**Patch ID: 9671feb**

**Patch Description:** l2pop: allow csnat ports in list of endpoints

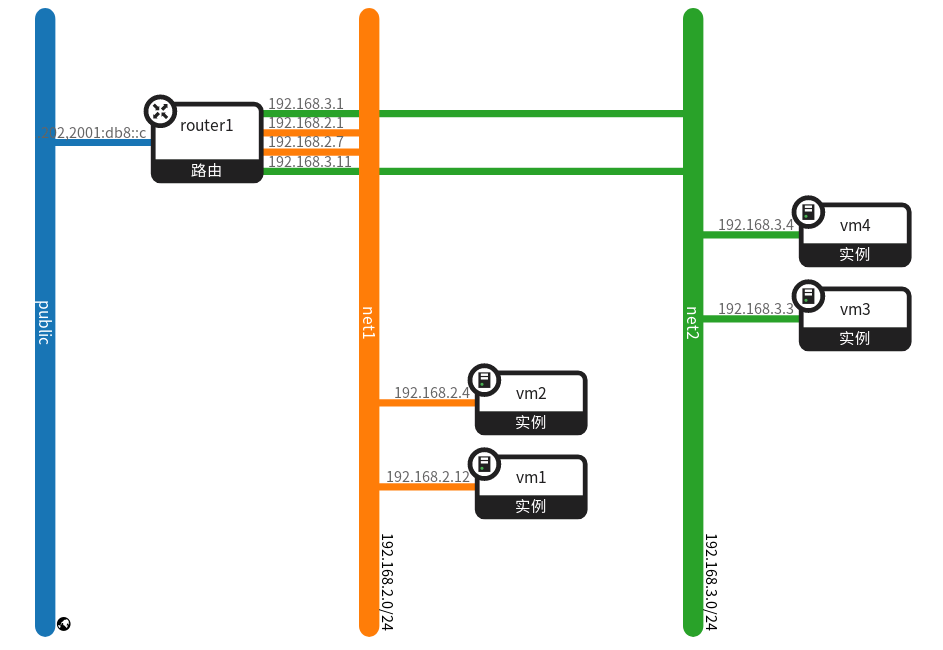
   Not quite sure why upstream is not impacted by this issue, but we need to have CSNAT ports included in the list of ports that the l2pop driver handles. Without this it is not possible for a VM on one node to reach the CSNAT port because the MAC address of the CSNAT port is not included in the static FDB entries.  Without this our vswitch drops the outgoing packet since that's what it does with packets to unknown destinations while in static mode.

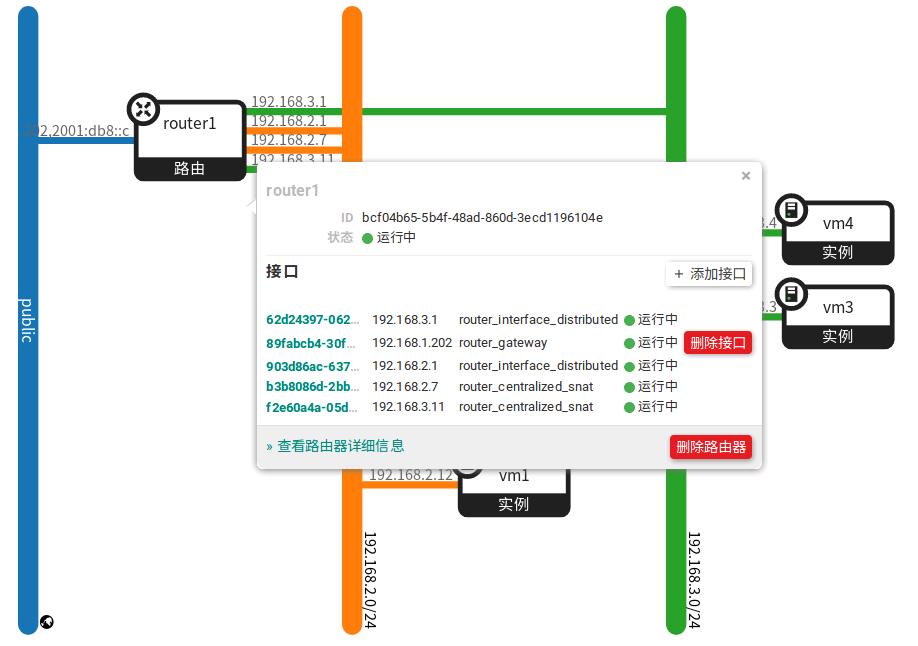
**Environment Setup**

Openstack\_pike: one controller node(network and storage) and two compute nodes

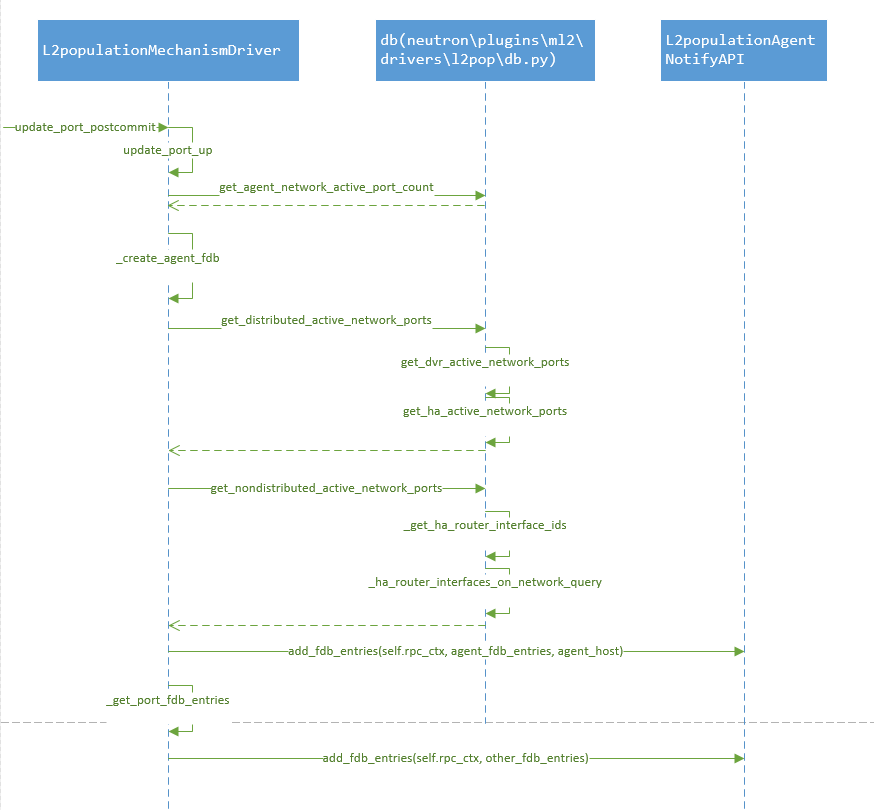
|  |  |
| --- | --- |
| Controller Node( Network and storage) | **/etc/neutron/neutron.conf**  enable\_dvr = true  router\_distributed = true  l3\_ha = true  **/etc/neutron/plugins/ml2/ml2\_conf.ini**  mechanism\_drivers = openvswitch,l2population  l2\_population = true  arp\_responder = true or false  enable\_distributed\_routing = true\  **/etc/neutron/l3agent.ini**  agent\_mode = dvr\_snat |
| Compute Node | **/etc/neutron/plugins/ml2/ml2\_conf.ini**  l2\_population = true  arp\_responder =true or  false  enable\_distributed\_routing = true  **/etc/neutron/l3agent.ini**  interface\_driver = openvswitch external\_network\_bridge = agent\_mode = dvr |

**Network Topology**





**Call Flow**



**Test**

**verify:**

In the compute node:

sudo ovs-appctl fdb/show br-int | grep <csnat port’s mac address>

sudo ovs-ofctl dump-flows br-tun | grep <csnat port’s mac address>

in the vm:

ping <router’s gateway>

|  |  |
| --- | --- |
| **Test Scenario** | **Result** |
| 1. set arp\_responder true or false  2. create network  3. create subnet  4. add subnet to router  5. create VM | 1. Ping gateway successfully  2. After add subnet to router, the flow exists |
| 1. set arp\_responder true or false  2. create network  3. create subnet  4. create VM  5. add subnet to router | 1. Ping gateway successfully  2. After add subnet to router, the flow exists |
| 1. set arp\_responder true or false  2. create network  3. create subnet  4. add subnet to router  5. add a new host to openstack  6. create VM(allocated to the new host) | 1. Ping gateway successfully  2. Before ping gateway, the flow doesn’t exist. After ping gateway, the flow exists. |

**Analysis**

Example:

VM 192.168.2.4 router\_centralized\_snat 192.168.2.7

router\_interface\_distributed 192.168.2.1 DHCP port 192.168.2.2

Router Gateway 192.168.1.202

Ping router gateway:

192.168.2.4 ------> 192.168.2.1 -------> 192.168.2.7 ------> 192.168.1.202

compute-1             compute-1 controller                 controller

Analysis:

    The function \_create\_agent\_fdb will be executed on the condition that in some host there is only one port within a specific network. Then the function \_create\_agent\_fdb will create fdbs and send the fdbs to the host. OVS agent or Linux Bridge agent on the host will receive the fdbs and add flows or fdbs.

    However in the environment that DVR, L2POP and L3HA enabled, the fdb for centralized snat port will not be sent to the host. Then the ovs agent won’t know the destination of the outgoing packet. Maybe avs agent will just drop the packet. This need WR to confirm. However, the OVS agent won’t drop the packet and will flood it. Thus in the third scenario, though the ovs agent does’t know the destination, it will flood it and finally learn the flow. So the bug is not valid for neutron upstream.

**Summary**

* Environment: DVR(Distributed Virtual Router) + L3 HA(High availability) + L2POP(Layer 2 population)
* In all test cases the gateway can be accessed.
* For the outgoing packet that destination unknown, avs agent will drop it while ovs agent will flood it and finally learn the flow. So the bug should be valid only with avs. Need WR to confirm.