

OpenStack Service Definition

Jing, mqjing@gmail.com

- **Dashboard, Horizon**
 - Provides a web-based self-service **portal** to interact with underlying OpenStack services, such as launching an instance, assigning IP addresses and configuring access controls.
- **Compute, Nova: provisions VM (instances)**
 - Manages the lifecycle of **compute** instances in an OpenStack environment. Responsibilities include spawning, scheduling and decommissioning of machines on demand.
- **Networking, Neutron: provide network connectivity for VM**
 - Enables network connectivity as a service for other OpenStack services, such as OpenStack Compute. Provides an API for users to define networks and the attachments into them. Has a pluggable architecture that supports many popular networking vendors and technologies.
- **Block Storage, Cinder: provide volumes for VM**
 - Provides persistent **block storage** to running instances. Its pluggable driver architecture facilitates the creation and management of block storage devices.
- **Image Service, Glance: provide images for VM (jing: image bank)**
 - Stores and retrieves virtual machine disk images. OpenStack Compute makes use of this during instance provisioning.
- **Object Storage, Swift: provide storage for Glance images**
 - You can store and retrieve lots of data with a simple API.
 - Stores and retrieves arbitrary unstructured **data objects** via a RESTful, HTTP based API. It is highly fault tolerant with its data replication and scale out architecture. Its implementation is not like a file server with mountable directories
- **Identity Service, Keystone: provide auth for all services**
 - Provides an authentication and authorization service for other OpenStack services. Provides a catalog of endpoints for all OpenStack services.
- **Telemetry, Ceilometer: monitor Cinder, Glance, Neutron, Nova**
 - Monitors and meters the OpenStack cloud for billing, benchmarking, scalability, and statistical purposes.
- **Orchestration, Heat**
 - Orchestrates multiple composite cloud applications by using either the native HOT template format or the AWS CloudFormation template format, through both an OpenStack-native REST API and a CloudFormation-compatible Query API.

Ref: Overview

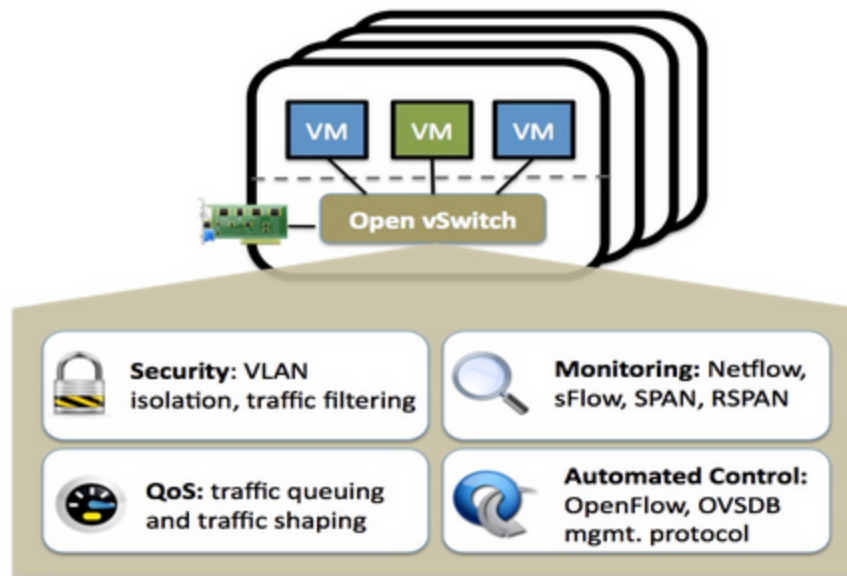
http://docs.openstack.org/trunk/install-guide/install/apt/content/ch_overview.html

- **Control node**
 - runs the **Identity** Service, **Image** Service, **dashboard**, and management portions of Compute and Networking. It also contains the associated API services, MySQL databases, and messaging system.
- **Compute node**

- runs the **hypervisor** portion of Compute, which operates tenant virtual machines. By default, Compute uses KVM as the hypervisor. The compute node also runs the Networking plug-in agent, which operates tenant networks and implements security groups. You can run more than one compute node
- **Network node**
 - runs the Networking plug-in agent and several layer 3 agents that provision tenant networks and provide services to them, including routing, NAT, and DHCP. It also handles external (internet) connectivity for tenant virtual machines

Ref: Arithecture http://docs.openstack.org/trunk/install-guide/install/apt/content/ch_overview.html

- **Open vSwitch (official site)**
 - multilayer virtual switch, Open vSwitch implements an Ethernet **switch**, which means that it is also an Ethernet **bridge**.
 - It is designed to enable massive network automation through programmatic extension, while still supporting standard management interfaces and protocols (e.g. NetFlow, sFlow, SPAN, RSPAN, CLI, LACP, 802.1ag)
 - It is designed to support distribution across multiple physical servers



- **Hypervisor**
 - A **hypervisor** or **virtual machine monitor (VMM)** is a piece of computer software, firmware or hardware that **creates and runs virtual machines**. The hypervisor presents the guest operating systems with a **virtual operating platform** and **manages** the execution of the guest operating systems. [\[ref\]](#)
 - ex:
 - Xen, KVM, Hyper-V, ESX, VMware
- **GRE**
- **Ceph**
 - Ceph is a free software storage platform designed to present object, block, and file storage from a single distributed computer cluster.

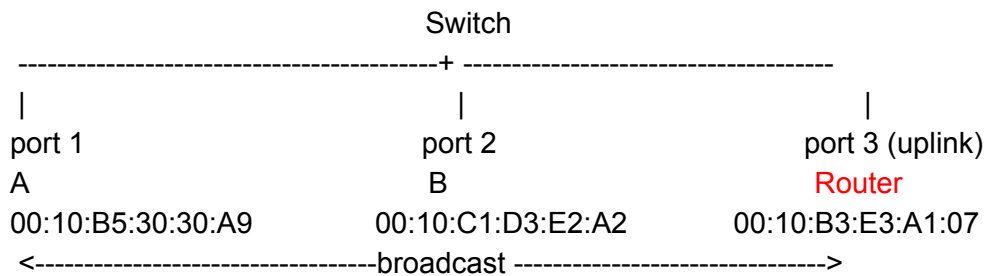
- Highly Available File Server Cluster with a massive storage backend.
- **NUMA**
 - Non-uniform memory access (NUMA) is a **computer memory** design used in **multiprocessing**, where the memory access time depends on the memory location relative to the processor. **Under NUMA, a processor can access its own local memory faster than non-local memory** (memory local to another processor or memory shared between processors). The benefits of NUMA are limited to particular workloads, notably on servers where the data are often associated strongly with certain tasks or users. [\[ref\]](#)
- **VLANs**: Isolating VM traffic using VLANs
 - is a way to partition a single switch into multiple switches.
 - Example that you have **two groups of machines**, group A and group B. You want the machines in group A to be able to talk to each other, and you want the machine in group B to be able to talk to each other, but you don't want the machines in group A to be able to talk to the machines in group B. You can do this with **two switches**, by plugging the machines in group A into one switch and the machines in group B into the other switch
 -
- **sFlow**: Monitoring VM traffic using sFlow
- **QoS Rate-Limiting**: Rate-limiting VMs using QoS policing.
-

Network

- **eth0**: is the first **ethernet device**
- **ppp0**: is the first **point to point protocol device**, usually associated with a modem
- **lo**: is the loopback interface, **127.0.0.1**
- **tap0**: is the first **virtual TUN/TAP ethernet device**
- Host-only networking:
 - the virtual machines can talk to **each other** and **the host** as if they were connected through a physical Ethernet switch. Similarly, as with internal networking however, a physical networking interface need not be present, and the virtual machines **cannot talk to the world** outside the host since they are not connected to a physical networking interface
- Bridge ([bird ref](#): IEEE 802.3: CSMA/CD):
 - **Function**: **filtering** and **forwarding** between **network segments**
 - Allow two or more communication networks, or two or more network segments,[1] to **create an aggregate network** ([wiki](#))
 - A network bridge is a network device
 - Connects multiple network segments. In the OSI model bridging acts in the first two layers, **below the network layer**.
- Router:
 - **Function**: **share** a single Internet connection among **multiple computers** ([ref](#)).

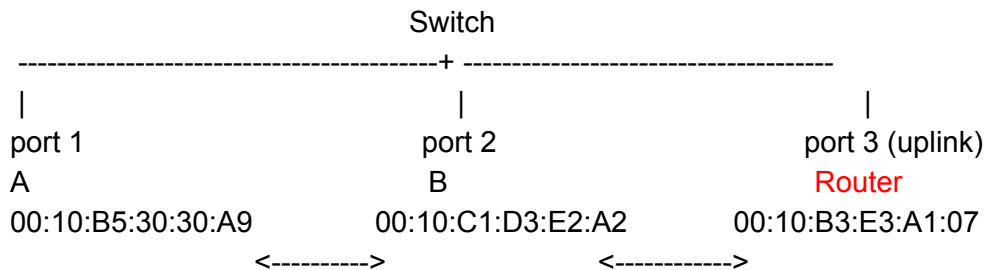
Switches cannot. Switches (level-1) allow you to share a single port on your router (or another switch) among multiple Ethernet devices.

- 透過 Router 連接上 Internet
-
- Switch ([bird ref](#): IEEE 802.3: CSMA/CD)
 - Function: **filtering** and **forwarding** between end-points
 - Layer-1 switch



- Any packet entering a port is **broadcast** out or "**repeated**" on every other port, save the port of entry. Since every packet is repeated on every other port, packet **collisions** result, which slows down the network

- Layer-2 switch



- Handle data based on the **MAC address** in the data packet. ([ref](#))
- Once a bridge **learns the topology** through a spanning tree protocol, it forwards data link layer frames using a layer 2 forwarding method.
- A network **bridge**, operating at the Media Access Control (MAC) sublayer of the data link layer, may **interconnect a small number of devices** in a home or office. This is a trivial case of bridging, in which the bridge learns the **MAC address** of each connected device. Bridges may also interconnect using a **spanning tree protocol** that allows the best path to be found within the constraint that it is a tree.

- Layer-3 switch

- Able to read IP information from packets, and make routing decisions

based on the networks different machines belong to.

- devstack
 - `FIXED_RANGE` and `FIXED_NETWORK_SIZE`
 - configure the `internal` address space used by the instances.
 - `FLAT_INTERFACE`
 - The ethernet interface that `connects the host to your local network`
 - `FLOATING_RANGE`
 - A range `not used` on the your local network
 - `VLAN_INTERFACE`
 - `FLAT_INTERFACE`
 - `PUBLIC_INTERFACE`

References

- Overview, http://docs.openstack.org/trunk/install-guide/install/apt/content/ch_overview.html
- Architecture http://docs.openstack.org/trunk/install-guide/install/apt/content/ch_overview.html
-