Network Layer V

Dmitri Loguinov
Texas A&M University

April 20, 2021

Original slides copyright © 1996-2004 J.F Kurose and K.W. Ross
Chapter 4: Roadmap

4.1 Introduction
4.2 Virtual circuit and datagram networks
4.3 What’s inside a router
4.4 IP: Internet Protocol
4.5 Routing algorithms
4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
4.7 Broadcast and multicast routing
Inter-AS Routing: BGP

- BGP (Border Gateway Protocol): de facto standard for inter-AS (exterior) routing
- BGP provides each AS a means to:
  - Obtain subnet reachability information from neighboring ASes
  - Propagate the reachability information to all routers internal to the AS
  - Determine “good” routes to subnets based on reachability information and policy
- Allows a subnet to advertise its existence to the rest of the Internet: “I am here”
BGP Basics

- Pairs of routers (BGP peers) exchange routing info over TCP connections: BGP sessions
  - Note that BGP sessions do not correspond to physical links
- When AS2 advertises a prefix 128.194/16 to AS1, AS2 is promising it will forward any datagrams destined to that prefix towards the prefix
  - AS2 can aggregate prefixes in its advertisement

![Diagram of BGP routing](image_url)
Distributing Reachability Info

- With eBGP session between 2a and 1b, AS2 sends prefix reachability info to AS1
  - 1b can then use iBGP do distribute this to all routers in AS1
  - 1c may (if beneficial to AS1) re-advertise these subnets to AS2 over the 1c-3a eBGP session
- Internal AS routers combine intra-AS data with iBGP broadcasts to set up actual forwarding tables
Path Attributes & BGP Routes

- When advertising an IP prefix (i.e., subnet), message includes BGP attributes
  - Notation: combination (IP prefix, attributes) = route
- Two important attributes:
  - **AS-PATH**: contains ASes through which the advert for the prefix passed (latest AS first)
  - **NEXT-HOP**: indicates the router that should receive traffic (usually border router of the AS that advertised prefix; multiple values possible)
BGP Route Selection

- When gateway router receives route advert, it uses an import policy to accept/decline
  - Filters and rules decide allowed/prohibited routes
- Router may learn about more than one route to some prefix, how does it decide which one is better?
  - Multi-exit discriminator (MED) attribute: policy of foreign AS that assigns different weight to different incoming points
  - Shortest AS-PATH
  - Closest NEXT-HOP router: hot potato routing
  - Local preference attribute: policy decision of accepting AS that assigns different weight to various exit points (only used in iBGP)
Example 1: different MED (lower # means higher priority) for paths into AS1

- 7.2.3/24: NEXT-HOP = blue, MED = 10
- 7.2.3/24: NEXT-HOP = red, MED = 50
- 192.10.3/25: NEXT-HOP = blue, MED = 50
- 192.10.3/25: NEXT-HOP = red, MED = 10

Example 2: hot-potato routing in AS2 (orange routers exit right, yellow left)
Customer BGP Policies

- BGP messages exchanged using TCP on port 179
  - Application-layer protocol

- $A, B, C$ are provider networks
- $X, W, Y$ are customer networks
- $X$ is dual-homed: attached to two networks
  - $X$ does not want to route from $B$ via itself to $C$
  - .. so $X$ will not advertise to $B$ any routes picked up from $C$
**Provider BGP Policies**

- A advertises to B and C the path AW
- B advertises to X the path BAW
- Should B advertise to C the path BAW?
  - Not unless B has agreed to route C’s traffic!
    - B gets no “revenue” for routing CBAW since W, A, C are not B’s customers
    - B may want to force C to route to W via A
- ISPs want to route *mainly* to/from their customers!
Chapter 4: Roadmap

4.1 Introduction
4.2 Virtual circuit and datagram networks
4.3 What’s inside a router
4.4 IP: Internet Protocol
4.5 Routing algorithms
4.6 Routing in the Internet
4.7 Broadcast and multicast routing
Multicast and Broadcast

- **Broadcast**: send a packet to all hosts in the network
- **Multicast**: send to a certain subset of nodes
- **Unicast**: one sender - one receiver

**Example**: video distribution to 1M receivers via unicast
- First link R1-R2 carries each packet 1M times
- 5 Mbps stream requires a 5-Tbps link!
Implementing Broadcast

- **(A) Controlled flooding:** routers re-broadcast each received packet only once
  - Must keep a table of all previously received pkts to avoid re-sending of the same data (not scalable)

- **(B) Reverse-path forwarding (RPF):**
  - Routers re-broadcast only packets received on the interface leading towards the source along their own shortest path
  - **Drawbacks of RPF:** redundant packets are still transmitted (e.g., C→B, B→C) and routing must be symmetric
Implementing Broadcast 2

- (C) Minimum Spanning Tree: a tree subgraph of $G$ that spans all network nodes and has the minimum cost of all such trees
  - Once the tree is built, all data travels along the tree, regardless of the source
  - Kruskal’s and Prim’s algorithms build MST in $O(E \log E)$ time

broadcast initiated at A

broadcast initiated at D
Construction of Spanning Trees

- MST is often impractical due to lack of global knowledge
  - Other spanning trees that approximate MST are used instead
- (D) Center-Based Spanning Tree: a “center” node is selected first (various methods exist)
  - All other nodes asynchronously send join requests using unicast routing towards the center until intersection with tree

![Graphical representation of a constructed spanning tree](image)

Stepwise construction of spanning tree (E is the center)
Multicast Routing: Problem Statement

- Broadcast floods the entire Internet and is expensive; in contrast, multicast involves a subset of routers
- Applications
  - Video/audio conferencing: participants form a multicast group to generate and consume content (many-to-many)
  - Video-on-demand or pay-per-view: multicast group is formed by one server and many receivers that consume pre-recorded content (one-to-many)
  - Patch distribution: OS provider distributes updates to hosts running its kernel (one-to-many)
  - Live TV: content received from video provider via multiple servers and fed to many receivers (many-to-many)
- **Goal:** create a tree between routers to which multicast group members are attached
Approaches to Building Mcast Trees

- (A) Source-based mcast forwarding tree: tree of shortest path routes from source $S$ to all receivers
  - Dijkstra’s algorithm when $S$ knows entire topology from some link-state routing algorithm (e.g., MOSPF)
- (B) Source-specific RPF (default opt-in)
  - Initially flood every router (even if R2, R5, R7 don’t want it)
Approaches to Building Mcast Trees

- Forwarding tree may contain subtrees with no mcast group members
  - No need to forward datagrams down subtree
  - “Prune” msgs sent upstream by router with no downstream group members

LEGEND:
- **S**: source
- **@**: router with attached group member
- **@**: router with no attached group member
- **P**: prune message
- **links with multicast forwarding**
Approaches to Building Mcast Trees

• (C) Steiner Tree: minimum cost tree connecting all routers with attached group members
  - Problem is NP-complete

• Even though heuristics exists, not used in practice:
  - Global information about entire network needed
  - Computational complexity
  - Monolithic: rerun whenever a router needs to join/leave

• (D) Center-Based Tree (CBT) (default opt-out)
  - Single delivery tree shared by all
  - One router identified as “center” of tree
  - Join messages sent towards center until existing tree is met
Internet Multicast Routing: DVMRP

- **DVMRP**: Distance Vector Multicast Routing Protocol, RFC 1075 (1988)
- **Flood and prune (default opt-in)**: reverse path forwarding (RPF), tree rooted at source
  - RPF tree based on DVMRP’s own routing tables constructed by communicating DVMRP routers
  - No assumptions about underlying unicast
  - Initial datagram to mcast group flooded everywhere via RPF
- IGMP broadcasts proceed between neighbor routers
- Multicast IP addresses are in 224.0.0.0/4
  - To join a particular group, use `setsockopt` with `IP_ADD_MEMBERSHIP`
Tunneling

**Q:** How to connect “islands” of multicast routers in a “sea” of unicast routers?

- Mcast datagram encapsulated inside “normal” (non-multicast-addressed) datagram
  - Unicast IP datagram sent thru “tunnel” via regular IP unicast to receiving mcast router
  - Receiving mcast router decapsulates mcast datagrams
PIM: Protocol Independent Multicast

Dense (default opt-in):
- Group membership by routers assumed until routers explicitly prune
- Data-driven construction of mcast tree (e.g., RPF)
- Bandwidth and non-group-router processing assumed sufficient

Sparse (default opt-out):
- No membership until routers explicitly join
- Receiver-driven construction of mcast tree (e.g., center-based)
- Bandwidth and non-group-router processing is conservative
Multicast Future

- Wide-area multicast deployment has been traditionally slow, now practically dead
  - Mbone was one such endeavor, worked via tunnels
- One issue is scalability
  - Flooding all Internet receivers is just insane
  - Opens loopholes for DoS attacks
- Another is ISP unwillingness to accept multicast traffic
  - Who pays for a single packet being replicated 1M times?
- Finally, multicast congestion control is hard
  - Mbone had 30-40% loss, which is much more than most applications can tolerate