Module 5: Networking Fundamentals

Agenda

* Basic linux native technologies.
* Neutron working
* Neutron Agents
* Virtual Devices in openstack architecture
* Network Packet Flow.
* Managing Network

Why this session ??

Neutron is a core component

Neutron is a hard to follow component

Understanding is key to troubleshooting

Before we start

Five main concept/linux virtual network devices

- **tap devices**
- **veth paid**
- **Linux bridge**
- **Openvswitch**
- **Network namespaces**

What does neutron do ?

- Allow user to create networks and routers
- Provides an api and manages the database
- Route L3 traffice
- Provide Floating IP’s
- Switches L2 traffic
- Provides DHCP to instances
- Provides metadata to instances

Neutron - Agents

AGENT	FEATURE
L3 agent	Provides external access to the OpenStack environment, including support for floating IP addresses, which allow external access to the instances running on the OpenStack environment.
DHCP agent	Provides support for subnets created using the OpenStack networking service. This agent starts a new DHCP service running in the namespace for each subnet being created. It provides the range of IP addresses this subnet has been created with.
Metadata agent	Provides information services for the instances that are being created to gain information about its configuration.
Open vSwitch agent	Manages the Open vSwitch plug-in for Neutron.

Virtual switches and interfaces

OPEN VSWITCH CONFIGURATION	
br-int	Provides both ingress and egress VLAN tagging for instance traffic.
br-ex	Provides an external bridge and connects to **qg** of the **qrouter** via a tap interface.
qg	Connects the router to the gateway.
qr	Connects the router to the integration bridge **br-int**.
qvo	Connects to the integration bridge.
qvb	Connects the firewall bridge to **br-int** via **qvo**.
qbr	Linux bridge to provide OpenStack security groups.

Packet Flow in Neutron ??

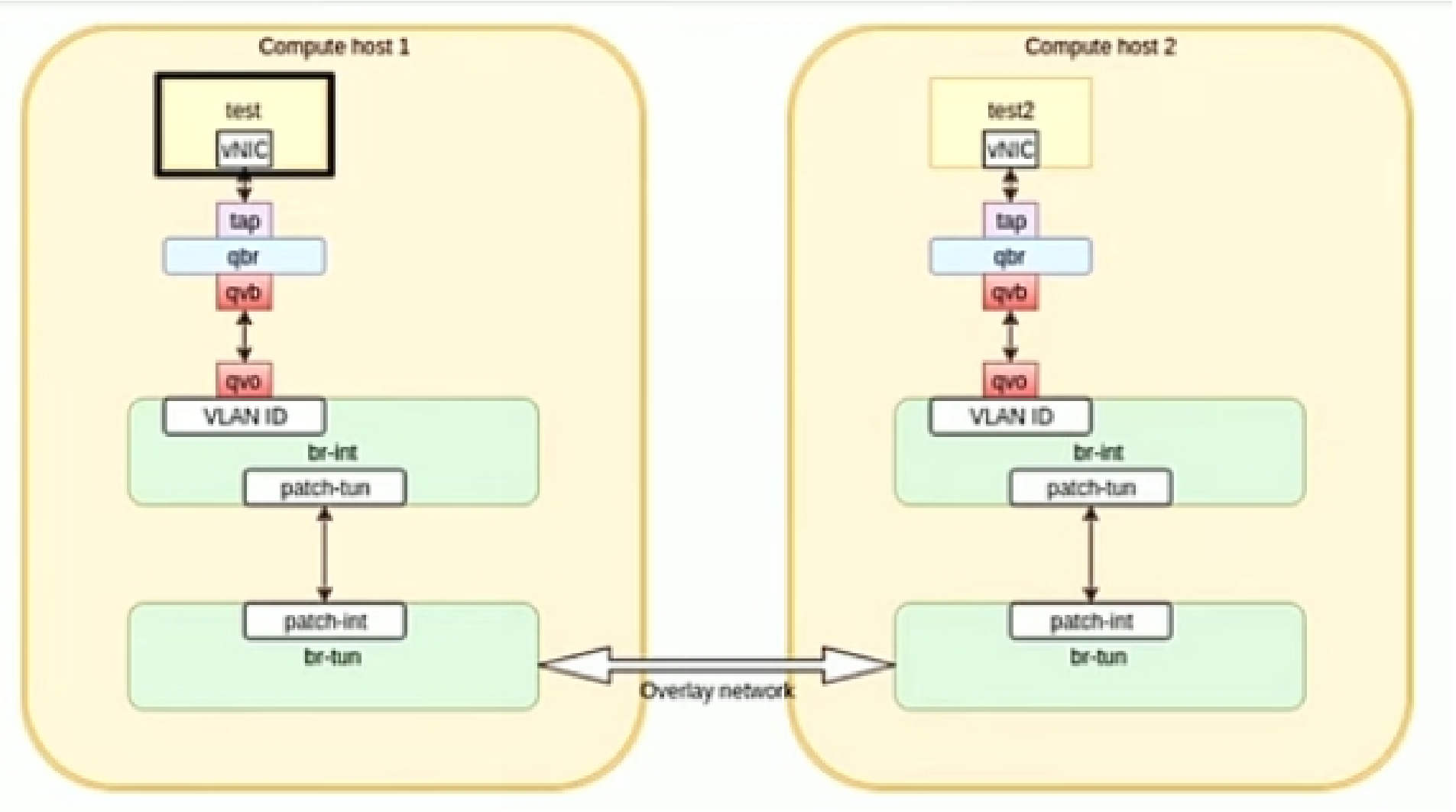
- Instance – Instance, same network
    - Instances living on same compute host
    - Instances living on different host

- Instance - Instance, different Network
- Instance – dhcp
- Instance – Public Network
- End User – Instance using Floating IP

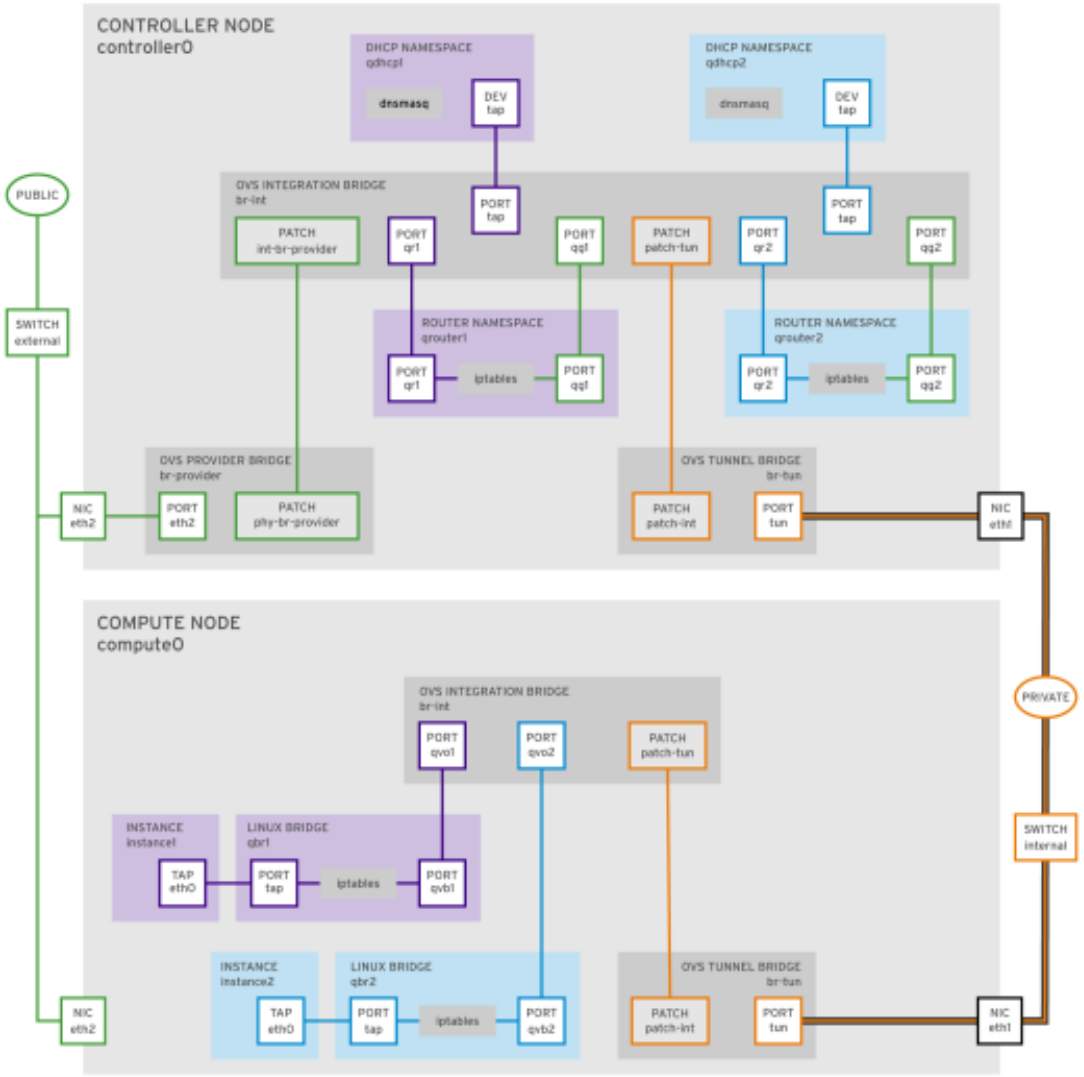
Instance – to – Instance

- > On Same Compute node

Instance to Instance Different Host



Full Flow


CONTROLLER NODE
controller0
DHCP NAMESPACE
qdhcp1
dnsmasq
DEV
tap
DHCP NAMESPACE
qdhcp2
dnsmasq
DEV
tap
PUBLIC
OVS INTEGRATION BRIDGE
br-int
PATCH
int-br-provider
PORT
qr1
PORT
qg1
PATCH
patch-tun
PORT
qr2
PORT
qg2
PORT
tap
PORT
tap
SWITCH
external
ROUTER NAMESPACE
qrouter1
PORT
qr1
iptables
PORT
qg1
ROUTER NAMESPACE
qrouter2
PORT
qr2
iptables
PORT
qg2
OVS PROVIDER BRIDGE
br-provider
OVS TUNNEL BRIDGE
br-tun
NIC
eth2
PORT
eth2
PATCH
phy-br-provider
PATCH
patch-int
PORT
tun
NIC
eth1
COMPUTE NODE
compute0
OVS INTEGRATION BRIDGE
br-int
PORT
qvo1
PORT
qvo2
PATCH
patch-tun
PRIVATE
INSTANCE
instance1
LINUX BRIDGE
qbr1
TAP
eth0
PORT
tap
iptables
PORT
qvb1
SWITCH
internal
INSTANCE
instance2
LINUX BRIDGE
qbr2
TAP
eth0
PORT
tap
iptables
PORT
qvb2
OVS TUNNEL BRIDGE
br-tun
NIC
eth2
PATCH
patch-int
PORT
tun
NIC
eth1