



CHAPTER 2

Red Hat Enterprise Linux OpenStack Platform

Red Hat Enterprise Linux OpenStack Platform provides the foundation to build private or public Infrastructure-as-a-Service (IaaS) for cloud-enabled workloads. It allows organizations to leverage OpenStack, the largest and fastest growing open source cloud infrastructure project, while maintaining the security, stability, and enterprise readiness of a platform built on Red Hat Enterprise Linux.

Red Hat Enterprise Linux OpenStack Platform gives organizations a truly open framework for hosting cloud workloads, delivered by Red Hat subscription for maximum flexibility and cost effectiveness. In conjunction with other Red Hat technologies, Red Hat Enterprise Linux OpenStack Platform allows organizations to move from traditional workloads to cloud-enabled workloads on their own terms and time lines, as their applications require. Red Hat frees organizations from proprietary lock-in, and allows them to move to open technologies while maintaining their existing infrastructure investments.

Unlike other OpenStack distributions, Red Hat Enterprise Linux OpenStack Platform provides a certified ecosystem of hardware, software, and services, an enterprise life cycle that extends the community OpenStack release cycle, and award-winning Red Hat support on both the OpenStack modules and their underlying Linux dependencies. Red Hat delivers long-term commitment and value from a proven enterprise software partner so organizations can take advantage of the fast pace of OpenStack development without risking stability and supportability of their production environments.

Red Hat Enterprise Linux OpenStack (Havana) Services

Red Hat Enterprise Linux OpenStack Platform 4 is based on the upstream Havana OpenStack release. Red Hat Enterprise Linux OpenStack Platform 4 is the fourth release from Red Hat. The first release was based on the Essex OpenStack release. The second release was based on the Folsom OpenStack release. It was the first release to include extensible block and volume storage services. The third release was based on grizzly.

The following OpenStack components are discussed.

- [Identity Service \(Keystone\)](#), page 2-2
- [Image Service \(Glance\)](#), page 2-2
- [Compute Service \(Nova\)](#), page 2-2
- [Block Storage \(Cinder\)](#), page 2-3
- [Network Service \(Neutron\)](#), page 2-3
- [Dashboard \(Horizon\)](#), page 2-3
- [Telemetry Service](#), page 2-4
- [Heat \(Orchestration Service\)](#), page 2-4

Identity Service (Keystone)

This is a central authentication and authorization mechanism for all OpenStack users and services. It supports multiple forms of authentication including standard username and password credentials, token-based systems and AWS-style logins that use public/private key pairs. It can also integrate with existing directory services such as LDAP.

The Identity service catalog lists all of the services deployed in an OpenStack cloud and manages authentication for them through endpoints. An endpoint is a network address where a service listens for requests. The Identity service provides each OpenStack service – such as Image, Compute, or Block Storage—with one or more endpoints.

The Identity service uses tenants to group or isolate resources. By default users in one tenant can't access resources in another even if they reside within the same OpenStack cloud deployment or physical host. The Identity service issues tokens to authenticated users. The endpoints validate the token before allowing user access. User accounts are associated with roles that define their access credentials. Multiple users can share the same role within a tenant.

The Identity Service is comprised of the keystone service, which responds to service requests, places messages in queue, grants access tokens, and updates the state database.

Image Service (Glance)

This service discovers, registers, and delivers virtual machine images. They can be copied via snapshot and immediately stored as the basis for new instance deployments. Stored images allow OpenStack users and administrators to provision multiple servers quickly and consistently. The Image Service API provides a standard RESTful interface for querying information about the images.

By default the Image Service stores images in the `/var/lib/glance/images` directory of the local server's file system where Glance is installed. The Glance API can also be configured to cache images to reduce image staging time. The Image Service supports multiple back end storage technologies including Swift (the OpenStack Object Storage service), Amazon S3, and Red Hat Storage Server.

The Image service is composed of the `openstack-glance-api` that delivers image information from the registry service, and the `openstack-glance-registry`, which manages the metadata, associated with each image.

Compute Service (Nova)

OpenStack Compute provisions and manages large networks of virtual machines. It is the backbone of OpenStack's IaaS functionality. OpenStack Compute scales horizontally on standard hardware enabling the favorable economics of cloud computing. Users and administrators interact with the compute fabric via a web interface and command line tools.

Key features of OpenStack Compute include:

- Distributed and asynchronous architecture, allowing scale out fault tolerance for virtual machine instance management
- Management of commodity virtual server resources, where predefined virtual hardware profiles for guests can be assigned to new instances at launch
- Tenants to separate and control access to compute resources
- VNC access to instances via web browsers

OpenStack Compute is composed of many services that work together to provide the full functionality. The `openstack-nova-cert` and `openstack-nova-consoleauth` services handle authorization. The `openstack-nova-api` responds to service requests and the `openstack-nova-scheduler` dispatches the requests to the message queue. The `openstack-nova-conductor` service updates the state database, which limits direct access to the state database by compute nodes for increased security. The `openstack-nova-compute` service creates and terminates virtual machine instances on the compute nodes. Finally, `openstack-nova-novncproxy` provides a VNC proxy for console access to virtual machines via a standard web browser.

Block Storage (Cinder)

While the OpenStack Compute service provisions ephemeral storage for deployed instances based on their hardware profiles, the OpenStack Block Storage service provides compute instances with persistent block storage. Block storage is appropriate for performance sensitive scenarios such as databases or frequently accessed file systems. Persistent block storage can survive instance termination. It can also be moved between instances like any external storage device. This service can be backed by a variety of enterprise storage platforms or simple NFS servers. This service's features include:

- Persistent block storage devices for compute instances
- Self-service user creation, attachment, and deletion
- A unified interface for numerous storage platforms
- Volume snapshots

The Block Storage service is comprised of `openstack-cinder-api`, which responds to service requests, and `openstack-cinder-scheduler` which assigns tasks to the queue. The `openstack-cinder-volume` service interacts with various storage providers to allocate block storage for virtual machines. By default the Block Storage server shares local storage via the ISCSI `tgt` daemon.

Network Service (Neutron)

OpenStack Networking is a scalable API-driven service for managing networks and IP addresses. OpenStack Networking gives users self-service control over their network configurations. Users can define, separate, and join networks on demand. This allows for flexible network models that can be adapted to fit the requirements of different applications.

OpenStack Networking has a pluggable architecture that supports numerous physical networking technologies as well as native Linux networking mechanisms including `openvswitch` and `linuxbridge`. It also supports Cisco Nexus plug-in.

Dashboard (Horizon)

The OpenStack Dashboard is an extensible web-based application that allows cloud administrators and users to control and provision compute, storage, and networking resources. Administrators can use the Dashboard to view the state of the cloud, create users, assign them to tenants, and set resource limits. The OpenStack Dashboard runs as an Apache HTTP server via the `httpd` service.

**Note**

Both the Dashboard and command line tools can be used to manage an OpenStack environment. This document focuses on the command line tools because they offer more granular control and insight into OpenStack's functionality.

Telemetry Service

The Telemetry service provides user-level usage data for OpenStack-based clouds, which can be used for customer billing, system monitoring, or alerts. Data can be collected by notifications sent by existing OpenStack components (for example, usage events emitted from Compute) or by polling the infrastructure (for example, libvirt). Refer to [Table 2-1](#).

Telemetry includes a storage daemon that communicates with authenticated agents through a trusted messaging system, to collect and aggregate data. Additionally, the service uses a plugin system, which makes it easy to add new monitors.

Table 2-1 *Telemetry Service Components*

Component	Description
ceilometer-agent-compute	An agent that runs on each Compute node to poll for resource utilization statistics.
ceilometer-agent-central	An agent that runs on a central management server to poll for utilization statistics about resources not tied to instances or Compute nodes.
ceilometer-collector	An agent that runs on one or more central management servers to monitor the message queues. Notification messages are processed and turned into Telemetry messages, and sent back out on to the message bus using the appropriate topic. Telemetry messages are written to the data store without modification.
MongoDB database	For collected usage sample data.
API Server	Runs on one or more central management servers to provide access to the data store's data. Only the Collector and the API server have access to the data store.

Heat (Orchestration Service)

The Orchestration service provides a template-based orchestration engine for the OpenStack cloud, which can be used to create and manage cloud infrastructure resources such as storage, networking, instances, and applications as a repeatable running environment.

Templates are used to create stacks, which are collections of resources (for example instances, floating IPs, volumes, security groups, or users). The service offers access to all OpenStack core services using a single modular template, with additional orchestration capabilities such as auto-scaling and basic high availability.

Features include:

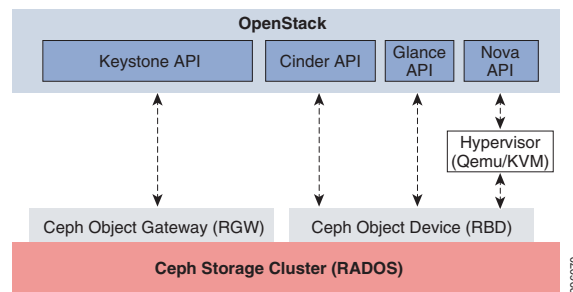
- A single template provides access to all underlying service APIs.
- Templates are modular (resource oriented).

- Templates can be recursively defined, and therefore reusable (nested stacks). This means that the cloud infrastructure can be defined and reused in a modular way.
- Resource implementation is pluggable, which allows for custom resources.
- Autoscaling functionality (automatically adding or removing resources depending upon usage).
- Basic high availability functionality.

Ceph Storage for OpenStack

While Ceph is not part of the default Red Hat Enterprise Linux OpenStack Platform distribution, Ceph has been certified as part of the solution. Ceph is a massively scalable, open source, software defined storage system (Figure 2-1). It offers an object store and a network block device, unified for the cloud. The platform is capable of auto-scaling to the exabyte level and beyond, it runs on commodity hardware, is self-healing and self-managing, and has no single point of failure. Ceph is in the Linux kernel, and has been integrated with OpenStack since the Folsom release. Ceph is ideal for creating flexible, easy to operate object and block cloud storage.

Figure 2-1 Ceph Architecture Overview



Unlike every other storage solution for OpenStack, Ceph uniquely combines object and block into one complete storage powerhouse for all your OpenStack needs. Ceph is a total replacement for Swift with distinctive features such as intelligent nodes and a revolutionary deterministic placement algorithm, along with a fully integrated network block device for Cinder. Ceph's fully distributed storage cluster and block device decouple compute from storage in OpenStack, allowing mobility of virtual machines across your entire cluster. Ceph block device also provides copy on write cloning that enables you to quickly create a thousand VMs from a single master image, requiring only enough space to store their subsequent changes.

Red Hat Enterprise Linux

Red Hat Enterprise Linux 6, the release of Red Hat trusted data center platform, delivers advances in application performance, scalability, and security. With Red Hat Enterprise Linux 6, physical, virtual, and cloud computing resources can be deployed within the data center.

Supporting Technologies

This section describes the supporting technologies used to develop this reference architecture beyond the OpenStack services and core operating system. Supporting technologies include:

- **MySQL**—A state database resides at the heart of an OpenStack deployment. This SQL database stores most of the build-time and run-time state information for the cloud infrastructure including available instance types, networks, and the state of running instances in the compute fabric. Although OpenStack theoretically supports any SQL-Alchemy compliant database, Red Hat Enterprise Linux OpenStack Platform 4 uses MySQL, a widely used open source database packaged with Red Hat Enterprise Linux 6.
- **Qpid** —OpenStack services use enterprise messaging to communicate tasks and state changes between clients, service endpoints, service schedulers, and instances. Red Hat Enterprise Linux OpenStack Platform 4 uses Qpid for open source enterprise messaging. Qpid is an Advanced Message Queuing Protocol (AMQP) compliant, cross-platform enterprise messaging system developed for low latency based on an open standard for enterprise messaging. Qpid is released under the Apache open source license.
- **KVM**—Kernel-based Virtual Machine (KVM) is a full virtualization solution for Linux on x86 and x86_64 hardware containing virtualization extensions for both Intel and AMD processors. It consists of a loadable kernel module that provides the core virtualization infrastructure. Red Hat Enterprise Linux OpenStack Platform Compute uses KVM as its underlying hypervisor to launch and control virtual machine instances.