
- Operation mechanisms: how to announce and propagate updates, maintain connections, etc.
- Pros and cons

RIP

- One table: dest, cost, next-hop
- Distributed Bellman-Ford algorithm
  - keep the shortest path, exchange with neighbors.
- Periodically re-announce, time out stale entries, triggered updates with a rate-limiting timer.
- Bouncing effect and count-to-infinity problem
- Split horizon and poison reverse as partial solutions.

- Simple, low overhead, distributed
- Slow convergence and routing loops

OSPF

- Maintain entire topology: every node and link
- Calculate shortest paths locally over the topology.
- Reliably flood the link-state information to the entire network
  - Seq. #, per-hop ack, re-flood at low frequency, database sync at session start time.

- Fast convergence, no loop, support multiple metrics
- Complex, high overhead on messages, CPU, and memory

BGP

- Similar to distance vector, but
  - Keep the entire path
  - Keep alternative paths
- Initial table exchange plus incremental updates
  - hard-state, run on top of TCP
- Path exploration causing slow convergence.
  - Real-world experiments and results

Path-Finding Algorithm

- Three tables:
  - link cost, best next-hop, alternative next-hop
  - Use predecessor to reconstruct the entire path
- In processing routing updates
  - Update all paths, including alternative paths
  - Enforce feasibility criterion
  - If cannot find feasible next-hop right away, lock up the path, send queries to neighbors.
- Free of loops, including transient ones.
- Much more complex, freeze data delivery

“Revised Routing Metric”

- Objective
  - devise a new routing metric that limits the route oscillations under heavy traffic load.
- Approach
  - Compress the dynamic range of the metric
  - Smooth the variation of the metric
  - Account for link type
- Explain the simulation results

BGP

- Path selection criteria
  - To control outgoing traffic
- Path Attributes
  - Mainly for traffic engineering purpose.
  - Local preference, ASPATH
- Rate limiting mechanism
  - MRAI and Route Flap Damping
- AS relationship and routing policy
  - Provider, customer, peer, sibling
  - No-valley paths
- Selectively announce routes
  - To control incoming traffic.

Damping

- Interactions
  - Path exploration causes false damping
  - Secondary charging
  - Muffling
- Damping with RCN
  - What is RCN
  - How does it help remove unintended interactions

AS Relationship and Policy

- AS relationship ➔ routing policy ➔implemented by route selection and route announcement ➔ result in no valley BGP paths
  - Impact on path exploration, path redundancy, etc.
- The inference algorithm and its basic idea

Network Topology

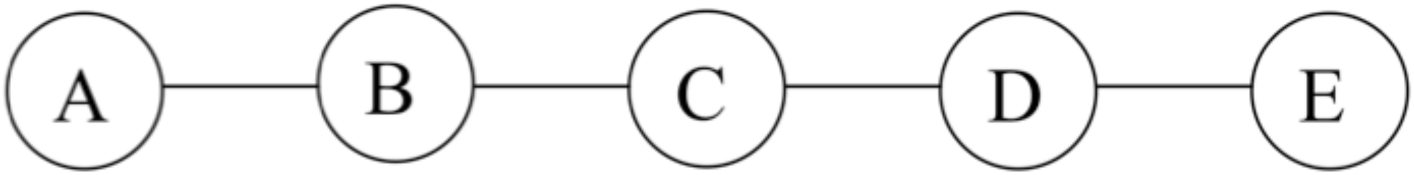
- AS-level topology
  - How BGP data is collected, and how to derive topology from it.
  - The degree power law, and the BA model.
- Route-level topology
  - How traceroute works

Prefix Hijack & Interception

- What is prefix hijack and what is interception?
- Analysis of them with routing policy in consideration.
- Be able to analyze these attacks given a topology and AS relationship.

Sample

In the following network, all the links have relatively high error rate.



To enable reliable file transfer from node A to node E, there're several design choices:

(1) Rely on the network. On each hop, the sending node and the receiving node will do verification and retransmission if necessary, to ensure the bits are transferred 100% reliably on every link.

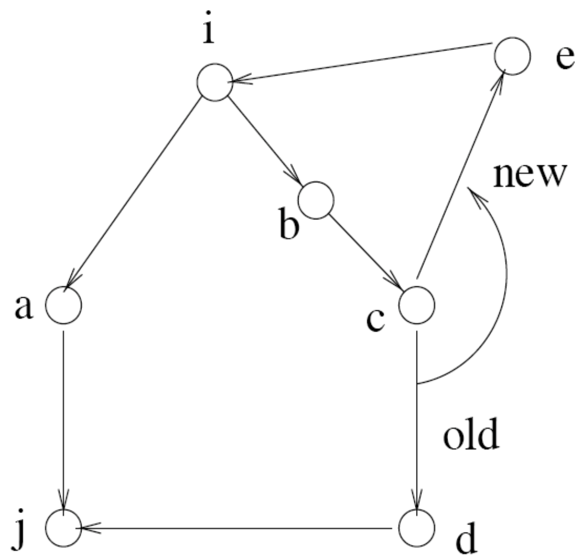
(2) Rely on the ends. Only A and E do the verification and necessary retransmission. Nodes in the middle just forward packets with best effort.

(3) Adopt both (1) and (2)

(4) Adopt (2), but with a better transmission technology to reduce link error rate.

Which option do you recommend? Explain why.

Sample



The paper “A path-finding algorithm for loop-free routing” uses this figure to illustrate temporary loop and how the algorithm can prevent it. Every link has weight of 1, and only consider node j as the destination. Node i has its best path [i, a, j] and two alternative paths [i, b, c, d, j] and [i, e, c, d, j]. Assume link c-d fails, and c decides to switch its best path from [c, d, j] to [c, e, i, a, j]. If link a-i fails after c made its decision. Explain, step by step,

(1) how temporary loop is possible for RIP or OSPF?
(2) how temporary loop is prevented in the path-finding algorithm?

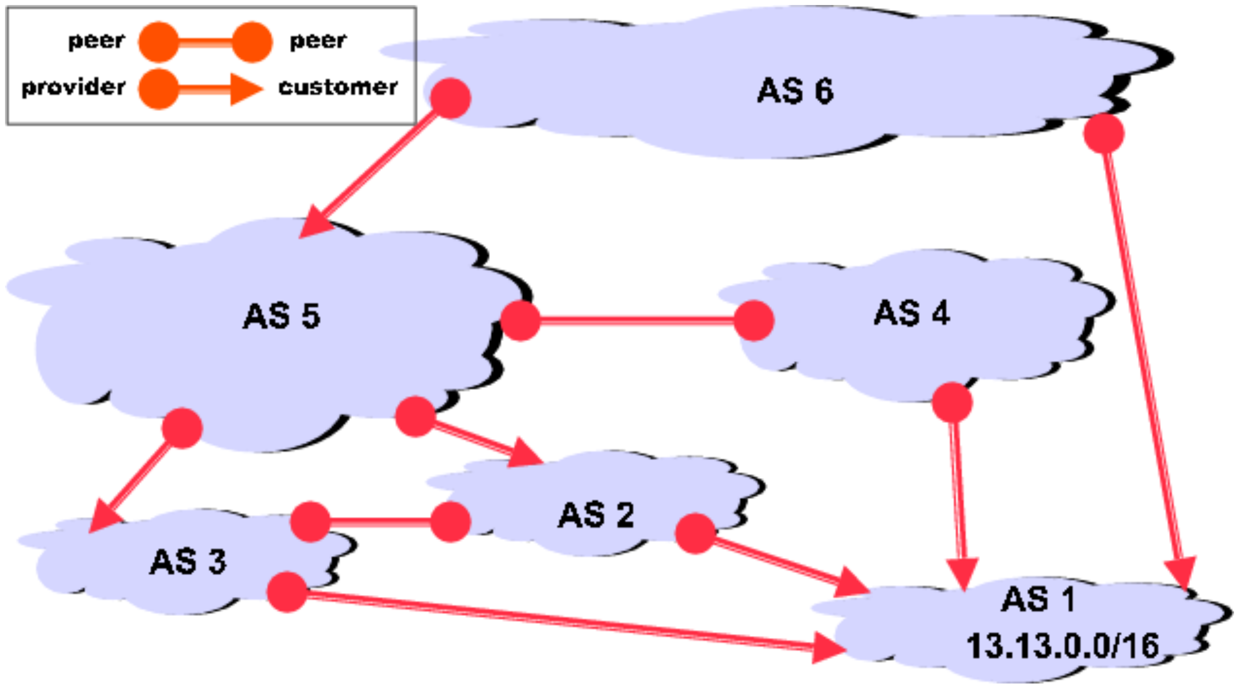
Sample

- (1) To assure correct operation a link-state routing protocol must guarantee reliable delivery of routing updates. Does a Distance-Vector routing protocol also need reliable delivery of routing update messages? What is the consequence if an update is lost in RIP?

- (2) What is the difference, if there is any, in their impact, between the loss of an update in OSPF and the loss of an update in RIP? (the update message is permanently lost, no recovery.)

- (3) List the routing tables and their contents that each unicast router needs to maintain for (a) RIP, (b) path-finding algorithm.

Sample



- To reach destination prefix 13.13.0.0/16, list all valid paths that AS3 can take.
- If AS1 is an attacker and it wants to hijack traffic sent from AS3 to AS5, what BGP announcement AS1 should make, and to which AS? List all successful attacks.