GLANCE (**Image Management)**

Let’s imagine, in a company, whenever an IT hardware maintenance employee wants to install or upgrade the operating system/Software in a computer, it request the particular operating system’s/Software’s DVD from the ‘Image management team’.

The ‘Image management team’ first check the authenticity of the employee and only after then, they issue the requested OS/Software’s DVD to the employee. It is the responsibility of the ‘Image management team’, to maintain the image of all the OS/Software, that may be requested by an employee.

The ‘Image management team’ will act as an ISO image repository for the IT hardware

allocation team and will provide the selected OS for the IT hardware allocation team to

install the operating system in the assembled bare metal.

The purpose of the image management team is to maintain the repository for the necessary

operating system and to keep the metadata records for each OS bundle. Metadata is the register that maintains the minimal hardware requirements to run the selected OS. For example, the minimum hardware requirements for the Ubuntu 14.04 LTS server OS is 1GB

RAM/8GB HDD and similarly 4GB RAM/40GB HDD for Windows 7 OS. In any case, if the

minimal resource requirement for any OS is not met, then the access to that selected OS will

be repudiated for the IT hardware allocation team, and the same will be reported to the user.

Similarly, the glance service serves as an image management team for the Nova service in

OpenStack. The job of the glance service is to manage the OS image repository for the user.

In simple words, similar to a DVD pouch in our home that contains different operating system images in an organized way to access whenever we install a new OS on our machine.

Whenever the Nova service requests the OS image bundle, the Glance image service will first contact the KeyStone service with the user's token ID to check whether the user is authorized to access the image service or not. Once all of the verification including minimal hardware requirements are passed, the glance service will provide the requested OS image bundle to the Nova service to begin the OS installation.

Unlike a typical ISO, cloud images that exist in Glance are typically snapshots of a disk's contents. These images have been previously configured by a person or script that has gone through the initial installation procedure and has installed specific programs and configuration files to ensure it is cloud aware.

Glance can store these images in a variety of backends called **data stores**. Glance supports a variety of data stores, including the local filesystem, NFS, or an OpenStack Swift container.

**Image file formats**

Disk images store all the contents of a hard drive and come in a variety of file

formats. Here are a few of the most popular options:

* **QCOW2 (QEMU Copy-On-Write)**: Used with the QEMU and QEMUKVM

hypervisors and named after their disk storage optimization strategies, which delay allocation of storage until actually needed.

* **VMDK (Virtual Machine Disk)**: Initially designed to be used with VMware's line of hypervisors but now an open format compatible with other non-VMware hypervisors such as VirtualBox.
* **VHD and VHDX (Virtual Hard Drive)**: Originally created by the Connectix Corporation for the Type-2 hypervisor *Virtual PC*, acquired by Microsoft in 2003. It's commonly used with Microsoft-related products and services, such as Hyper-V and Azure.
* **VDI (Virtual Disk Image)**: Used by Oracle's VirtualBox.
* **ISO (International Organization for Standardization, ISO9660)**: An archive file of an optical disc. It is composed of the data contents from every written sector on an optical disc, including the optical disc filesystem.
* **RAW**: An uncompressed disk image. Nova will uncompress any compressed cloud image and place it into raw format at the time of booting a virtual machine instance.
* **AKI**: An Amazon kernel image.
* **AMI** : An Amazon machine image.
* **ARI** : An Amazon ramdisk image.

Although we use Glance to register, store, and retrieve images, it does not care about the type of data it stores. In fact, Glance will store a .txt, .doc, or any other file type you throw at it. It's up to the OpenStack administrator to ensure Glance images are compatible with the hypervisor (or hypervisors) deployed in the environment.

**Glance architecture**

Glance is comprised of two primary daemons:

1. **glance-api**: The primary gateway to Glance. One must interact with glance-api

to store and retrieve disk images.

2. **glance-registry**: Responsible for storing metadata associated with the image in the relational database. Examples of image metadata include the image name, image location, UUID, image size, owner (project ID), availability status, and disk format.

Glance can store images in a variety of data stores, including Swift, Amazon S3, the local filesystem on which the glance-api daemon resides, or even a publicly accessible web server.

**Openstack CLI**

**NOVA compute**

**Openstack**

**Horizon**

Glance API

**KeyStone**

**Storage plugins**

**Glance Registry**

**Http**

**File System**

**Ceph**

**Swift**

**Database**

*Architecture of Glance*

As you can also see from the Figure, Glance itself does not store images by themselves. Glance uses plug-ins for particular storage, which can be your local file system, Swift object storage, Ceph storage, NFS (Network File System), or other back ends. Metadata of images are stored in the Glance database, usually as a MariaDB instance.

Let’s check these services on the OpenStack controller:

# systemctl | grep glance

openstack-glance-api.service

loaded active running OpenStack Image Service (code-named Glance) API server

openstack-glance-registry.service

loaded active running OpenStack Image Service (code-named Glance) Registry server

As you can see, both services are up and running.

Usually when the Compute service Nova is trying to spawn a new virtual machine, it sends a GET request to the URL http://path\_to\_Glance\_service/images/paticular\_image\_ID . If the glance-api finds the requested image, the service will return the URL where the image is located. After that, Nova sends the link to the hypervisor’s driver and hypervisor will download the image directly.

Then you need to look through the main configuration files for the glance-api and glance-registry. Both /etc/glance/glance-api.conf and /etc/glance/glance-registry.conf contain similar settings.

Deploying an image:-

First, you need to download the image:

$ wget -P /tmp http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x86\_64-disk.img

Then you can deploy the image in the cloud:

$openstack image create --file /tmp/cirros-0.3.4-x86\_64-disk.img --disk-format qcow2 --container-format bare --public cirros-0.3.4-x86\_64

To get a list of all available images for the current user, use the following:

$openstack image list

**Command Purpose of Command**

openstack image create Create/upload an image

or glance image-create

openstack image delete Delete image(s)

or glance image-delete

openstack image add project Associate project with an image

or glance member-create

openstack image remove project Disassociate project with an image

or glance member-delete

openstack image list List available images

or glance image-list

openstack image save Save an image locally on disk

or glance image-download

openstack image show Display image details

or glance image-show

openstack image set Set image properties

glance image-update Set image metadata

**Managing Image Back Ends**

Glance can support various data store back ends, such as Swift, Ceph, NFS, local file system, and others. Storage vendors like EMC or NetApp produce plug-ins for their own hardware. You can define each particular back end in the [glance\_store] section of the configuration file /etc/glance/glance-api.conf .

Here is the simplest example of the local file system:

[glance\_store]

...

default\_store = file

filesystem\_store\_datadir = /var/lib/glance/images/

If you look at this directory, you can find files with the names that are equal to the image’s ID:

# ls -l /var/lib/glance/images/

total 329080

-rw-r----- 1 glance glance 13287936 Mar 12 21:25 e5791edb-30dd-475a-9bc4-5938341db655

-rw-r----- 1 glance glance 323682816 Mar 12 21:41 f42295b8-d600-4a67-86b7-dcda07652db4

ls -l /var/lib/glance/images/

$ openstack image list

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

| ID | Name |

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

| f42295b8-d600-4a67-86b7-dcda07652db4 | ubuntu-amd64 |

| e5791edb-30dd-475a-9bc4-5938341db655 | cirros-0.3.4-x86\_64 |

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

Glance can serve multiple back ends at the same time. In this case Glance will choose a particular back end depending on the free space and priority. For example, if you have two mounted disks in /var/lib/glance/images/ , you can add something like this:

[glance\_store]

filesystem\_store\_datadirs = /var/lib/glance/images/mountA/:10

filesystem\_store\_datadirs = /var/lib/glance/images/mountB/:20

...

To limit the size of images, you need to add the image\_size\_cap parameter and maximum size in bytes to the glance-api configuration file and restart the glance-api service. Here is an example for adding a 1GB parameter:

[default]

image\_size\_cap = 1073741824

...

# systemctl restart openstack-glance-api

If you need to limit the storage amount per user in Glance, use another option:

[default]

user\_storage\_quota = 500MB

...