Multicast

Brad Smith
Logistics

• Projects
  • Status reports due today... summary of how things are going... specifically any problems you’ve run into.

• Thursday
  – JJ presentation – “A Deployable Identifier/Locator Split Architecture”
  – Multicast quiz

• Sunday – BGP lab due

• Monday – BGP extra credit lab due

• Next week – JJ lectures (TBD)
Project Deliverables

- Presentation… 10 mins w/ some time for questions
  - Volunteers for early presentations (Tue/Thu of the last week)? Get 
    special grading consideration… send me e-mail.
  - I’ll post final assignments this coming weekend… people who haven’t 
    volunteered will be randomly assigned to open slots.

- Turn in… by the day of our final slot (Thursday, June 15th)
  - Slides from presentation
  - Paper describing
    - Technology covered in the lab
    - Lessons learned
  - Lab, answer key, netref content
## Project Presentation Schedule

<table>
<thead>
<tr>
<th>Tuesday 6/6</th>
<th></th>
<th>Tuesday 6/15 @ 8am</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slot 1</td>
<td>Bryant Pham (ACLs)</td>
<td>Slot 1</td>
</tr>
<tr>
<td>Slot 2</td>
<td></td>
<td>Slot 2</td>
</tr>
<tr>
<td>Slot 3</td>
<td></td>
<td>Slot 3</td>
</tr>
<tr>
<td>Slot 4</td>
<td></td>
<td>Slot 4</td>
</tr>
<tr>
<td>Slot 5</td>
<td></td>
<td>Slot 5</td>
</tr>
<tr>
<td>Slot 6</td>
<td></td>
<td>Slot 6</td>
</tr>
<tr>
<td>Slot 7</td>
<td></td>
<td>Slot 7</td>
</tr>
<tr>
<td><strong>Thursday 6/8</strong></td>
<td></td>
<td><strong>Saturday 6/10</strong></td>
</tr>
<tr>
<td>Slot 1</td>
<td>Barbara Moretto (EIGRP)</td>
<td>Slot 9</td>
</tr>
<tr>
<td>Slot 2</td>
<td></td>
<td>Slot 10</td>
</tr>
<tr>
<td>Slot 3</td>
<td></td>
<td>Slot 11</td>
</tr>
<tr>
<td>Slot 4</td>
<td></td>
<td>Slot 13</td>
</tr>
<tr>
<td>Slot 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slot 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slot 7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slot 8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Multiple Receivers

• Important applications involve multiple receivers
  – Broadcasts of Radio or Video
  – Videoconferencing
  – Shared Applications

• Special mechanisms needed to efficiently support such applications.
"Together, Internet broadcasting and multicasting are the next chapters in the evolution of the Internet as a revolutionary catalyst for the information age."

Vint Cerf, Senior vice president of MCI/Worldcom, April 1999.
Multicasting

- Multicast communications refers to one-to-many (group) communications.

- IP Multicasting refers to the implementation of multicast communication in the Internet
Multicast using Unicast

• Without support for multicast at the network layer:

Multiple copies of the same message are transmitted on the same link.
With Network Layer Multicast

At most one copy of a message on any link.

Requires a set of mechanisms:
1) Packet forwarding can send multiple copies of same packet
2) Multicast routing algorithm that builds a spanning tree to reach all group members
3) Hosts can join groups
Semantics of IP Multicast

• IP multicast works as follows:
  – Multicast groups are identified by class D addresses...
    • 1110...
    • 224.0.0.0 - 239.255.255.255
  – Every host (interface) can join/leave a multicast group dynamically
    • no access control
  – Every IP datagram sent to a multicast group is transmitted to all members of the group
    • no security, no “floor control”

• IP multicast is...
  – Unreliable
  – Anonymous
IP Multicast Transport Layer

- IP Multicasting only supports UDP as higher layer
- There is no multicast TCP!
Review

• Goal of Network Layer multicast is group communication with at most one copy of a packet on any link.

• Requires
  – Packet forwarding that can send multiple copies of a packet
  – Ability for hosts to join groups
  – Routing algorithm that builds a spanning tree to reach all group members

• Only unreliable (UDP) transport service is defined for multicast.
Three Components of IP Multicast

• IP Multicast Addressing
• IP Group Management
• Multicast Routing
Three Components of IP Multicast

• IP Multicast Addressing
  • IP Group Management
  • Multicast Routing
Multicast Addresses

• All Class D addresses are multicast addresses:

<table>
<thead>
<tr>
<th>Class D</th>
<th>1 1 1 0</th>
<th>multicast group id</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>From</td>
<td>To</td>
</tr>
<tr>
<td>D</td>
<td>224.0.0.0</td>
<td>239.255.255.255</td>
</tr>
</tbody>
</table>

• Multicast addresses are dynamically (i.e. not!) assigned.
• IP datagrams sent to a multicast address are forwarded to all hosts that have joined the group
• When an application is terminated, the multicast address is (implicitly... i.e. not!) released.
Reserved Multicast Addresses

• The address range 224.0.0.0 - 224.0.0.255 is reserved for routing protocols and other low-level topology discovery or maintenance protocols.

• Multicast routers should not forward any multicast datagram with destination addresses in this range.

• Examples of special and reserved Class D addresses...
  – 224.0.0.1  All systems on this subnet
  – 224.0.0.2  All routers on this subnet
  – 224.0.1.1  NTP (Network Time Protocol)
  – 224.0.0.9  RIPv2
Multicast Address Translation

- In Ethernet MAC addresses, a multicast address is identified by setting the lowest bit of the “most left byte”

```
--------1--------
```

- Generally filtered in hardware (on the Ethernet NIC)
- Sometimes filtered in software (in the Ethernet device driver)
Multicast Address Translation

- Identifies Class D
- Ignored
- 23-bit address

Ethernet Addresses with 01:00:5e in the first 3 bytes are reserved for IP multicast
Review

• Multicast groups are defined by class D IP addresses
  – In binary, start with “1110”
  – 224.0.0.0 - 239.255.255.255

• IP addresses can often be directly mapped to MAC addresses
  – E.g. Ethernet
Three Components of IP Multicast

• IP Multicast Addressing
• IP Group Management
• Multicast Routing
The **Internet Group Management Protocol (IGMP)** is a simple protocol for the support of IP multicast.

- Defined in RFC 1112.
- Operates on an IP subnet.
- Used by routers to keep track of membership in a multicast group.

**Support for:**
- Joining a multicast group
- Querying group membership
- Sending membership reports
IGMP Protocol

• A host sends an **IGMP report** when it joins a multicast group
  – Multiple processes can join a group... a report is only sent for the first one.
  – No report is sent when a host leaves a group

• A router regularly multicasts an **IGMP query** to all hosts
  – Group address is set to zero.
  – A host responds to an IGMP query with an IGMP report.

• Routers
  – Track which multicast groups have members on each subnet (but does not keep track of which hosts have joined a group... *anonymous*).
  – Forward packets on subnets that provide access to group members (directly or as transit).
IGMP Protocol

H1

H2

Ethernet

R1

IGMP query

IGMP Report
IGMP Packet Format

- IGMP messages are only 8 bytes long
- Type: 1 = sent by router, 2 = sent by host
IGMP Protocol

IGMP general query
IGMP group address = 0
Destination IP address = 224.0.0.1
Source IP address = router’s IP address

IGMP group-specific query
IGMP group address = group address
Destination IP address = group address
Source IP address = router’s IP address

IGMP membership report
IGMP group address = group address
Destination IP address = group address
Source IP address = host’s IP address
Networks with multiple routers

• Only one router generates IGMP queries (**Querier**)
  – Router with smallest IP address becomes the querier on a network.

• One router forwards multicast packets to the network (**Forwarder**).
Review

• IGMP used to track group membership on subnets.

• Supports
  – Hosts joining multicast groups
  – Routers querying subnets for current membership
  – Hosts reporting membership (in response to queries)

• Hosts
  – Notify routers when they join a group
  – Respond to router queries
  – Are not required to notify routers when they leave a group
Review

- Routers
  - Track which multicast groups have members on each subnet (but don't track which hosts have joined each group)
  - Forward traffic for a group only on subnets that provide access to group members (directly or as transit)

- One router is selected on each subnet to serve as Querier and Forwarder
Three Components of IP Multicast

• IP Multicast Addressing
• IP Group Management
• Multicast Routing
Goal of Multicast Routing

• Build a spanning tree that reaches all members of a multicast group
Multicast routing as a graph problem

- **Problem**: Embed a tree such that all multicast group members are connected by the tree
Multicast routing as a graph problem

- **Problem**: Embed a tree such that all multicast group members are connected by the tree

- **Solution 1**: *Shortest Path Tree or source-based tree* - build a tree that minimizes the path cost from the source to each receiver
  - If there are multiple senders, need one tree per sender
  - Easy to compute
Multicast routing as a graph problem

- **Problem**: Embed a tree such that all multicast group members are connected by the tree

- **Solution 2**: **Minimum-Cost Tree** - build a tree that minimizes the total cost of the edges
  - Single tree serves all senders
  - Very expensive to compute (not practical for more than 30 nodes)
Multicast routing in practice

• Routing Protocols implement one of two approaches:

1. **Source-based Tree:**
   - Essentially implements Solution 1.
   - Builds one shortest path tree for each sender
   - Tree is built from receiver to the sender
     • Use *reverse path forwarding*

2. **Shared- or Core-based Tree:**
   - Build a single distribution tree that is shared by all senders
   - *Does not use Solution 2* (because it is too expensive)
   - Selects one router as a “*core*” (also called “*rendezvous point*”)
   - All receivers build a shortest path to the core
     • Use *reverse path forwarding*
Reverse Path Forwarding (RPF)

- Build a shortest path tree by using unicast routing tables... backwards.
  - Reverse in the sense that the path used to forward unicast traffic to the node is used to forward multicast traffic from the node.

- Routers join the tree when they have downstream group members
  - Notify the unicast next hop router towards the root that they want to join the tree.

- The multicast tree is composed of interfaces
  - On subnets with source or group members (receivers)
  - That connect a router to its RPF neighbor, or
  - With neighbors for which the router is their RPF neighbor

- Multicast forwarding
  - Accept packets from interfaces on the multicast tree, and
  - Forward them out all other interfaces on the tree
Routing table entries differ for source-based and for core-based trees...

- **Source-based** tree: (Source, Group) or (S, G) entry.
- **Core-based** tree: (*, G) entry.

<table>
<thead>
<tr>
<th>Source IP address</th>
<th>Multicast group</th>
<th>Incoming interface (RPF interface)</th>
<th>Outgoing interface list</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>G1</td>
<td>I1</td>
<td>I2, I3</td>
</tr>
<tr>
<td>*</td>
<td>G2</td>
<td>I2</td>
<td>I1, I3</td>
</tr>
</tbody>
</table>
Multicast routing in practice

- Multicast routing algorithms implement one of two approaches:

1. **Source-based tree:**
   - Establish a reverse path to the source(s)

2. **Core-based tree:**
   - Establish a reverse path to the core router (or RP)
Multicast routing in practice

- Multicast routing algorithms implement one of two approaches:

1. **Source-based tree:**
   - Establish a reverse path to the source(s)

2. **Core-based tree:**
   - Establish a reverse path to the core router (or RP)
Building a source-based tree - Flood & Prune

• Determine RPF interfaces from unicast routing tables

- Diagram showing network topology with routers (R1, R2, R3, R4, R5, R6, R7, R8) and hosts (H1, H2, H3, H4, H5). The diagram illustrates the process of flood and prune in a source-based tree, with RPF interfaces identified from unicast routing tables.
Building a source-based tree - Flood & Prune

- **Flood** - forward packets that arrive on RPF interface on all non-RPF interfaces
Building a source-based tree - Flood & Prune

- **Prune** - send a prune message
  - When a packet is received on a non-RPF interface, or
  - When have no downstream receivers
- Disables routing table entry
Pruning

• **Prune message** temporarily disables a routing table entry
  – Effect: Removes a link from the multicast tree
  – No multicast messages are sent on a pruned link
  – Prune message is sent in response to a multicast packet
  – *Why is routing table only temporarily disabled?*

• Who sends prune messages?
  – A router...
    • ...with no group members on directly connected subnets
    • ...which has received prune messages from routers on all non-RPF interfaces
  – A router that has received a packet from a non-RPF neighbor
Building a source-based tree - Flood & Prune

- Receiver drops data packets not received on RPF interface
Building a source-based tree - Flood & Prune

- When a receiver joins need to re-activate a pruned routing table entry
- **Graft** - send a Graft message to disable previous prune, and re-activate a routing table entry.
Building a source-based tree - Explicit Join

- This only works if the receiver’s router knows the source
- **Explicit-Join** – Receiver’s router sends a Join message to RPF neighbor, creates (S,G) routing table entry, and is forwarded.
Multicast routing in practice

- Multicast routing algorithms implement one of two approaches:

1. **Source-based tree:**
   - Establish a reverse path to the source(s)

2. **Core-based tree:**
   - Establish a reverse path to the core router (or RP)
Building a core-based tree

- One router is the core
- Receiver sends a Join message to RPF neighbor with respect to core
- Join message creates (*, G) routing table entry
Building a core-based tree

- Source sends data to the core
- Core forwards data according to routing table entry
Relative Strengths/Weaknesses

• Flood & prune
  – Less configuration
  – Source-based trees lead to less congestion
  – Lots of overhead… best for densely populated groups

• Explicit join
  – More configuration
  – Core-based trees lead to more congestion
  – Less overhead
Review

• Goal of multicast routing is to build a spanning tree that reaches all members of a multicast group

• Two graph-based approaches
  – Shortest-path tree from each source for a group to the group members
  – Build a minimum-cost spanning tree between senders and group members

• Minimum-cost tree solution is too expensive to be practical

• Two practical approaches
  – Source-based tree from above
  – Shared- or Core-based tree where a shortest-path tree is built from a "core" or "rendezvous point" router
Review

• Both solutions depend on the use of RPF paths
  – Build a tree using the unicast routing table entries for the root of the tree
  – When a router has group members downstream of itself from the root (in terms of unicast routes to the root), it notifies the next hop router to the root that it wants to join the tree for that root

• The resulting multicast tree is composed of interfaces
  – Attached to subnets with a source or group member(s)
  – That connect a router to its RPF neighbor
  – With neighbors for which the router is their RPF neighbor

• Multicast forwarding work by
  – Accepting packets from interfaces on the multicast tree, and
  – Forwarding them to all other interfaces on the tree.
Review

• Routing table entries... source-based trees vs. core-based trees
  – Source-based trees use (Source, Group) pairs to access entries
  – Core-based trees use just (*, Group) to access entries
  – Both entries are composed of a list of interfaces on the tree

• Building source-based trees with flood & prune
  – The first packet from a source is flooded over the tree.
  – Prune messages are sent to disable non-RPF links, and links to routers with no downstream group members
  – Prune state is temporary so flood & prune cycle will repeat at some point to detect new members
Review

• Building source-based trees with explicit joins
  – When a router detects a new member it sends a join to the source
  – Requires router to know, or be able to determine, the sources for a group

• Building core-based trees
  – Same as source-based trees with explicit joins, except the root of the tree is the core

• Strengths/Weaknesses
  – Flood & prune
    • Less configuration
    • Source-based trees lead to less congestion
    • Lots of overhead... best for densely populated groups
  – Explicit join
    • More configuration
    • Core-based trees lead to more congestion
    • Less overhead
Existing Multicast Routing Protocols

- **Distance Vector Multicast Routing Protocol** (DVMRP):
  - First multicast routing protocol
  - Implements flood-and-prune

- **Multicast Open Shortest Path First** (MOSPF):
  - Multicast extensions to OSPF. Each router calculates a shortest-path tree based on link state database
  - Not widely used

- **Core Based Tree** (CBT):
  - First core-based tree routing protocol

- **Protocol Independent Multicast** (PIM) (RFC 2362):
  - Two modes: PIM Dense Mode (PIM-DM) and PIM Sparse Mode (PIM-SM).
  - PIM-DM - source-based trees using flood-and-prune
  - PIM-SM - core-based trees as well as source-based trees with explicit joins.
PIM Messages (PIM version 2)

- Encapsulated in IP datagrams with protocol number 103.
- PIM messages can be sent as unicast or multicast packet
- 224.0.0.13 is reserved as the ALL-PIM-Routers group

<table>
<thead>
<tr>
<th>Version (= 2)</th>
<th>Type</th>
<th>Reserved</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Message type specific part

<table>
<thead>
<tr>
<th>PIM-DM messages</th>
<th>Type</th>
<th>PIM-DM</th>
<th>PIM-SM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hello</td>
<td>0</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Register</td>
<td>1</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Register-Stop</td>
<td>2</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Join/Prune</td>
<td>3</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Bootstrap</td>
<td>4</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Assert</td>
<td>5</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Graft</td>
<td>6</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Graft-Ack</td>
<td>7</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Candidate-RP-Advertisement</td>
<td>8</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>
PIM-DM: PIM Dense Mode

- Implements flood-and-prune
- Orange = multicast data
- Blue = PIM message
PIM-SM: PIM **Sparse Mode**

- Implements explicit join
- Core called **rendezvous-point** (RP)
- Receivers know RP (statically configured or dynamically elected)
- When receiver joins, a **Join** message is sent to RP on RPF path.
PIM-SM: PIM Sparse Mode

- Host H3 joins: **Join** message is only forwarded until the first router that is part of the core-based tree.
PIM-SM: Data transmission

- Source sends multicast data to RP attached to an **RP Register** message

- **RP**
  - Forwards data in the tree
  - Sends a Join on RPF path to source

(a) PIM-SM: Register message to RP
PIM-SM: Data transmission

- When Join messages reaches R1, it sends a native multicast packet to the RP (in addition to the packet attached to the register message)
PIM-SM: Data transmission

- When RP receives native multicast packet it sends a **Register Stop** message to R1. This message stops the transmission of register messages from R1.
PIM-SM: Switch to source-based tree

- When data to receivers exceeds a threshold, routers switch to a source-based tree
- This is done by sending an explicit Join message to the source
- There may be duplicate packets being sent for some time

(a) PIM-SM: R3 switches to a SPT
PIM-SM: Switch to source-based tree

- When data arrives from source (as opposed to RP), a Prune message is sent to the RPT
- Data is now forwarded only along the shortest-path tree
Questions?