# Vxlan Network

## Network Framework



## Linux Network Device

bridge: Bridge is a piece of software used to unite two or more network segments. A bridge behaves like a hub, working transparently (the other machines don't need to know or care about its existence).

**Tun/tap:** Are a feature offered by Linux (and probably by other UNIX-like operating systems) that can do userspace networking, that is, allow userspace programs to see raw network traffic (at the ethernet or IP level) and do whatever they like with it.

br-int: bridge-integration. A typical Openstack deployment is using one instance of the OVS,  br-int, as the point to connect all VMs, DHCP servers and the “non default gateway” side of all routers. br-int is using classic Vlans to separate the broadcast domains.

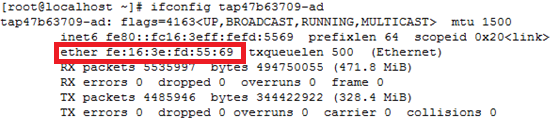
BR-TUN: An OVS instance, and is used to provide the VXLAN function. br-tun is connected to br-int via an internal link. This links is a trunking port, it is using dot1q tagging to transport vlan ids.

BR-EX: bridge-external. I used to connect to external network.

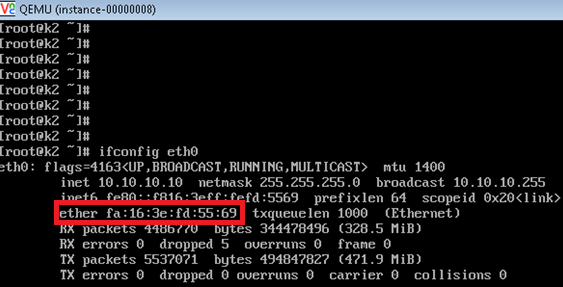
EVTH: Virtual Ethernet device. If a packet is sent to one device it will come out the other device.It used to connect to ovs switch each other.

## Compute Node

The mac address of tap device on compute node

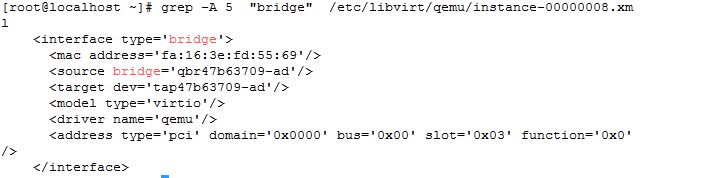


The mac address of instance on compute node

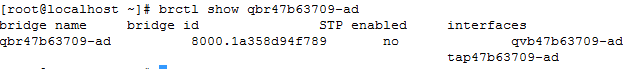


These mac addresses are same. We can understand that the packet from instance’s eth0 will send to tap device.

In instance configure file, the vm connects to linux bridge “qbr47b63709-ad”.

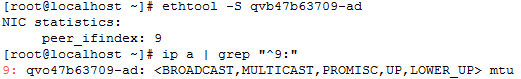


Now we take a look at linux bridge “qbr47b63709-ad”.

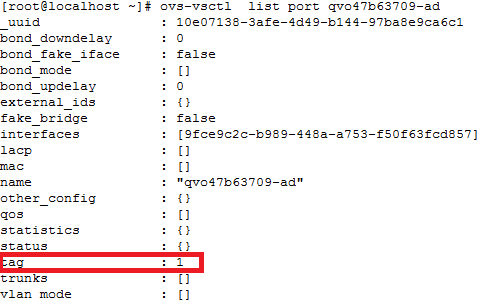


Because the linux bridge is a layer two network device, so the packet received by interface “tap47b63709-ad” will send to other interfaces. Interface “qvb47b63709-ad” receive packet too.

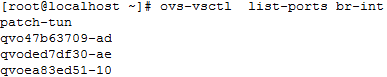
Take a look the interface “qvb47b63709-ad”.its peer interface index is 9.And then find out the number 9 is interface “qvo47b63709-ad”.So interface “qvb47b63709-ad” and“qvo47b63709-ad” is a veth pair.



The packet received by interface “qvo47b63709-ad”, it will be attached a vlan tag “1”.



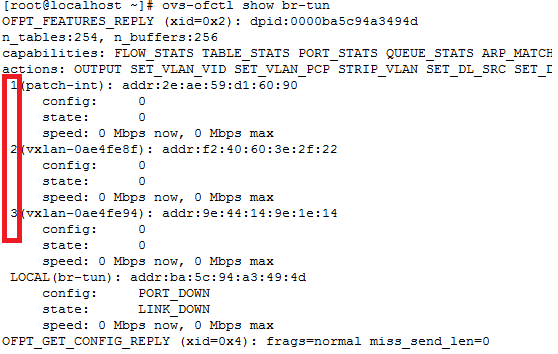
The interface “qvo47b63709” belongs to ovs switch “br-int”.



After the packet received by ovs “br-int”,it will be sent from interface “patch-tun”.



The interface “patch-tun” on ovs switch “br-int” connects to interface “patch-int” on ovs switch “br-tun”.So the ovs switch “br-tun” received packet. It will search the openflow rules, but before that,it will confirm the ports using the command “ovs-ofctl show br-tun”.



Ovs switch “br-tun” received packet from port=1 ”patch-int”.

[root@localhost ~]# ovs-ofctl dump-flows br-tun

cookie=0x0, duration=594639.324s, table=0, n\_packets=925914, n\_bytes=39691148, idle\_age=0, hard\_age=65534, priority=1,in\_port=1 actions=resubmit(,2)

Any packer entering port 1 is resubmit to table 2.

cookie=0x0, duration=594639.107s, table=2, n\_packets=904951, n\_bytes=38018140, idle\_age=0, hard\_age=65534, priority=0,dl\_dst=01:00:00:00:00:00/01:00:00:00:00:00 actions=resubmit(,22)

The packet is a multicast or broadcast will resubmit to table 22.

cookie=0x0, duration=594639.177s, table=2, n\_packets=20963, n\_bytes=1673008, idle\_age=36, hard\_age=65534, priority=0,dl\_dst=00:00:00:00:00:00/01:00:00:00:00:00 actions=resubmit(,20)

The packet is a unicast will resubmit to table 20.

cookie=0x0, duration=594638.787s, table=20, n\_packets=1200, n\_bytes=92460, idle\_age=166, hard\_age=65534, priority=0 actions=resubmit(,22)

Other packets will resubmit to table 22.

cookie=0x0, duration=386.147s, table=20, n\_packets=18, n\_bytes=1236, hard\_timeout=300, idle\_age=36, hard\_age=36, priority=1,vlan\_tci=0x0001/0x0fff,dl\_dst=fa:16:3e:a4:d1:61 actions=load:0->NXM\_OF\_VLAN\_TCI[],load:0xb->NXM\_NX\_TUN\_ID[],output:2

cookie=0x0, duration=166.920s, table=20, n\_packets=1, n\_bytes=42, hard\_timeout=300, idle\_age=161, hard\_age=161, priority=1,vlan\_tci=0x0001/0x0fff,dl\_dst=fa:16:3e:39:b1:3b actions=load:0->NXM\_OF\_VLAN\_TCI[],load:0xb->NXM\_NX\_TUN\_ID[],output:2

Drop the vlan tag, and attach the tunnel id, and then send from port 2 vxlan-0ae4fe8f

cookie=0x0, duration=594634.929s, table=22, n\_packets=906119, n\_bytes=38107964, idle\_age=0, hard\_age=65534, dl\_vlan=1 actions=strip\_vlan,set\_tunnel:0xb,output:2,output:3

The packet attached vlan id 1, it will be drop vlan id ,and attached tunnel id, and then sent from port3 vxlan-0ae4fe94

The above openflow rules are created for the packet from instance will send out compute.Now we take a look about openflow rules about the packets from outside will send to instance.

[root@localhost ~]# ovs-ofctl dump-flows br-tun

cookie=0x0, duration=405073.948s, table=0, n\_packets=0, n\_bytes=0, idle\_age=65534, hard\_age=65534, priority=1,in\_port=3 actions=resubmit(,4)

Any packer entering port 3 vxlan-0ae4fe94 is resubmit to table 4.

cookie=0x0, duration=417995.823s, table=0, n\_packets=3723, n\_bytes=171254, idle\_age=10, hard\_age=65534, priority=1,in\_port=2 actions=resubmit(,4)

Any packer entering port2 vxlan-0ae4fe8fis resubmit to table 4.

cookie=0x0, duration=596437.976s, table=4, n\_packets=14578, n\_bytes=975752, idle\_age=10, hard\_age=65534, priority=1,tun\_id=0xb actions=mod\_vlan\_vid:1,resubmit(,10)

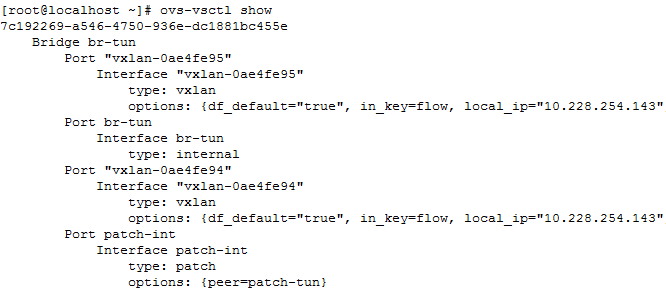
Packet from tunnel, it will be attached vlan tag 1,and then resubmit to table 10.

cookie=0x0, duration=596441.982s, table=10, n\_packets=14578, n\_bytes=975752, idle\_age=10, hard\_age=65534, priority=1 actions=learn(table=20,hard\_timeout=300,priority=1,NXM\_OF\_VLAN\_TCI[0..11],NXM\_OF\_ETH\_DST[]=NXM\_OF\_ETH\_SRC[],load:0->NXM\_OF\_VLAN\_TCI[],load:NXM\_NX\_TUN\_ID[]->NXM\_NX\_TUN\_ID[],output:NXM\_OF\_IN\_PORT[]),output:1

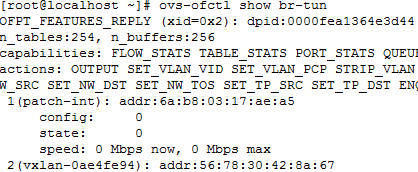
Table 10 learn from table 20. If the vlan id is 0-11，drop the vlan id, drop tunnel id, send to port 1 patch-int .

## Network Node

The interface on network node received packer is port vxlan-0ae4fe94.



The port number of port vxlan-0ae4fe94 is port 2.



Take a look openflow rules about ovs switch “br-tun” .

[root@localhost ~]# ovs-ofctl dump-flows br-tun

NXST\_FLOW reply (xid=0x4):

cookie=0x0, duration=409947.847s, table=0, n\_packets=0, n\_bytes=0, idle\_age=65534, hard\_age=65534, priority=1,in\_port=2 actions=resubmit(,4)

Any packer entering port 2 vxlan-0ae4fe94is resubmit to table 4.

cookie=0x0, duration=418602.980s, table=2, n\_packets=51, n\_bytes=4866, idle\_age=65534, hard\_age=65534, priority=0,dl\_dst=01:00:00:00:00:00/01:00:00:00:00:00 actions=resubmit(,22)

The packet is a multicast or broadcast will resubmit to table 22.

cookie=0x0, duration=418603.061s, table=2, n\_packets=3698, n\_bytes=168203, idle\_age=552, hard\_age=65534, priority=0,dl\_dst=00:00:00:00:00:00/01:00:00:00:00:00 actions=resubmit(,20)

The packet is a unicast will resubmit to table 20.

cookie=0x0, duration=418597.968s, table=4, n\_packets=629619, n\_bytes=26689542, idle\_age=3, hard\_age=65534, priority=1,tun\_id=0xb actions=mod\_vlan\_vid:1,resubmit(,10)

Packet from tunnel id 0xb will attached vlan tag 1,and then resubmit to table 10.

cookie=0x0, duration=418540.372s, table=4, n\_packets=0, n\_bytes=0, idle\_age=65534, hard\_age=65534, priority=1,tun\_id=0xc actions=mod\_vlan\_vid:2,resubmit(,10)

Packet from tunnel id 0xc will attached vlan tag 2,and then resubmit to table 10.

cookie=0x0, duration=418602.737s, table=10, n\_packets=629619, n\_bytes=26689542, idle\_age=3, hard\_age=65534, priority=1 actions=learn(table=20,hard\_timeout=300,priority=1,NXM\_OF\_VLAN\_TCI[0..11],NXM\_OF\_ETH\_DST[]=NXM\_OF\_ETH\_SRC[],load:0->NXM\_OF\_VLAN\_TCI[],load:NXM\_NX\_TUN\_ID[]->NXM\_NX\_TUN\_ID[],output:NXM\_OF\_IN\_PORT[]),output:1

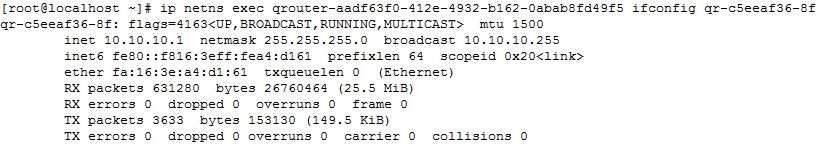
Table 10 learn from table 20. If the vlan id is 0-11，drop the vlan id, drop tunnel id, send to port 1 patch-int .

Ovs switch “br-int” received packet from interface “patch-tun”.

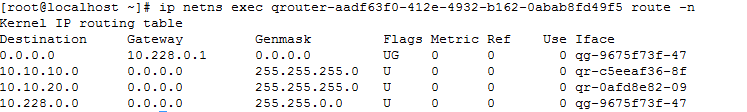


The packet has attached vlan tag 1,So send it to interface “qr-c5eeaf36-8f”.

Interface “qr-c5eeaf36-8f” is an interface from the namespace “qrouter-aadf63f0-412e-4932-b162-0abab8fd49f5”.



If the packet want to arrive external, than the packer will send to interface “qg-9675f73f-47” by namespace “qrouter-aadf63f0-412e-4932-b162-0abab8fd49f5”.



Because the interface “qg-9675f73f-47” belongs to linux bridge “br-ex”, and the linux bridge is a layer two device, an interface received packet will send to other interface, so the packet will send to eth0 .



## Summary

1. On compute, the packet will be attached vlan tag by ovs switch “br-int”.
2. Before the packet out the compute node, remove the vlan tag, and the attach the tunnel id.
3. The packet will map an suitable vlan tag according to the openflow rules by ovs switch “br-tun” on network node .