



---

# Making Multicast Work

**Configuration and Verification  
Troubleshooting**

**DREN Networking Team  
18 August 2010**



# Configuration and Verification



- **What it Looks Like When It's Working**
  - Show Commands
  - Basic Elements
  - Details, Details
- **What to Look for When It's Not**
  - Routes
  - Interfaces, Protocols
  - Architecture
  - Filters



# Configuring the Multicast Network



- **Enable Internet Group Management Protocol (IGMP) for each link with hosts**
  - Enabling Protocol Independent Multicast (PIM) also enables IGMP
  - Group Membership Reports, Joins, Leaves
  - Designated Router (DR) chosen to do active querying
  - DR also converts IGMP information into PIM messages
- **Enable PIM along entire path for session establishment and Source/Receiver propagation**
  - PIM neighbor relationships
  - Negotiate a DR for each local link
- **Define Rendezvous Point (RP) for all routing devices**
  - Static or Local (know where to send PIM messages)
- **Define multicast routing table**
  - Mroute or Routing Information Base (RIB) inet.2
  - May be different paths for each depending on infrastructure
- **Unicast route to RP (Source Registers)**



# Enabling Multicast Traffic on the Network



- **Hosts need ability to handle multicast packets and send IGMP**
- **IGMP used to indicate to routers a desire to join a particular multicast group**
- **DR handles IGMP and sends PIM joins; receives source traffic and sends PIM registers**
- **RP to register hosts within an Autonomous System (AS)**
- **Multicast Border Gateway Protocol (mBGP) for sharing multicast routes between AS's (or use static mroutes)**
- **Routers enabled for multicast routing and an Multicast Intra-Gateway Protocol (M-IGP) for propagating mcast routes internally (or static)**
- **Multicast Source Discovery Protocol (MSDP) shares Active Sources between RP's**



# Configuration and Operation Verification



- **Verify Configuration**
  - Multicast Routing Enabled
  - Unicast Routes, Multiple Route Tables
  - PIM on Interfaces
  - RP Defined
  - Filters, Policies, Access Control Lists (ACLs)
- **Verify Properly Enabled Devices**
  - Show IGMP Interfaces
  - Show PIM Interfaces, Neighbors, DR
  - Show Routes
- **Verify Proper Paths**
  - Traceroute
  - Mtrace
- **Verify Session Establishment/Packet Flow**
  - Show IGMP Groups/Members
  - Show PIM Join
  - Show Internet Protocol (IP) Mroute, Show Multicast Route Detail
  - Show MSDP Source-Active
- **Other Tools for Network Characterization**



# Troubleshooting Multicast: Network Characterization



- **Verify IGMP is working correctly**
  - Tcpdump Ethereal
- **Verify PIM is working correctly**
  - debugging
- **Verify RP defined at the router**
- **Verify multicast routing table**
  - Mroute or RIB inet.2
- **Verify Unicast route to RP (Sources)**
- **Verify ACLs or Firewalls configurations**



# Troubleshooting Overview



- **Identify the information you need to gather information, such as:**
  - What are you trying to receive?
  - What is the source IP?
  - What is the receiver group?
- **Make sure that there are receivers interested in receiving traffic.**
- **Verify knowledge of Sources.**
- **Verify RP Source registration.**
- **Trace the forwarding state back using the multicast routing table.**



# Troubleshooting a Receiver



- Is the interface facing the hosts in PIM sparse mode?
- Are IGMP reports getting to router?
- Cisco routers “show ip igmp groups”
- Juniper routers “show igmp groups”
- Use Ethereal or Tcpcmdump on the host.
- Debug IGMP on a Cisco router? (very processor intensive)
- Debug IGMP on a Juniper router.



# Troubleshooting a Receiver



- **Verify that hosts are sending reports**
- **Verify that routers are sending queries**
- **If both are the case, could it be switch issues? This can be a problem with older operating systems on switches.**
- **IGMP snooping is used on switches to prevent excess flooding.**



# Troubleshooting PIM



- Do you have the interfaces facing RP in PIM?
- Do you have PIM neighbor relationship between routers?
- Cisco routers “show ip pim neighbor”
- Juniper routers “show pim neighbor”
- Debug PIM on a Cisco router? (very processor intensive)
- Debug PIM on a Juniper router.



# Troubleshooting RP-Tree (RPT)



- Once you have verified that that IGMP is working correctly and PIM is enabled correctly and you have PIM neighbors, its time to make sure your Joins are in the routers.
- In Cisco use the “show ip mroute” to look for the ( \* , G)
- In Juniper use the “show pim joins” to look for the ( \* , G)
- As the join travels to the RP, the router interface it came in on is added to the Outgoing Interface List (OIL)
- The ( \* , G) will travel from downstream to upstream interfaces until it reaches RP.



# Troubleshooting Multicast Source



- Can you ping the RP?
- Check RP routers to see if they have the registration.
- How?
  - Juniper router use the “show pim rp extensive”
  - For MSDP use the “show msdp source-active”



# Other Troubleshooting Tools



- **Mping**
  - Ping a multicast address on a certain port from both ends
  - Always uses the interface that has the default route
  - Creates Multicast sessions, but no indication of why it doesn't work
- **Mtrace**
  - Uses IGMP to map the PIM path source to receiver (lots of options, depending on vendor)
  - Is NOT PIM, so not entirely telling, but supports third party



# Mping Success case



- **IPv4 example**

Receiver

```
root@nodeA:/# mping -r -R -c 3
mping version 1.1
Listening on 227.1.1.1/4098:
```

```
Replying to mping from 9.3.207.195 bytes=32 seqno=0 ttl=1
Replying to mping from 9.3.207.195 bytes=32 seqno=1 ttl=1
Replying to mping from 9.3.207.195 bytes=32 seqno=2 ttl=1
```

Sender

```
root@nodeB:/# mping -R -s -c 3
mping version 1.1
mpinging 227.1.1.1/4098 with ttl=1:
```

```
32 bytes from 9.3.207.190 seqno=0 ttl=1 time=0.985 ms
32 bytes from 9.3.207.190 seqno=1 ttl=1 time=0.958 ms
32 bytes from 9.3.207.190 seqno=2 ttl=1 time=0.998 ms
```

```
--- 227.1.1.1 mping statistics ---
```

```
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.863/0.941/0.998 ms
```



# Mping Failure case



- **IPv6 example**

Receiver

```
root@nodeA:/# mping -r -R -c 3 -6  
mping version 1.1  
Listening on ff05::7F01:0101/4098:
```

```
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=0 ttl=1  
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=1 ttl=1  
Replying to mping from fe80::18ae:19ff:fe72:1a15 bytes=48 seqno=2 ttl=1
```

Sender

```
root@nodeB:/# mping -R -s -c 3 -6  
mping version 1.1  
mpinging ff05::7F01:0101/4098 with ttl=1:
```

```
--- ff05::7F01:0101 mping statistics ---  
3 packets transmitted, 0 packets received, 100% packet loss  
round-trip min/avg/max = 0.000/0.000/0.000 ms
```



# Mtrace



- **Similar in Function to Traceroute, but much more complex**
  - Request from anywhere, Response to anywhere
  - Hop-by-hop trace from receiver back to source and report multicast connectivity from source to receiver
- **Each router in the path adds its own routing, interface/path, and Multicast statistics, passes the request on to the next hop. All data collected and sent to the response address in an Mtrace Response Packet (MRESP)**
- **Basic command Mtrace <source> or define the following**
  - Multicast Source (to trace back to)
  - Multicast Receiver (to trace from, default is requesting IP)
  - Multicast Group (to use in the trace, 0.0.0.0 if not set)
  - Response Address (default is requesting IP, response may be set to unicast or multicast address)



# Mtrace Query Default Behavior

- Attempts to trace the full reverse path, (unless the number of hops to trace is explicitly set with the -m flag).
- If no response within a 3-second timeout interval (changed with the -w flag), an \* (asterisk) is printed and switches to hop-to-hop mode.
- Hop-to-hop Mode queries are issued starting with a maximum hop count of one and increasing by one until the full path is traced or no response is received.
- Multiple Probes sent at each hop (default is three, changed with -q flag), the first 2 with the reply address set to standard multicast address, mtrace.mcast.net (224.0.1.32) with the time to live (ttl) set to 32 more than what is needed to pass the thresholds seen so far along the path to the receiver. For each additional attempt, the ttl is increased by another 32 each time up to a maximum of 192.
- Because the desired router may not be able to send a multicast reply, the remainder of the attempts request that the response be sent via unicast directly to the host running the Mtrace command.