Configuring Multicast networking and an IGMP Querier in a Layer 2 domain

In this topology both the sender (30.0.0.1) and receiver (30.0.0.2) are connected to the same Cisco L3 switch (3750). The routing engine is enabled on the L3 switch.

The sender and receiver are both connected to the same VLAN 30. No multicast options are configured outside the default. IGMP snooping IS enabled by default as usual:
The traffic captures below have been produced by Wireshark / tcpdump.

```
#show ip igmp snooping vlan 30
Global IGMP Snooping configuration:
-------------------------------------------
IGMP snooping                : Enabled

[…]

Vlan 30:
--------
IGMP snooping                        : Enabled
```

When the sender generates Multicast transmissions, will they be seen on the receiver interface within the same VLAN?
For this purpose, the sender will ping to 239.100.100.100. On the receiver the traffic is received:

```
IP 30.0.0.1 > 239.100.100.100: ICMP echo request, id 7170, seq 101, length 64
```

Note: if Linux will allow you to ping to the multicast address, then make sure you provide a Multicast Route using the outgoing interface, example:

```
# ip route add 224.0.0.0/4 dev eth0
```

The Multicast ping is *flooded* to the port of the recipient, despite that the recipient at this point has not flagged any interest via IGMP.

So, despite the fact that IGMP snooping should prevent the flooding from happening, it has not been activated yet as there is *no IGMP querier* present.

The default says :

"In the absence of a pim enabled router or igmp querier within the vlan, the multicast is just treated as a broadcast frame, ie -- flooded to all ports in the vlan, This is the same behavior if there was a querier and 'no ip igmp snooping vlan X' was configured, all the ports would get it."

And:

"In regards to IGMP snooping and flooding behavior, different Cisco switches will operate differently. For example, on 2960/3560/3750 platforms, if there is no IGMP snooping mrouter detected on the vlan, the switch will flood all multicast packets. Other platforms, will not flood but also will be unable to program their IGMP snooping tables preventing multicast traffic from working on the same vlan entirely."

So what we need in this Layer 2 domain is an *IGMP querier*. We do not want to introduce a 2nd device as Multicast Router at this stage though. So, the way to solve this is by making the L3 switch an *IGMP querier*. Of course, it will need an IP source address to send the queries, we'll configure this first.

We'll create an SVI / VLAN interface in VLAN 30 (and we'll enable the routing engine just to make sure, it probably already is) :

```
ip routing
interface vlan 30
       ip address 30.0.0.254 255.255.255.0
       No shutdown
```

Note: as opposed to using an SVI/VLAN Interface, we could have used a physical interface, converted it to L3 with "no switchport" and assigned the IP address there, it's all a matter of topology really.

Once you enable debugging for IGMP snooping and manually disable and enable IGMP snooping, the MLS will actually tell you why it's not working:

```
(config)# no ip igmp snooping vlan 30
(config)# ip igmp snooping vlan 30
vlan_id 30: cannot be oper enabled due to: IGMP switch querier is disabled
on this vlan
```

Now enable *ip pim passive* on the interface VLAN 30 which *should trigger the IGMP querier*.

```
(config-if)#ip pim passive
 WARNING: "ip multicast-routing distributed" is not configured,
         IP Multicast packets will not be forwarded
*Mar  1 00:49:13.975: %PIM-5-DRCHG: DR change from neighbor 0.0.0.0 to 30.0.0.254 on interface
Vlan30
```

Since we are not routing the multicast stream here, the warning can safely be ignored.

The multicast stream is now no longer flooded and the IGMP querier has started on the L3 switch, querying for interested parties:

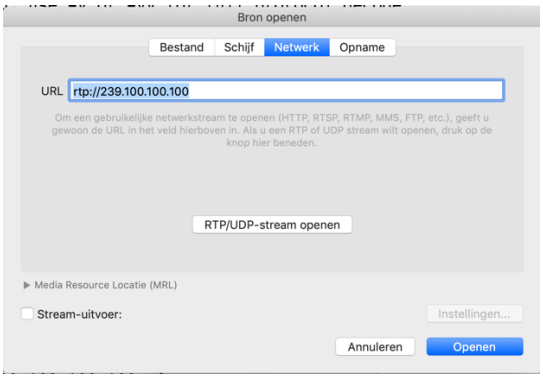
```
16:46:31.123706 IP 30.0.0.254 > 224.0.0.1: igmp query v2
```

The switch will show that there are no interested parties at current :

```
# show ip igmp membership
Flags: A  - aggregate, T - tracked
       L  - Local, S - static, V - virtual, R - Reported through v3
       I - v3lite, U - Urd, M - SSM (S,G) channel
       1,2,3 - The version of IGMP, the group is in
[…]

 Channel/Group                  Reporter        Uptime    Exp.  Flags  Interface
```

Now how *do* we now singal our interest in receiving the IGMP stream on the Receiver? For this we can use a simple Multicast Client like 'vlc' and tell it to listen op 239.100.100.100:



VLC will craft the IGMP v.2 membership report on the receiver (tcpdump -i eth0 -n igmp) :

```
16:51:48.354280 IP 30.0.0.2 > 239.100.100.100: igmp v2 report 239.100.100.100
```

The Ping is now forwarded successfully by the Switch to the interested party only and echoed back by the receiver!

```
#show ip igmp membership
Flags: A  - aggregate, T - tracked
       L  - Local, S - static, V - virtual, R - Reported through v3
       I - v3lite, U - Urd, M - SSM (S,G) channel
       1,2,3 - The version of IGMP, the group is in
Channel/Group-Flags:
       / - Filtering entry (Exclude mode (S,G), Include mode (G))
Reporter:
       <mac-or-ip-address> - last reporter if group is not explicitly tracked
       <n>/<m>      - <n> reporter in include mode, <m> reporter in exclude

 Channel/Group                 Reporter        Uptime   Exp.  Flags  Interface
 *,239.100.100.100             30.0.0.2        00:01:37 02:26 2A     Vl30
```

Notice the Reporter IP address here -30.0.0.2- is the address of the receiver where VLC runs.

What happens when we stop the stream within VLC?

```
16:53:22.418604 IP 30.0.0.2 > 224.0.0.2: igmp leave 239.100.100.100
```

Clearly VLC sends an IGMP *leave report*, whereas the IGMP querier (L3 switch) will immediately send out membership queries, to find out if there are any other interesting parties remaining.

➔ **`#show ip igmp membership`** → no longer shows us any clients as expected and the ping is no longer forwarded.

Alternatively, the following command can be used to trigger the IGMP query:
`(config)#ip igmp snooping querier address 30.0.0.254`

Sometimes the '**`show ip igmp membership`**' command seems to be lacking and does not show IGMP members despite that IGMP snooping works.

In that case, use:

```
#sh ip igmp snooping groups vlan 30
Vlan      Group                Type         Version     Port List
------------------------------------------------------------------------
30        239.100.100.100      igmp         v2          Fa1/0/5
```

What also shares light on the matter is "**`deb ip igmp snooping`**",- (don't forget **`terminal monitor`** if you are not connected to the console) but of course you should be conservative with debugging Cisco MLS in a production environment!

```
#deb ip igmp snooping
```

Another thing to consider is that once the querier functionality has been removed, it takes some time before IGMP snooping will really fail again. Disabling / enabling IGMP snooping can speed up the process: "[**`no] ip igmp snooping vlan 30`**".