IGMP/MLD Snooping in Bridge Driver

Satish Ashok — Cumulus Networks

LinuxCon
August 18th, 2015
Topics

Introduction to IGMP/MLD Snooping

- Hardware offload
- MultiChassis Link Aggregation
- Vlan Filtering
- STP with IGMP Snooping
- Future Enhancements
Introduction to Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD) Snooping

Bridge - A

swp1
swp2
swp3

IGMP / MLD
Query messages

IGMP / MLD
Report and leave messages

host-1
host-2

IGMP / MLD
Router

IGMP / MLD Snooping Switch
IGMP/MLD Snooping Topology
Introduction to IGMP/MLD Snooping

- Bridge driver maintains Multicast Database (MDB) groups and router ports

- IGMP V1/V2/V3 and MLD V1/V2

- Automatic and static router port

- Querier configuration and multicast timers
```markdown
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>setmclmc</code></td>
<td>set multicast last member count</td>
</tr>
<tr>
<td><code>setmclmi</code></td>
<td>set multicast last member interval</td>
</tr>
<tr>
<td><code>setmcmi</code></td>
<td>set multicast membership interval</td>
</tr>
<tr>
<td><code>setmcqi</code></td>
<td>set multicast query interval</td>
</tr>
<tr>
<td><code>setmcqifaddr</code></td>
<td>set multicast query to use ifaddr</td>
</tr>
<tr>
<td><code>setmcqpi</code></td>
<td>set multicast querier interval</td>
</tr>
<tr>
<td><code>setmcqri</code></td>
<td>set multicast query response interval</td>
</tr>
<tr>
<td><code>setmcquerier</code></td>
<td>set multicast querier</td>
</tr>
<tr>
<td><code>setmcqv4src</code></td>
<td>set multicast ipv4 querier address</td>
</tr>
<tr>
<td><code>setmcrouter</code></td>
<td>set multicast router</td>
</tr>
<tr>
<td><code>setmcsnoop</code></td>
<td>set multicast snooping</td>
</tr>
<tr>
<td><code>setmcsqc</code></td>
<td>set multicast startup query count</td>
</tr>
<tr>
<td><code>setmcsqi</code></td>
<td>set multicast startup query interval</td>
</tr>
</tbody>
</table>
```

MDB Router/Group State:

cumulus@switch:~# bridge mdb help
Usage: bridge mdb { add | del | replace} dev DEV port PORT grp GROUP [permanent | temp] [ vlan VID ]

bridge mdb {show} [ dev DEV ]

cumulus@switch:~# sudo bridge -d mdb show
dev br0 port swp2 grp 234.10.10.10 temp
dev br0 port swp1 grp 238.39.20.86 permanent
dev br0 port swp1 grp 234.1.1.1 temp
dev br0 port swp2 grp ff1a::9 permanent
router ports on br0: swp3
```
cumulus@switch:~# sudo brctl showstp br0

br0
bridge id              8000.7072cf8c272c
designated root        8000.7072cf8c272c
root port               0
max age                 20.00
hello time              2.00
forward delay           15.00
ageing time             300.00
hello timer             0.00
topology change timer   0.00
hash elasticity         4096
mc last member count    2
mc router               1
mc last member timer    1.00
mc querier timer        255.00
mc response interval    10.00
mc querier              0
flags

swp1 (1)
port id                8001
designated root        8000.7072cf8c272c
designated bridge      8000.7072cf8c272c
designated port        8001
designated cost        0
mc router               1
flags
```
Hardware Offload (In kernel driver/Userspace):

Option 1: Switchdev:
- In kernel switch ASIC driver implementing switchdev API to offload to switch ASIC

Option 2: Hardware offload driver in Userspace:
- Listens to netlink notifications and programs/offloads to switch ASIC
IGMP/MLD Snooping offload with userspace switch ASIC driver

User Space

Linux Kernel

- Multicast Database
- Table in Bridge Driver
- Ethernet Interfaces

CPU, RAM, Flash, etc.

Switch HAL

- switchd
- Switch Driver
- Switch Silicon
- Front Panel Ports
Hardware Offload Changes:

- IGMP snooping enabled, hardware punts IGMP/MLD messages to CPU
- Don’t send unknown multicast to CPU, only reserved link-local multicast
- Create cache of MDB router ports and groups per VLAN
- Program MDB in hardware - union of group and router ports
Hardware Offload Changes (Contd):

- To Scale: Avoid creating groups for reserved link-local multicast (224.0.0/24, FF02::xx, FF02::1:FFxx:xxxx)

- Optimized Router forwarding

- SOC create groups using group IP or mapped MAC
Multi-Chassis Link Aggregation Group (MLAG)

MLAG

- MLAG enables host/switch with a 2 port bond connected to 2 different switches, operate as if they are connected to 1 switch
Multi-Chassis Link Aggregation Group (MLAG)

- Duplicate Packets: Packets received on peer link not sent to dually connected bond

- Continual Address Movement: Mac learning disabled on peer link

- Black Holing: Sync dynamic learnt Mac address to dually connected link
MLAG with IGMP Snooping Code Changes

- CLAG daemon synchronizes and refreshes MDB groups and router ports on peer switch

- Added “peerlink” and “duallink” bridge port attributes - Broadcast, unknown unicast and Multicast traffic not forwarded from peer link to dual-link in bridge driver
Vlan filtering in Bridge Driver

- VLAN filtering feature in bridge driver

- VLAN configuration on bridge ports - port VLAN ID, allowed vlans

  bridge vlan add vid 100 dev bond0
  bridge vlan add vid 101 dev bond0
  bridge vlan add vid 102 dev bond0
  bridge vlan add vid 103 dev bond0
  bridge vlan add vid 104 dev bond0
  bridge vlan add vid 10 untagged pvid dev bond0
Vlan filtering in Bridge Driver

ifupdown2 stanza:

auto bridge
iface bridge
  bridge-vlan-aware yes
  bridge-ports swp1 swp2
  bridge-vids 100 200
  bridge-pvid 1
  bridge-stp on
Vlan filtering Code Changes Upstreamed

- Maintain VLAN when MDB group created
- Add/delete static MDB for VLAN
- Netlink notifications per VLAN for router port
Vlan filtering Future Code Changes

- All MDB configuration is per bridge, ideally it needs to be per VLAN per bridge

- At Least, snooping enable/disable, querier enable and IP configuration needs to be per VLAN
STP with IGMP Snooping:

- If IGMP snooping is enabled, on STP topology change, send IGMP query to reduce network convergence time (RFC 4541, Section 2.1.1)

- Send general leave instead, so that active querier sends query
Source Code, Documentation

Source Code:
oss.cumulusnetworks.com

Documentation:

- http://docs.cumulusnetworks.com/display/DOCS/IGMP+and+MLD+Snooping
- http://docs.cumulusnetworks.com/display/DOCS/Multi-Chassis+Link+Aggregation+-+MLAG
- http://docs.cumulusnetworks.com/display/DOCS/VLAN-aware+Bridge+Mode+for+Large-scale+Layer+2+Environments
References

RFCs:
- RFC 2236, RFC 3376, RFC 4604 - IGMPv1, IGMPv3, MLDv2
- RFC 4541 - Considerations for IGMP and MLD snooping switches

Manpages:
brctl, bridge
Thank You!

CUMULUS, the Cumulus Logo, CUMULUS NETWORKS, and the Rocket Turtle Logo (the “Marks”) are trademarks and service marks of Cumulus Networks, Inc. in the U.S. and other countries. You are not permitted to use the Marks without the prior written consent of Cumulus Networks. The registered trademark Linux® is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a world-wide basis. All other marks are used under fair use or license from their respective owners.