EXTEND L2POP FDB INFORMATION

Layer 2 population (l2pop) mechanism driver implements the ML2 driver to improve open source plugins overlay implementations (VXLAN with Linux bridge and GRE/VXLAN with OVS)**[1]**. L2pop avoid the broadcast in mac learning and ARP resolution by prepopulate the bridge forwarding table**[2]**. However, some projects connect neutron network with outside networks, for example, project bgpvpn-networking**[3]** can interconnect BGP/MPLS VPNs to Openstack Neutron networks, routers and ports**[4]**. For the connection to the outside networks, l2pop may need to provide more information than what it does now.

This rfe proposes to add an extension registration mechanism to l2pop so that other subprojects can extend the information included in a full FDB messages before it is sent to the agent.

Problem Description

Some projects connect neutron network with outside networks. For example, project bgpvpn-networking aims at supporting inter-connection between L3VPNs and Neutron resources, i.e. Networks, Routers and Ports**[5]**. A typical use-case is the following: a tenant already having a BGP IP VPN (a set of external sites) setup outside the datacenter, and by using the project bgpvpn-networking they can establish the connectivity between VMs and these VPN sites.

Figure-1 illustrates an environment that an openstack deployed in the DataCenter and BGPVPN-1 is a bgp-based vpn. The openstack has enables l2pop and bgpvpn. When a vm first sends a packet to some device in the bgpvpn, broadcast for mac learning and ARP resolution can’t be avoided. The use case is listed below:

Use Case

1. The cloud/network admin creates BGPVPN-1 for a tenant based on contract and OSS information about the VPN for this tenant
2. The tenant associates BGPVPN-1 with network-1(VM-11 belongs to network-1)
3. The vm-11 in network-1 first sends a packet to the device-1 in BGPVPN-1.

+------------------------------+         +------------+ +---------+

|          DataCenter         | | WAN   | | BGPVPN-1|

|                         +----+ +----+ +----+        +----+ |

|                         |CE-1+-------- +PE-1| |PE-2+--------+CE-2|    |

|                         +----+ +----+ +----+        +----+ |

|+----------------------------+|         | | |+--------+|

||          Host 0      || +------------+        ||Device-1||

|| +-----+  +-----+ +-----+ ||                               |+--------+|

|| |VM-01|  |VM-02|...|VM-0m| ||                               | |

|| +-----+  +-----+ +-----+ ||                               |+--------+|

|+----------------------------+|                               ||Device-2||

|                              | |+--------+|

|+----------------------------+|                               | . |

||          Host 1      ||                 | . |

|| +-----+  +-----+ +-----+ ||                               | . |

|| |VM-11|  |VM-12|...|VM-1p| ||                               |+--------+|

|| +-----+  +-----+ +-----+ ||                               ||Device-t||

|+----------------------------+|                               |+--------+|

|              . |                               | |

|              . |                               +----------+

|              . |

|+----------------------------+|

||          Host n      ||

|| +-----+  +-----+ +-----+ ||

|| |VM-n1|  |VM-n2|...|VM-nq| ||

|| +-----+  +-----+ +-----+ ||

|+----------------------------+|

+------------------------------+

**Figure-1**

In the step 3, broadcast for mac learning and ARP resolution can’t be avoided. Because neutron doesn’t have the port information of the device in the bgpvpn, thus the fdbs sent by neutron won’t include the port information of the Device-1. As a result of that, before step 3, there are no flows related to Device-1 existing in the Host 1. So ARP request are sent out.

Therefore, we are introducing an extension registration mechanism which enables other subprojects such as project bgpvpn-networking can register their own function to l2pop. The registered function can add the port information in the outside networks to the fdbs. Thus the broadcast for mac learning and ARP resolution can be avoided.

Proposed Change

The idea is to add an extension registration mechanism to l2pop mech\_driver.py so that other subprojects can register their own function to l2pop. A global variable l2pop\_fdb\_extend\_funcs should be created to store the registered function by other subprojects. Function register\_fdb\_extend\_func and run\_fdb\_extend\_funcs should created. Function register\_fdb\_extend\_func should be used by other subprojects to register function to l2pop. For example, project bgpvpn-networking can import the l2pop\_driver by line 1 and define a new function to register its own function bgpvpn\_fdb\_extend\_func to l2pop\_driver by line 2 and 3.

*1      from neutron.plugins.ml2.drivers.l2pop import mech\_driver as l2pop\_driver*

*2      def register\_callbacks(self):*

*3             l2pop\_driver.register\_fdb\_extend\_func( constants.BGPVPN,*

*self.bgpvpn\_fdb\_extend\_func)*

Function run\_fdb\_extend\_funcs should be called every time fdbs are created. Thus the pre-registered functions stored in the global variable l2pop\_fdb\_extend\_funcs will be called to extend the newly created fdb. Thus subprojects can extend the information included in the fdb by such mechanism. All changes can be viewed through the link below:

<https://review.openstack.org/#/q/I87450e332c1a6d8bd529eb8082292e73c533676e>

Data Model Impact

None

REST API Impact

None

Command Line Client Impact

None

Other Impact

None

Other Deployer Impact

None

Performance Impact

Performance testing should be conducted to see what is the overhead of adding more information to fdb.

Implementation

Assignee(s)

Work Items

* Add the register\_fdb\_extend\_func and run\_fdb\_extend\_funcs functions to l2pop mech\_driver.py.
* Add related tests.
* Documentation.

Dependencies

None

Testing

Unit tests, functional tests and scenario tests are necessary.

Documentation Impact

How to register the fdb\_extend\_function to l2pop should be documented.

References

**[1]** <https://github.com/openstack/neutron/tree/master/neutron/plugins/ml2/drivers/l2pop>

**[2]** <https://wiki.openstack.org/wiki/L2population_blueprint>

**[3]** <https://github.com/openstack/networking-bgpvpn>

**[4]** <https://github.com/openstack/networking-bgpvpn>

**[5]** https://docs.openstack.org/networking-bgpvpn/latest/user/overview.html