

Tutorial : Working With Open Stack

Dr. Bibhas Ghoshal

Assistant Professor

Department of Information Technology

Indian Institute of Information Technology Allahabad



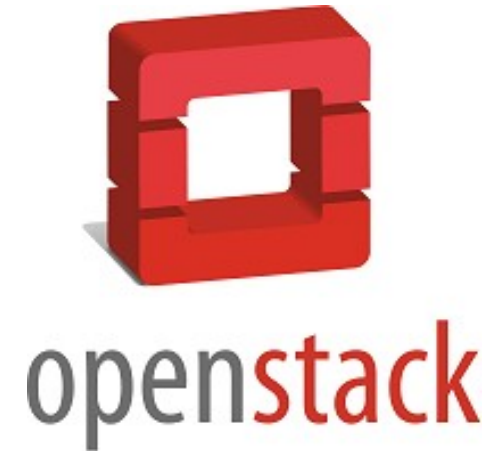
Cloud and Edge Computing
Instructor : Dr. Bibhas Ghoshal

Spring 2023

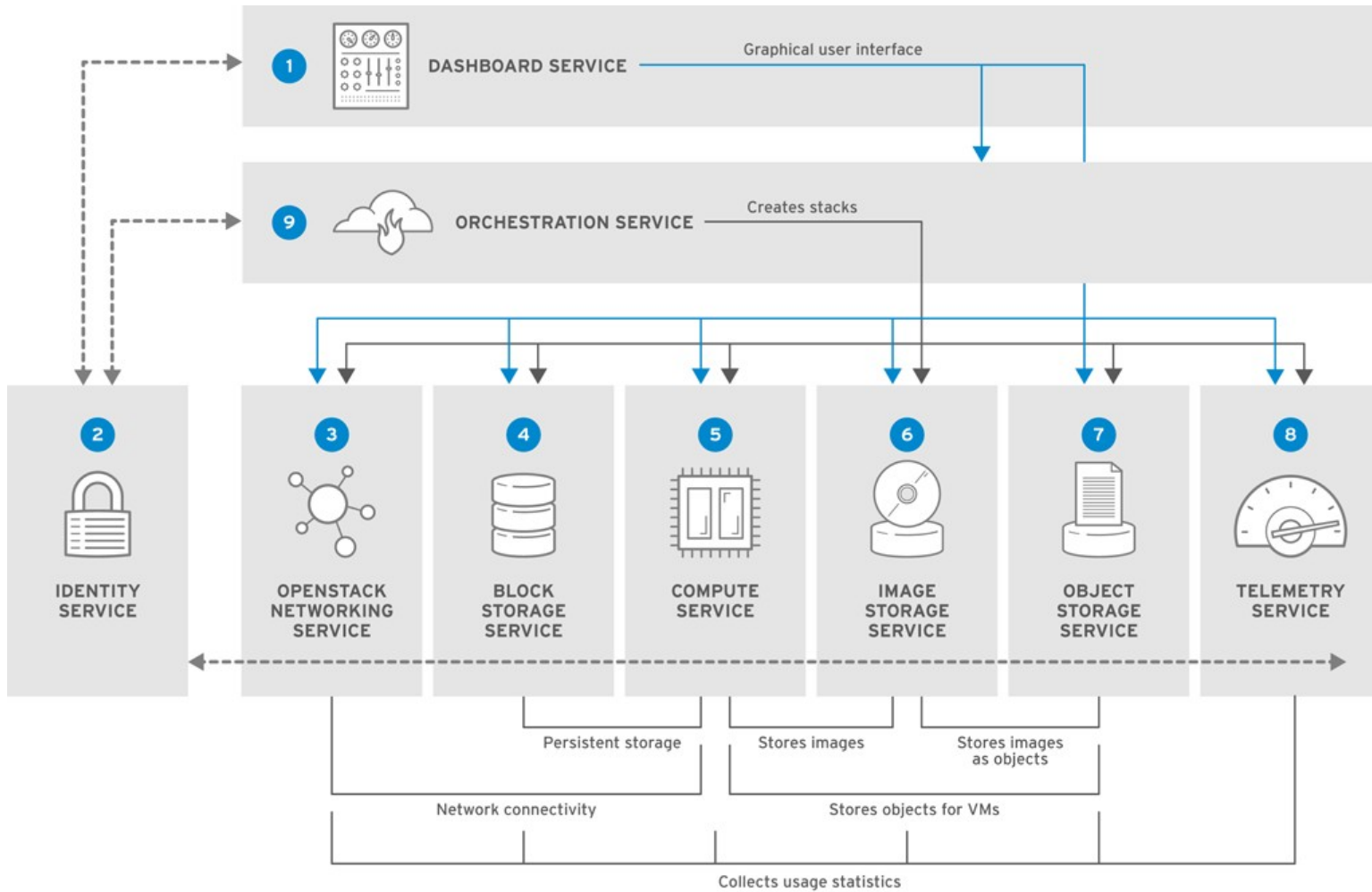
Open Stack

Software platform (Cloud OS) that is used to :

- control the large pools of computing, storage, and networking resources within a data center (private and public clouds)
- OpenStack Platform **IaaS** cloud is implemented as : collection of interacting services that control compute, storage, and networking resources.
- Cloud can be managed with a web-based dashboard or command-line clients, which allow administrators to control, provision, and automate OpenStack resources.
- OpenStack also has an extensive API, which is also available to all cloud users.
- **Free and Open source Software**
- **Huawei, Intel, Red Hat** have adopted **OpenStack** for their cloud management



Open Stack Core Services



RHELOSP_347192_1015



Open Stack Core Services

Service	Code	Description
Dashboard	Horizon	Web browser-based dashboard that you use to manage OpenStack services.
Identity	Keystone	Centralized service for authentication and authorization of OpenStack services and for managing users, projects, and roles.
OpenStack Networking	Neutron	Provides connectivity between the interfaces of OpenStack services.
Block Storage	Cinder	Manages persistent block storage volumes for virtual machines.
Compute	Nova	Manages and provisions virtual machines running on hypervisor nodes.



Open Stack Core Services

Service	Code	Description
Image	Glance	Registry service that you use to store resources such as virtual machine images and volume snapshots.
Object Storage	Swift	Allow users to store and retrieve files and arbitrary data
Telemetry	Ceilometer	Provides measurements of cloud resources.
Orchestration	Heat	Template based Orchestration

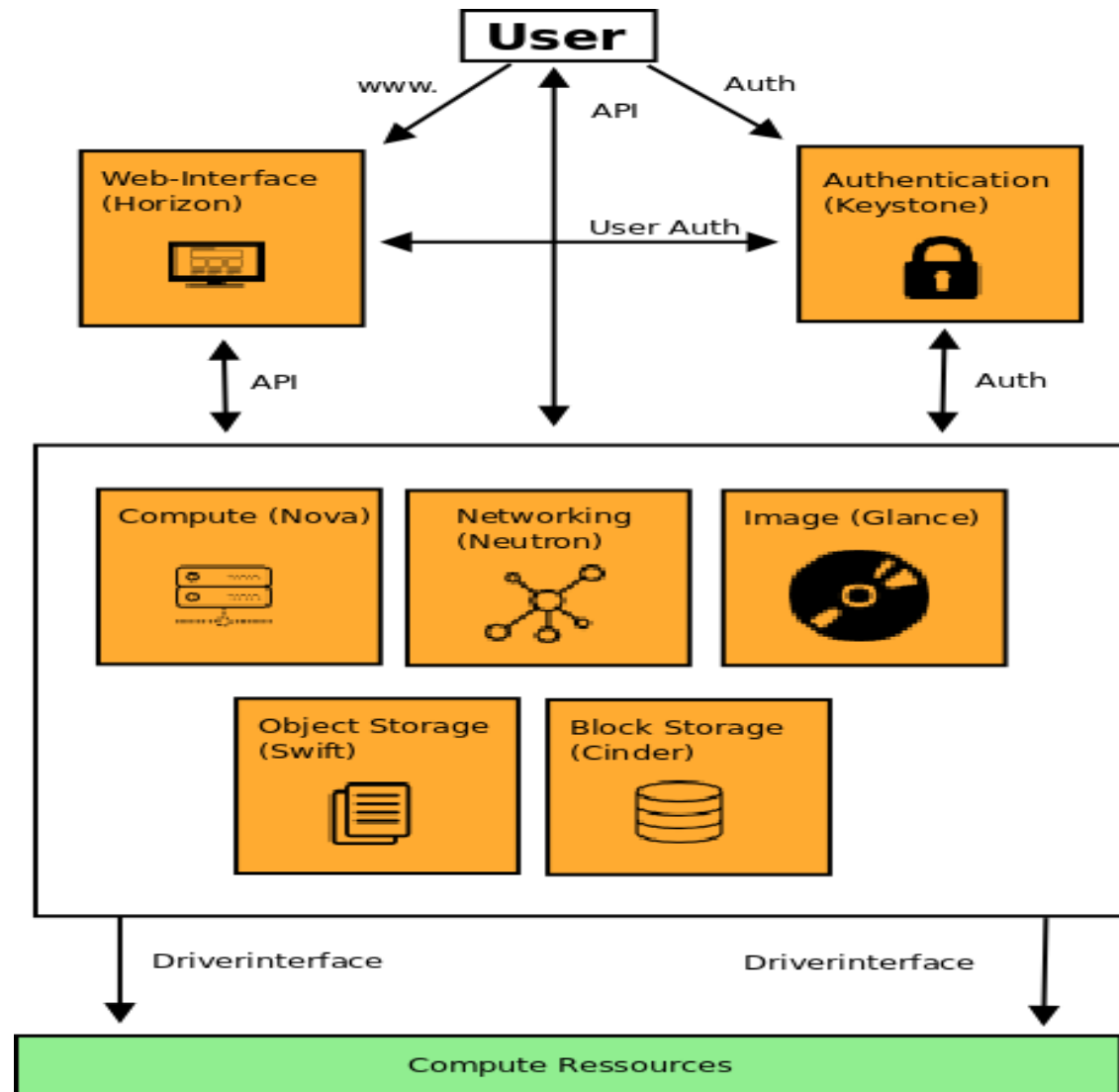


Open Stack Additional Services

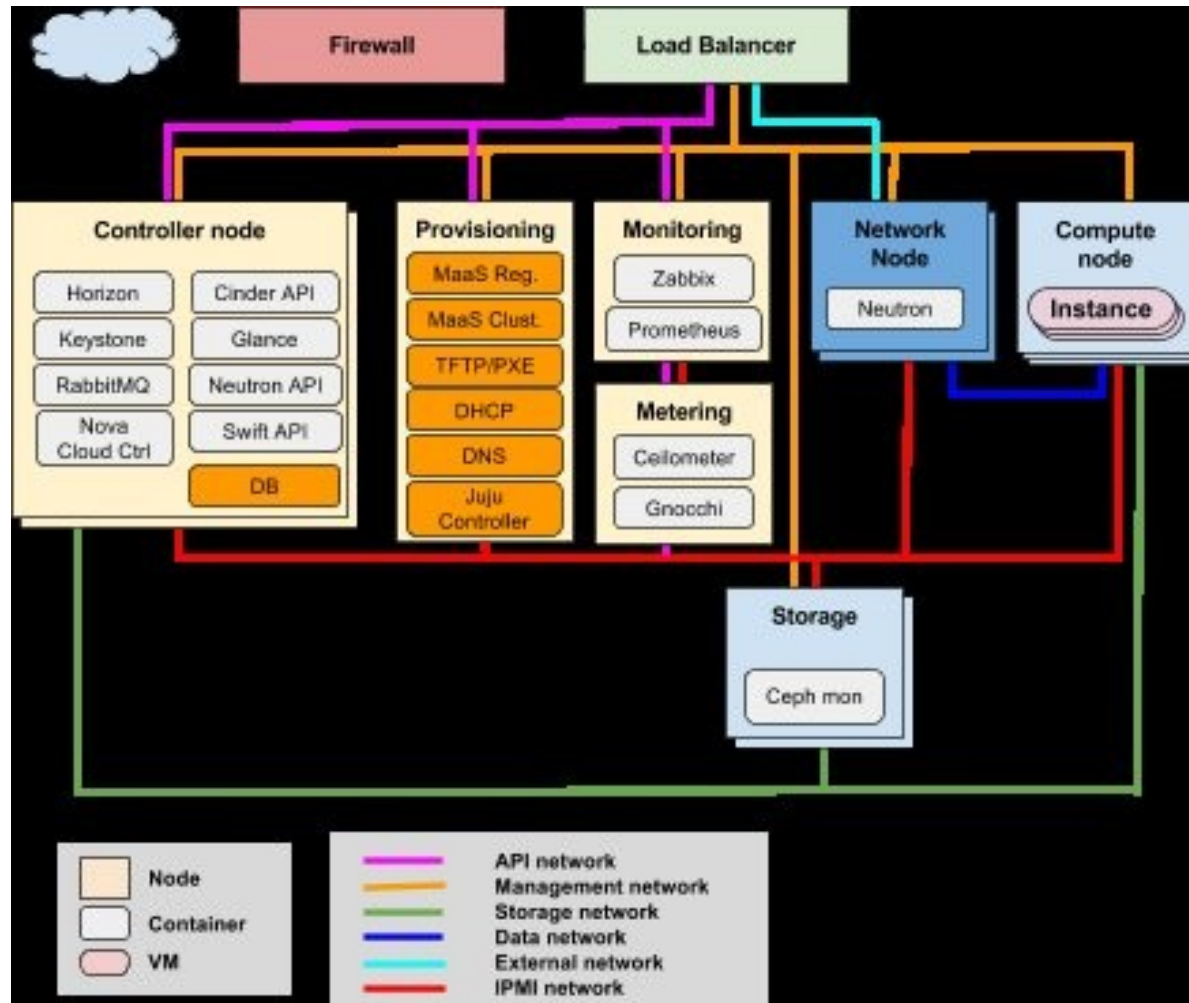
Service	Code	Description
Bare metal provisioning	Ironic	Enables users to provision physical machines (bare metal) with a variety of hardware vendors
Database as a service	Trove	Allow users to deploy relational and non-relational database engines and handles complex database administration tasks
Data Processing	Sahara	Allows users to provision and manage Hadoop clusters on OpenStack.



Open Stack Core Services



Open Stack Reference Architecture



Open Stack Deployment Models

Public	Private Cloud	Private Cloud As a Service
Off Premises	On Premises	On or Off Premises
Shared Infrastructure	Dedicated Infrastructure	Deployment and upgrades by vendor
Operated by vendor	Operated by customer	Operated by vendor



Learning Open Stack



Resources to Learn Open Stack

- Open Stack Foundation

<http://www.openstack.org>

- Official OpenStack Documentation

<http://docs.openstack.org>

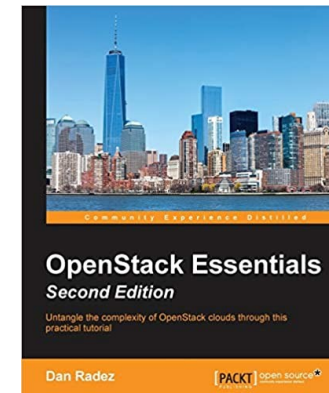
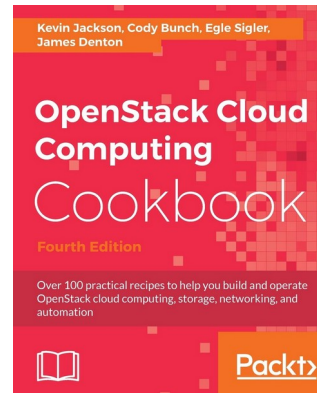
- The Openstack Cloud Computing Cookbook

<https://www.packtpub.com/virtualization-and-cloud/openstack-cloud-computing-cook-book-fourth-edition>

- OpenStack essentials

<https://www.packtpub.com/virtualization-and-cloud/openstack-essentials>

- *Open Stack Community* : *<http://www.openstack.org/community/>*



Deploying Open Stack

- Open Stack Installation Guide

<https://docs.openstack.org/install-guide/index.html>

- OpenStack Deployment using MicroStack

<https://discourse.ubuntu.com/t/get-started-with-microstack/13998>

Install MicroStack from the beta channel (For Ubuntu 18.04+) :

```
$ sudo snap install microstack --devmode --beta
```

When the installation process has finished you should see the following message on the terminal :

```
$ microstack (beta) ussuri from Canonical ✓ installed
```

Initialize MicroStack :

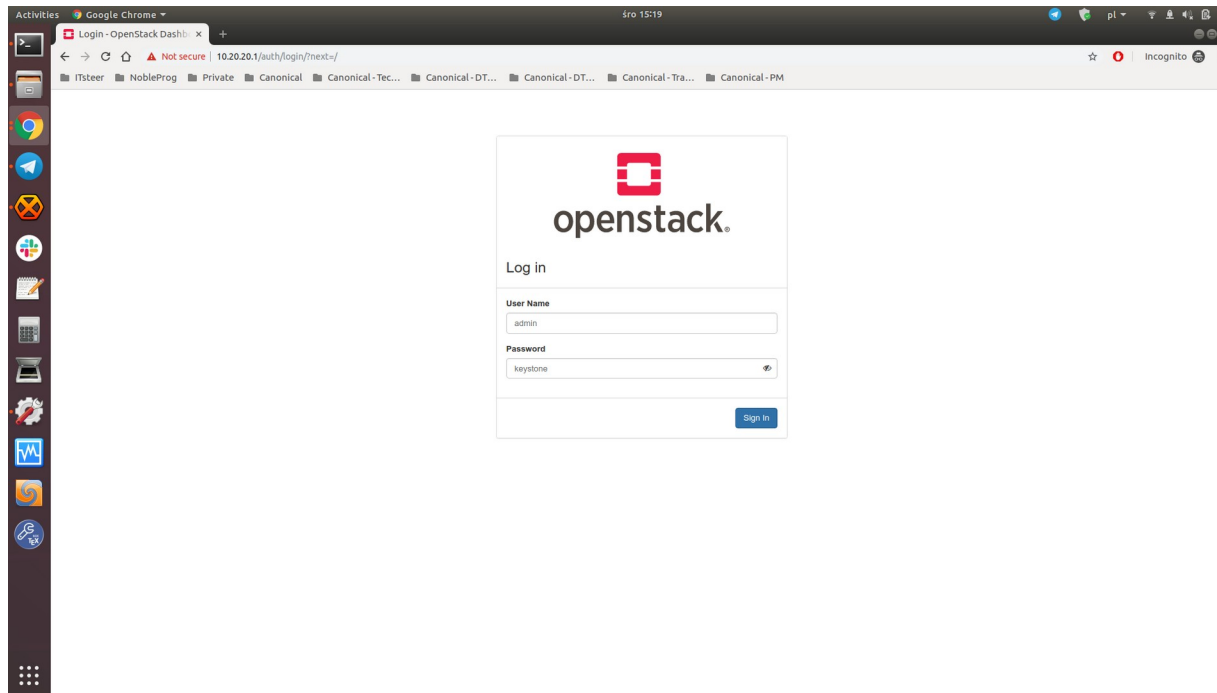
```
$ sudo microstack init --auto --control
```



Interacting with Open Stack

Interact via Web UI : *\$ http://10.20.20.1/*

Obtain password for admin user : *\$ sudo snap get microstack config.credentials.keystone-password*

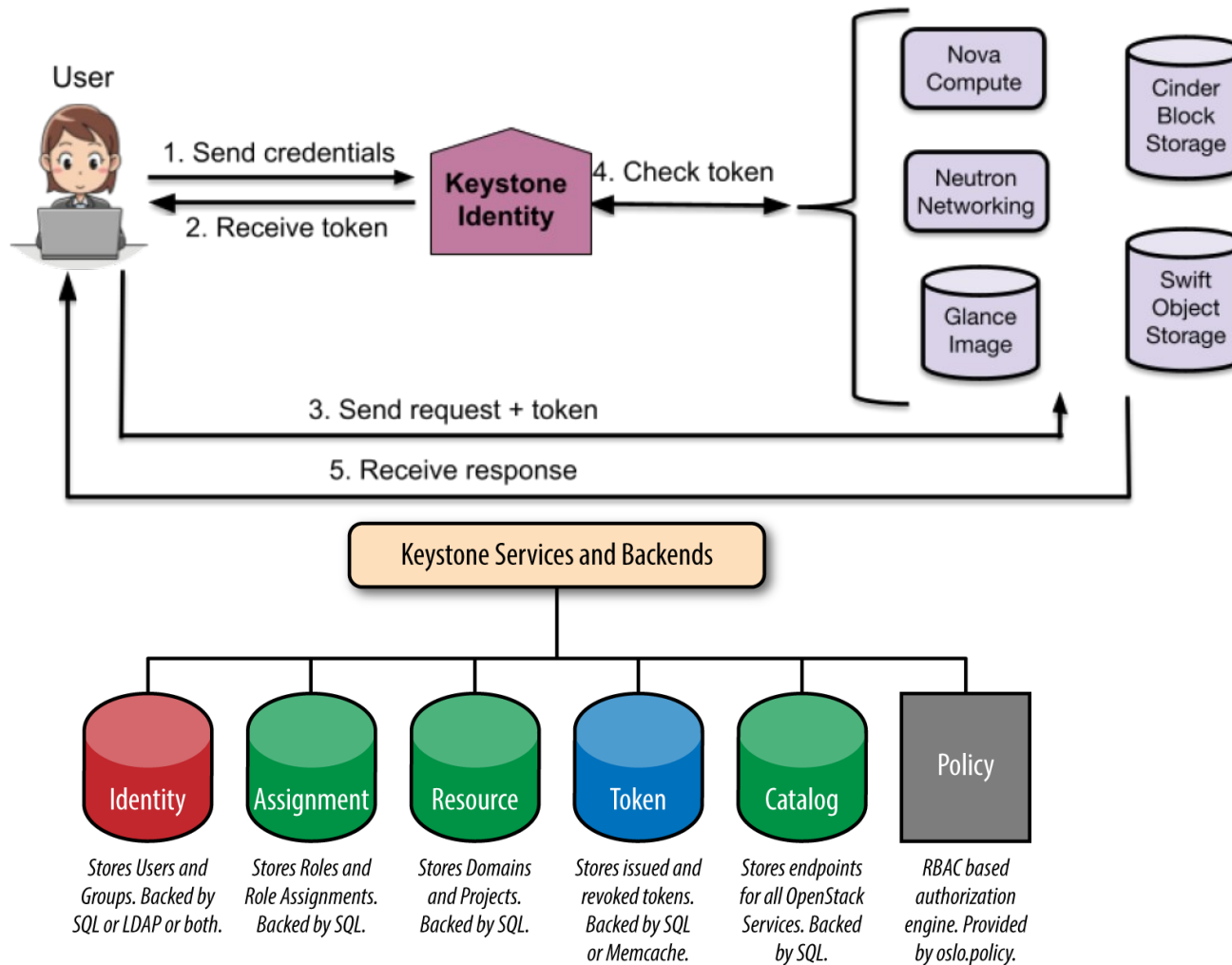


- Login using the password provided
- **Keystone OpenStack service** comes in play - authentication and authorization of OpenStack services

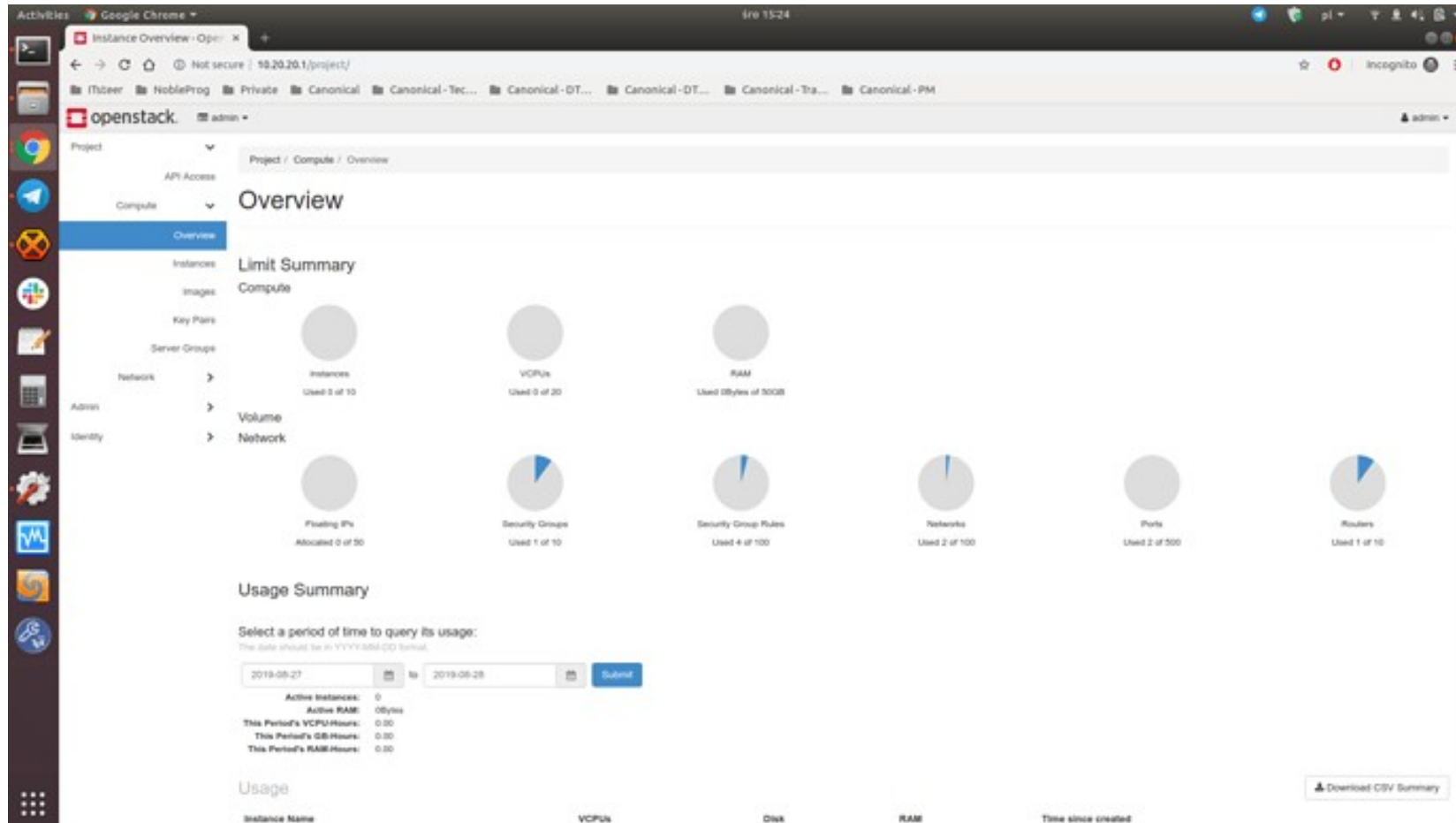


Keystone

Authentication and authorization of OpenStack services

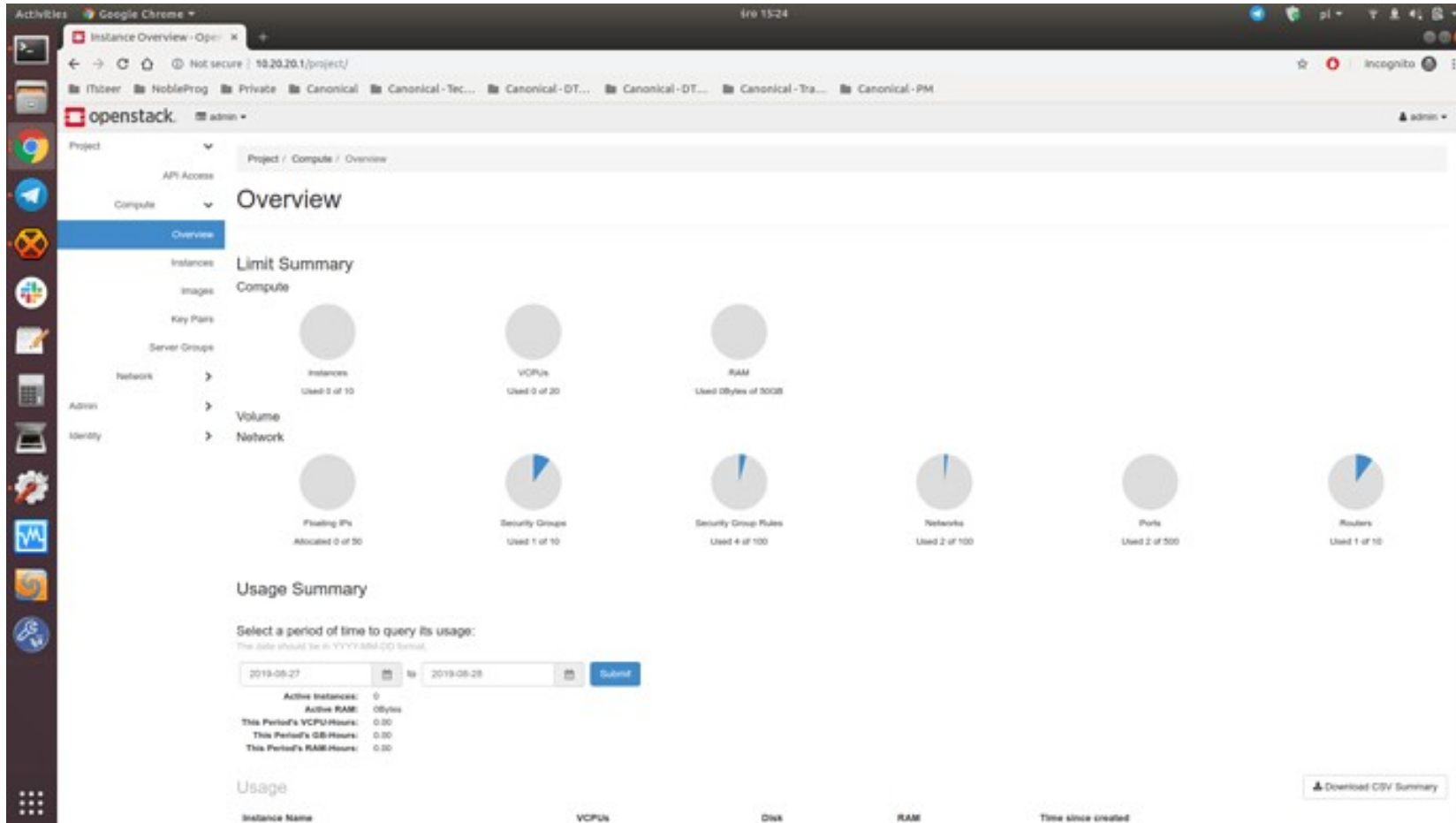


OpenStack Landing Page



Horizon

Dashboard to manage OpenStack services



Horizon

Dashboard to manage OpenStack services.

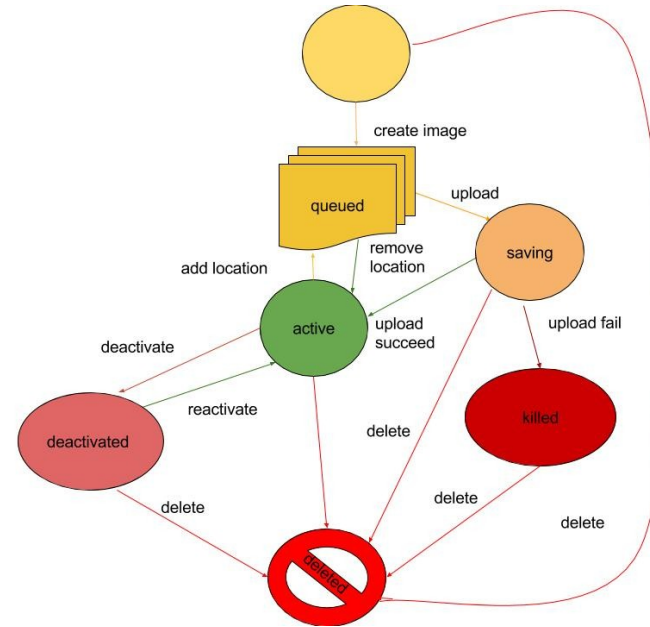
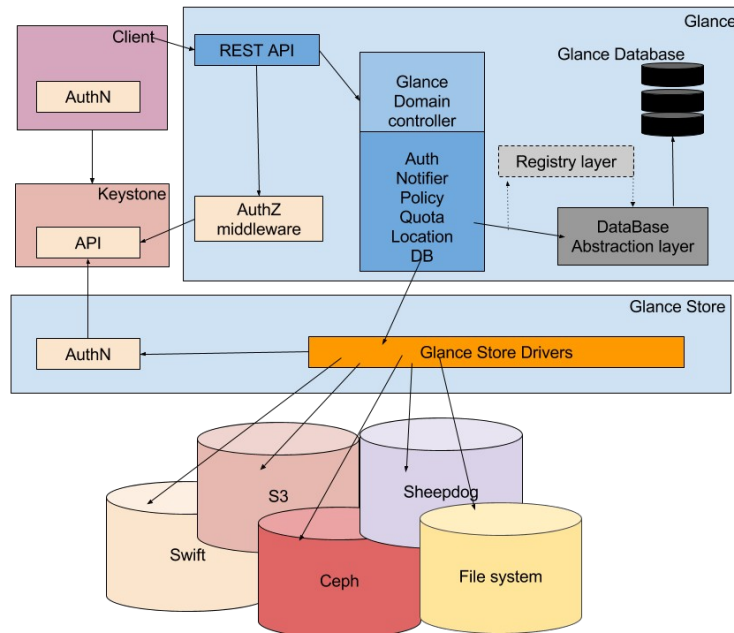
The screenshot shows the OpenStack Horizon dashboard interface. The top navigation bar is orange and contains the 'ubuntu' logo and the user 'default • admin'. The left sidebar is a vertical menu with several sections: 'Project', 'Compute', 'Network', 'Routers', 'Firewalls', 'Admin', 'Identity', and 'cloudbuilder.in'. Three arrows point to specific elements: 'Dashboard' points to the 'Project' menu item, 'Panel Group' points to the 'Network' menu item, and 'Panel' points to the 'Networks' panel in the main content area. The 'Networks' panel displays a table with two rows of network information.

<input type="checkbox"/>	Name	Subnets A
<input type="checkbox"/>	network2	subnet2 19
<input type="checkbox"/>	network1	subnet1 20

Displaying 2 items

Glance

Image service which provides discovering, registering, retrieving for disk and server images

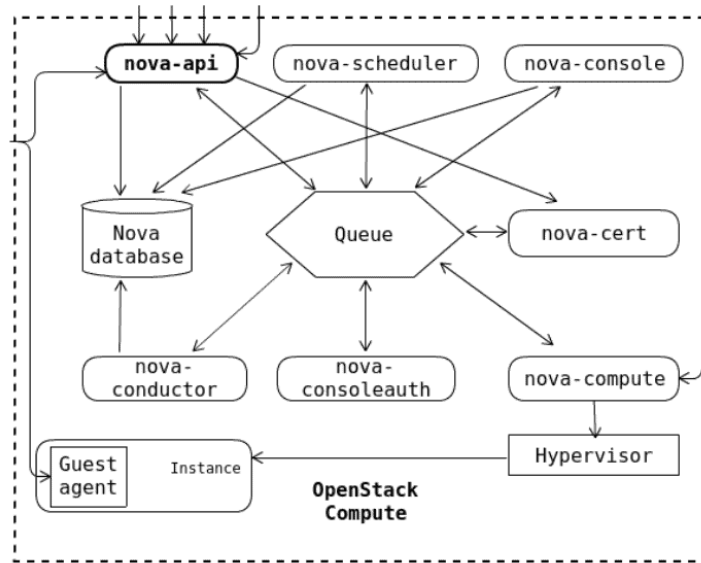


Glance Components :

1. **glance-api** - accepts API calls for image discovery, retrieval and storage.
2. **glance-registry** - stores, processes, and retrieves metadata information for images.
3. **database** - stores image metadata
4. **storage repository** - integrates with various outside OpenStack components such as regular file systems, Amazon S3 and HTTP for image storages.

Nova

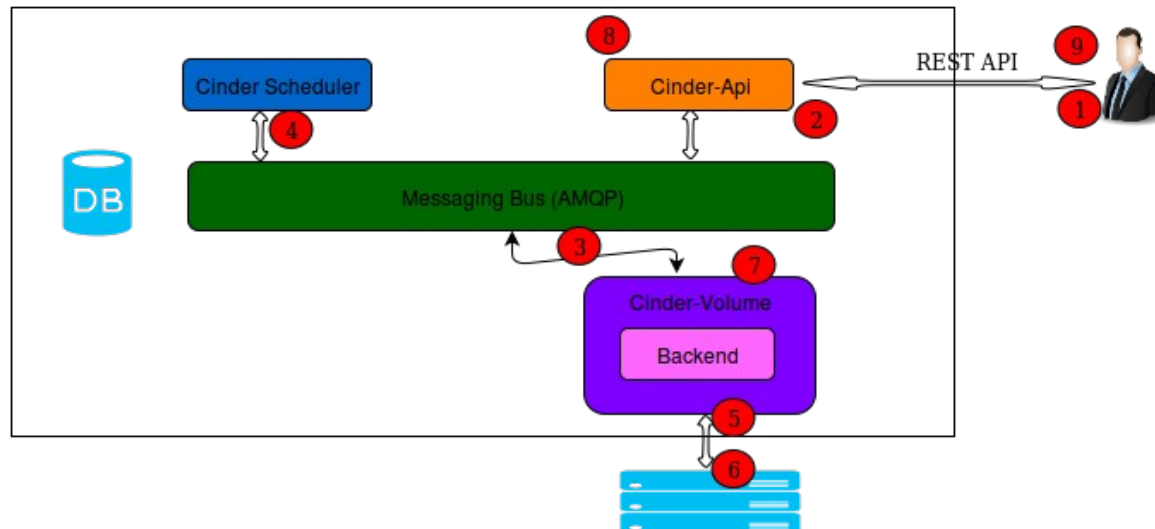
Manages and provisions virtual machines running on hypervisor nodes



1. End user submits the virtual machine creation request via the OpenStack dashboard
2. The **nova-api** calls will get sent to the **nova-api service**.
3. The **nova-api service** will pass the APIs for instance creation to the **nova-compute service**
(knows what API the underlying KVM hypervisor will support)
4. Pointing to the underlying KVM hypervisor, the nova-compute will trigger the **libvirt-api** for virtual machine creation.
5. KVM hypervisor processes the **libvirt-api** request and creates a new virtual machine.

Cinder

Block Storage Service



DB: sql database for data storage. Used by all components

Cinder API : component that receives http requests, converts commands and communicates with other components via the queue or http.

AMQP broker : usually RabbitMQ that sits between any two Cinder components and allows them to communicate in a loosely coupled fashion.

Scheduler : decides which host gets each volume.

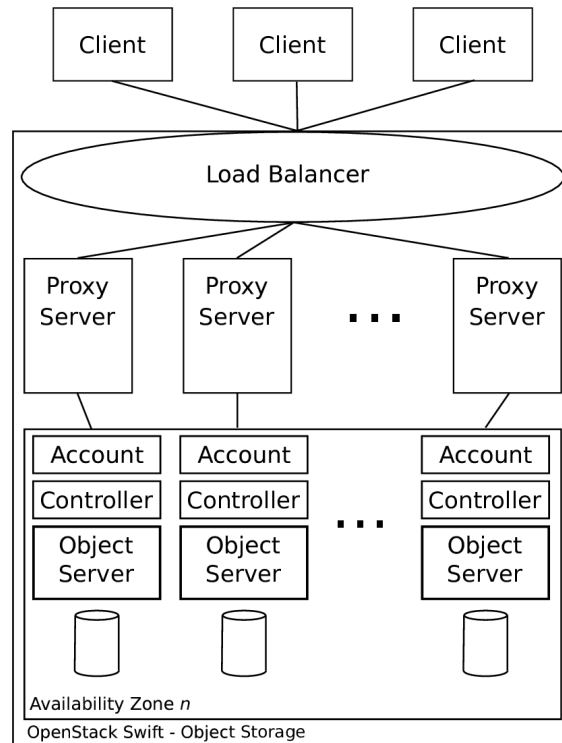
Volume : manages dynamically attachable block devices.

Backup : manages backups of block storage devices.

Auth Manager : component responsible for users/projects/and roles. Can backend to DB or LDAP. This is not a separate binary, but rather a python class that is used by most components in the system.

Swift

Object Storage Service



Account : Service provider creates your account and you own all resources in that account. The account defines a namespace for containers. A container might have the same name in two different accounts.

Container : Defines a namespace for objects. An object with the same name in two different containers represents two different objects.

Object : Stores data content, such as documents, images, and metadata

Neutron

Network Service

