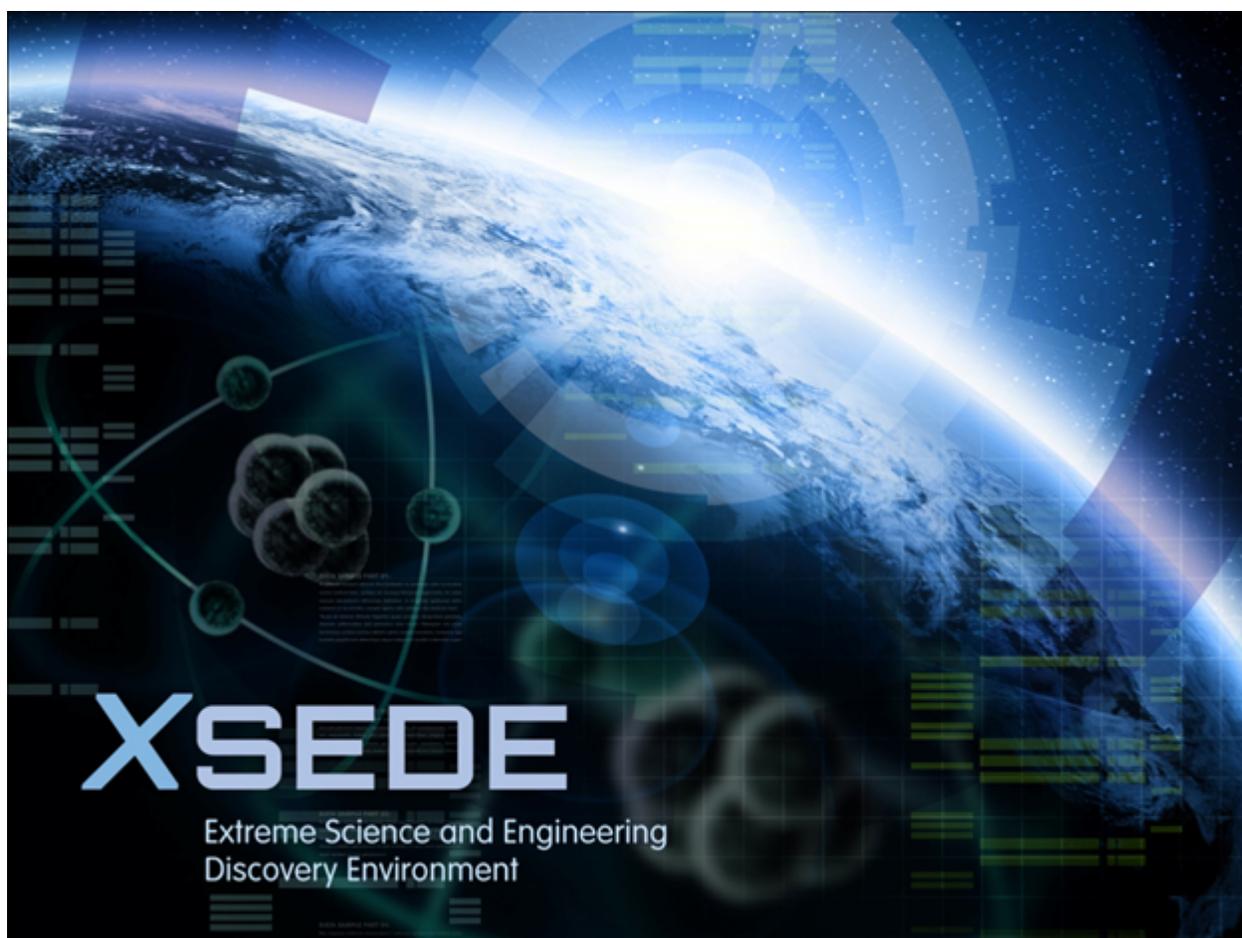


# OpenStack Evaluation Report

28 August 2015

Version 1.3



## Table of Contents

[Document History](#)

[Document Scope](#)

[Executive Summary](#)

[Introduction](#)

[Prerequisites](#)

[Evaluation Details](#)

[Test run 1](#)

[Test run 2](#)

[Test run 3](#)

[Test run 4](#)

[Results](#)

[Installation Information](#)

[Usage Information](#)

[Evaluation Result](#)

[Appendix 1: Suggestions for Evaluation Procedure Revisions](#)

## A. Document History

Document History				
Entire Document	1.0	04/23/2015	Created Document	Jason Charcalla
Entire Document	1.1	05/07/2015	Major edits	Jason Charcalla
Entire Document	1.2	07/30/2015	Major edits	Peter Enstrom, Jason Charcalla
Entire Document	1.3	08/28/2015	Final Draft	Peter Enstrom, Jason Charcalla, Robert Budden, Derek Simmel

## **B. Document Scope**

This document is the evaluation report for the functional evaluation of OpenStack. The results of the evaluation and the collected information here are intended for the benefit of the XSEDE project and the general eScience community.

## C. Executive Summary

An evaluation of the OpenStack Icehouse and Juno releases was performed by the Technology Investigation Service (TIS) evaluation team led by Robert Budden. The other members of the evaluation team were Jason Charcalla, and Derek Simmel. The team evaluated multiple versions of OpenStack, a public and private cloud software stack that allows users to provision virtual and physical resources. The team has concluded that OpenStack provides technologies that will be essential to XSEDE SP cloud operations in the near future. XSEDE SP operations staff should become familiar with OpenStack as it is rapidly emerging as the de facto standard for virtualized infrastructure deployment.

OpenStack is a complex suite that provides an abstraction layer for all data center infrastructure and services. As such, OpenStack deployment, monitoring, maintenance and operation requires a team of staff with knowledge, roles and responsibilities that cover the equivalent space as for conventional machine room infrastructure and service deployments, including systems management and administration, networking, security, user interfaces and user support.

We recommend adoption of OpenStack for larger-scale implementations with enough staff and resources dedicated to it for long term operation. Projects interested in the services and capabilities provided by OpenStack are well-advised to identify and work with an existing production-quality operator of OpenStack infrastructure rather than to “roll their own.”

Once familiarized with the various components administrators should find that OpenStack provides a reliable platform for infrastructure service virtualization that is maturing at a growing rate.

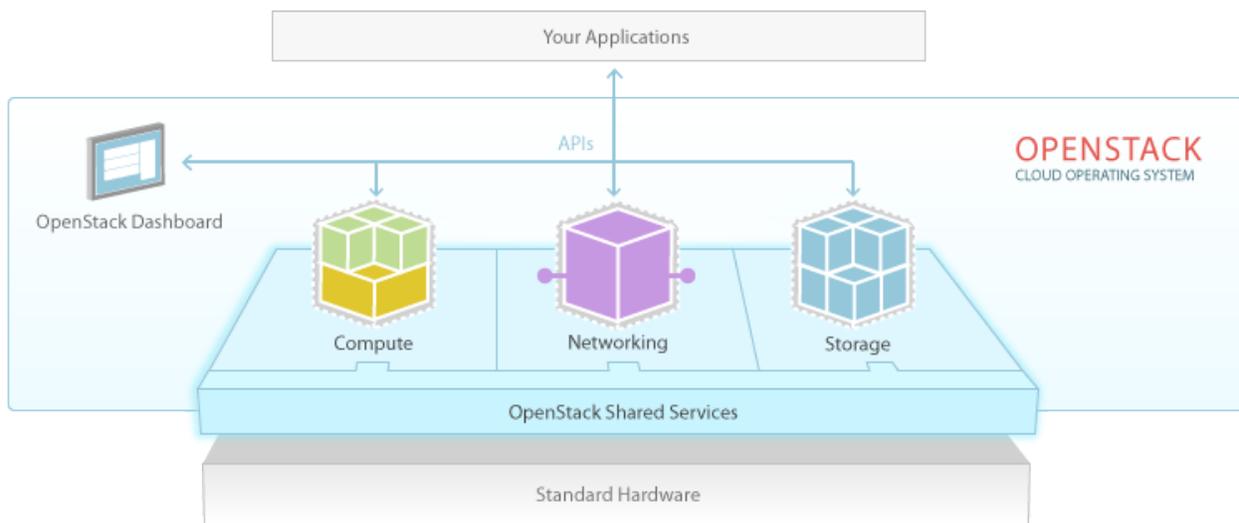
Team recommends software: YES

## D. Introduction

OpenStack is a large, complex collection of open source projects that combined provide end users the ability to provision virtual (and physical) machines, networks, storage and other resources. OpenStack is an abstraction layer which provides the tools necessary to implement, scale, and support large pools of computing, storage and network resources..

OpenStack enables organizations to provide end users and customers with a robust selection of PAAS (platform as a service) and SAAS (software as a service) offerings out of the box. OpenStack currently supports the provisioning of “bare metal” servers, Linux (LXC) and Docker containers, as well as typical KVM virtual machine support. Additionally it provides the ability to provision networks, storage resources, and database services. The upcoming OpenStack project ‘Sahara’ introduces big data processing capabilities with Hadoop.

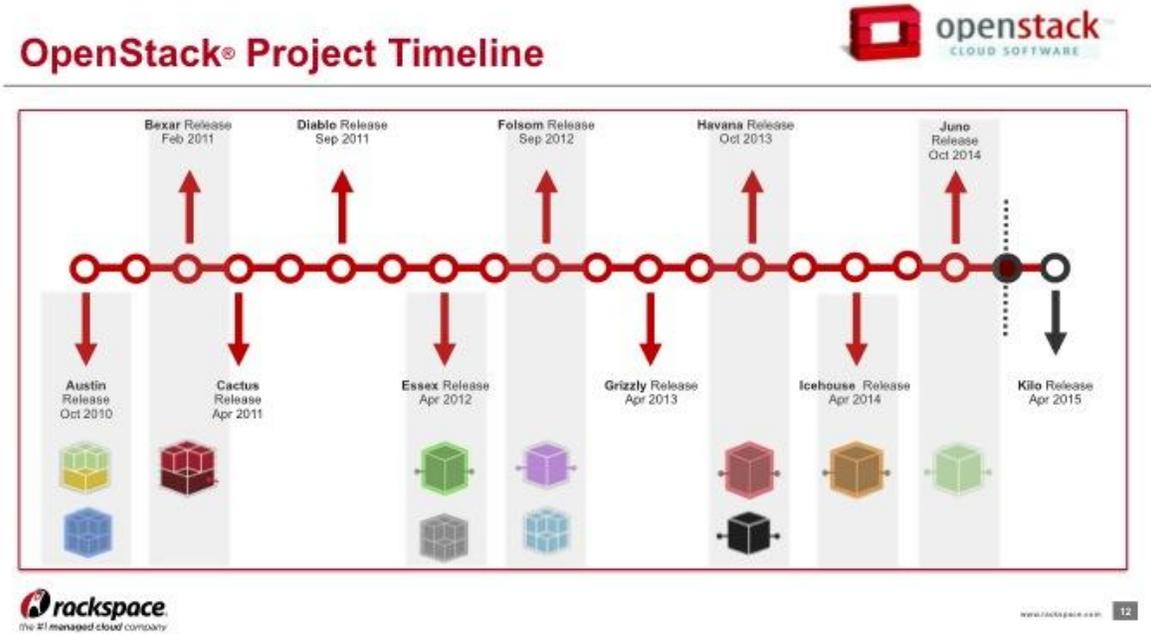
Essential components of OpenStack include *Keystone* (Account management), *Nova* (Compute resources and scheduling), *Neutron* (Network provisioning), *Swift* (object storage), *Glance* (Image catalogue), and *Cinder* (block storage). Most of these projects rely heavily on various common open source technologies such as databases, LVM, DHCP, message queuing (RabbitMQ, Qpid, ZeroMQ), Open vSwitch (OvS), OpenFlow, and others. These combined technologies allow it to virtualize an entire machine room, resulting in an infrastructure project that is very complicated and hard to master without expert support. The following illustration from the [www.OpenStack.org](http://www.OpenStack.org) website<sup>1</sup> provides a high-level view of the OpenStack ecosystem:



OpenStack has in recent years rapidly gained a major share of the public and private cloud infrastructure providers around the globe. Turnout for the bi-annual OpenStack Summit conference has grown from 75 in 2010 to over 6000 in May 2015. The annual OpenStack Summit and OpenStack Developer conferences coincide with the 6 month OpenStack release cycle, which

<sup>1</sup> <http://www.openstack.org/themes/openstack/images/openstack-software-diagram.png>

combined with outstanding community support has further fueled the project’s explosive growth.



As more computational resources continue to move from bare metal to IAAS (Infrastructure as a service) cloud implementations, OpenStack may become an essential piece in XSEDE’s SP infrastructure. The OpenStack-based NSF *Chameleon* and *CloudLab* projects and forthcoming PSC *Bridges* and TACC/IU *JetStream* systems highlight XSEDE’s growing interest in OpenStack as an essential infrastructure technology. This TIS evaluation provides valuable advance experimentation and documentation for these projects to build upon.

OpenStack can be installed on various compute and network hardware platforms as well as various flavors of Linux. Several major operating system vendors have begun to package their own distribution of OpenStack. In this evaluation, we chose to implement on CentOS using RedHat’s RDO installation scripts. RDO is not an acronym; it is the name of a project started by RedHat to build a distribution of OpenStack. The RDO product relies heavily on the Puppet configuration management tool to ensure the consistency of service and node configurations. The team investigated the RDO installation process, administrator features such as user management, and end user features including CLI and web-based provisioning of virtual resources.

### Prerequisites

Prerequisites for a successful OpenStack deployment include multiple dedicated servers, with storage and appropriate networking, including availability of publicly accessible IP addresses for virtual systems that are meant to be directly accessible by remote users. The base operating system for OpenStack infrastructure and cloud nodes is typically a well-supported Linux distribution. For the purposes of this evaluation, we have employed CentOS 6.6 and CentOS 7.1. As we have chosen to implement OpenStack using the Redhat RDO distribution, access to the Redhat RDO repositories

(<https://www.rdoproject.org>) is also necessary.

Based on our experimentation using *Icehouse*, *Juno*, and *Kilo* releases of OpenStack, a complete and self-contained RedHat RDO installation of OpenStack may be deployed on a single system with the following minimum host system capacities:

- 8GB RAM
- one or more multi-core x86\_64 Intel Xeon or equivalent processors
- 100GB local disk (CentOS 7 + OpenStack installation occupies ~6.2GB)
- one or more network interfaces with at least 1Gb/s throughput

In addition, it is helpful to have a private administrative console interface available via IPMI or serial console to the host systems on which the OpenStack installation is operating. It is essential that these administrative interfaces are strongly secured against public access; ideally these interfaces should incorporate multi-factor authentication methods.

In a multi node configuration, special attention must be paid network topology. Although it is technically possible to deploy OpenStack with a single network interface per node, the team recommends a minimum of two. This method allows you to isolate OpenStack management traffic from the (user) tenant *Neutron* traffic. It should also be noted that OpenStack provides its own DHCP services; therefore, unless you have an isolated subnet, coordination with your site's network administration team will be required.

For the *Juno* and *Kilo* releases of OpenStack, the recommended base operating system for the RDO installation host is RedHat Enterprise Linux 7 or its equivalent. CentOS 6 can be used for previous versions. We employed a minimal installation of CentOS 7 for this purpose in this evaluation.

## E. Evaluation Details

The requirements defined for this evaluation were:

- The software will notify admins of service failures.
- The software will allow users to deploy VMs quickly within an environment that will allow use of custom network settings, license keys, and user definable compute resources.
- The software will provide user error messages that are appropriate, useful and non-cryptic.
- The software will allow per-project networks to be created dynamically by a user
- The software will provide multiple ways for a user to access their VMs (direct SSH to public IP, dashboard console)
- The dashboard interface must be easy to learn.
- The dashboard interface must allow users to upload and download VM images

Due to OpenStack's size and complexity, the test plan was designed to touch upon a limited but sufficient collection of components. This includes the installation process via the RDO deployment scripts, modifying an existing OpenStack cluster, and simple and complex virtual machine provisioning that emulates the user experience. Administrative features such as user management and troubleshooting logs were also included.

OpenStack test deployments were configured at both PSC and NICS.

Pittsburgh Supercomputing Center (PSC):

Several different systems were employed to prototype and learn about RDO-based OpenStack deployment. A final test setup was deployed using three servers, named *riverrun.psc.edu*, *winterfell.psc.edu* and *thunder.psc.edu*. The *riverrun* and *winterfell* nodes have Supermicro X7DBN motherboards with 8-core Intel Xeon E5345 processors running at 2.33GHz and 8GB RAM installed. The *thunder* node is an older nVidia-based system with a dual-core AMD Opteron 165 processor running at 1.8GHz and 4GB RAM installed. All three nodes have pairs of 250GB or 512GB local SATA hard drives. Complete and self-contained RDO installations of OpenStack were made on *riverrun* and *winterfell*, and *thunder* was added as an additional compute node. Two Intel e1000 1Gbps network interfaces on each node were connected via CAT5 copper to a local Cisco switch in a conventional (i.e., non-trunk) mode.

National Institute for Computational Sciences (NICS):

The NICS OpenStack deployment consist of 6 Cray (Appro) blades with dedicated network switching hardware. Each blade has a single Intel Xeon E5-2670 processor running at 2.6GHz with 64GB RAM and 1Tb of local SATA HDD storage. A dedicated 1Gbps Ethernet switch is utilized for all management traffic while *Neutron* networking operates on a separate 10GigE interface. *Neutron* was configured in VLAN mode with multiple VLANs trunked to each corresponding 10GigE port allowing tenant networks to map to a VLAN for ease of administration and troubleshooting. All 6 nodes sit behind a single login server. This

allows for added security in that all OpenStack API endpoints and virtual machine floating IP's reside in a non-routable subnet only accessible from the designated login machine. The login server requires standard NICS RSA One-Time Password (OTP) authentication. Additionally this allows for more floating IP's to be available.

The *Icehouse* version of OpenStack was chosen for this instance as it was the current version at the time. The cluster is configured with one cloud controller node (*Keystone, Cinder, Swift, Nova, and Horizon*), one network node (*Neutron*), and 4 cloud compute nodes (*Nova*).

### **Test run 1 - Cloud Installation:**

In this first test OpenStack was installed on machines at both PSC and NICS. At PSC, *Juno* was installed on the machines *riverrun.psc.edu* and *winterfell.psc.edu*. At NICS, *Icehouse* was installed on the *Thunderhead* system which is comprised of 6 Cray (Appro) blades and can be accessed from its designated login node.

- Install OpenStack via the RDO install tool, as described at <https://www.rdoproject.org/Quickstart>
- Test that the install was a success with various web-based and CLI commands.
- Add an additional compute node to the cluster using the RDO packstack utility.

### **Test run 2 - Simple node provisioning:**

The simple node provisioning tests evaluated the ability for users to provision virtual machines from both the CLI and *Horizon* dashboard web interface. These tests were designed to exercise provisioning of multiple VM's simultaneously. They also incorporated situations designed to induce errors to ensure they were reported in a user-friendly way.

- Successfully log into the *Horizon* dashboard or source the provided CLI *keystonerc\_admin* file generated by RDO. The *keystonerc\_admin* file contains login credentials and the authentication URL of the keystone service.
- Provision a single virtual machine instance with the appropriate SSH key, security groups, and network configuration.
- Log into the new instance via SSH.
- Provision a new instance with additional disk, memory, and CPU resources (flavors).
- Provision five virtual machines simultaneously.
- Provision ten virtual machines simultaneously.
- Attempt to provision more machines than your quota will allow and record the error.

### **Test run 3 - Complex node provisioning:**

The complex node provisioning portion of the tests evaluated the ability to attach persistent block storage devices to virtual machines via the *Cinder* APIs. The team also tested advanced networking functionalities by assigning multiple virtual networks to running virtual machine instances.

- Create and attach a *Cinder* block storage volume to a virtual machine instance.
- Verify block storage is writable and persistent.
- Create and attach multiple networks to a virtual machine instance.
- Perform basic network testing with *ping* and *iperf3*.

### Test run 4 - Cloud management:

The team tested administration functionality of OpenStack. These features would be used by user support staff in a full deployment and not just the OpenStack system administrators. Because of this, error reporting for basic features (e.g., duplicate user and failed authentication attempts) were incorporated into the evaluation. User management functionality including adding new users and projects (“tenants”), as well as the ability to upload custom images and to configure tenant networks were also tested.

- Add users and tenants via the CLI tools and verify they can log into *Horizon*.
- Add a tenant Network and router.
- Add a public cloud image that is accessible by all users.
- Verify error messages were logged for troubleshooting

### Test Run Results

PSC:

The RDO-based installation method provides a quick way to get a working installation with the primary components of OpenStack deployed. RDO handles the gory details and manual setup of passwords, authentication keys, databases, and numerous configuration files. At the start of this investigation, a *Havana* setup was manually set up at PSC without the aid of RDO, which was immensely more complicated and time consuming than the automated RDO installation.

As of the *Juno* release, the configuration of networking interfaces, virtual routers or switches, and pool of addresses for VM instances requires modification of network interface definitions prior to RDO installation of OpenStack, which is not initially obvious. Access to running instances of VMs from external systems also requires modifications that are not well-documented.

As of the *Juno* and *Kilo* releases, there are still some quirks with the initial networking setup, specifically with the host portion of the advanced networking service, *Neutron*. RDO attempts to edit the files in */etc/sysconfig/network-scripts/ifcfg-XXX* to migrate RedHat/CentOS 6-style eth# interfaces to an OvS bridge. Most of the configuration is done properly but hand checking is required BEFORE rebooting. Namely, the use of *systemd* conventions default on the RedHat/CentOS 7 host OS may need to be reverted (i.e. IPADDR0, GATEWAY0, etc.). The */etc/sysconfig/network-scripts/ifcfg-br-ex* network interface configuration file needs to be manually corrected after RDO runs to define a working configuration as an OVSBridge interface to the external network. It is advisable to have a secondary private network interface running or IPMI in case of configuration errors, so that the host remains accessible via a secured IPMI or other secondary console despite

network configuration problems on the primary external network interface.

Additionally, the internal *Neutron* configuration is left to the administrator. Scripts were developed and are available on the XSEDE wiki to aid/automate the initial setup of external networks, project/tenant virtual routers, and private virtual subnets. These pieces are required to successfully deploy virtual machines under OpenStack. The script and sample configuration file can be found at: <https://www.xsede.org/web/staff/staff-wiki/-/wiki/Main/OpenStack+Cloud+Evaluation#section-OpenStack+Cloud+Evaluation-KiloNeutronConfigurationSimplified>

One key issue with RDO and *Neutron* is that the default DNSMasq setup is not configured properly; this breaks SNAT connectivity from spawned virtual machines. It is necessary to edit `/etc/neutron/dhcp_agent.ini` and update the `dnsmasq_dns_servers` and `dhcp_domain` lines accordingly. These values are required for correct virtual router operation.

When deploying a multi-node OpenStack setup using RDO and using *Neutron* for network management, it is important NOT to use the `--all-in-one` option in the packstack answers file. This configuration option affects the initial OVS bridge setup and may introduce problems during the “add a compute node” phase.

NICS:

NICS utilized the RDO installation method as well. While extremely useful in ensuring that no pieces of OpenStack are overlooked during the process, several attempts were required to produce a working RDO answer file. The team encountered multiple instances where errors were not correctly reported or had little documentation. The majority of these issues gravitated around the networking components. A thorough understanding of networking and a proper implementation plan are essential to producing a working RDO deployment of OpenStack.

OpenStack is still an active and rapidly developed software which is maturing quickly. At the time of this writing the versions our team tested have been superseded by newer releases. The current release, *Kilo*, was made available April 30, 2015, with the follow on release, *Liberty*, scheduled for October 2015. Because of the 6-month release cycle, the documentation and install procedures are not as polished and current as desired. Initial planning of a cloud infrastructure based on OpenStack is problematic, due to the complexity of OpenStack, its large number of configuration options and numerous component services. Additionally, vague error messages add to the difficulty level and increase the slope of the OpenStack learning curve. The team agrees that these issues will probably be worked out as the project matures and greater experience is gained with the OpenStack ecosystem. It should be noted that the rapidly growing OpenStack community is working hard to reduce the complexity of the installation with the development of multiple open source and commercial products for deployment.

Once all the pieces are in place and functioning correctly, OpenStack functions as a robust cloud software stack providing users with multiple methods for provisioning and maintaining virtual machines and other resources such as storage and custom networking setups.



## F. Installation Information

As a collection of projects and commonly used open source software, OpenStack has multiple installation methods. The most time-consuming (albeit instructive) of these options is to manually install all of the various components. Because of its overall complexity, the team chose to use RDO, a RedHat-sponsored OpenStack installer tool. RDO greatly simplifies the process by allowing the administrator to select all desired components in a single answer file. This answer file is then used to deploy all the required and selected components on a single or multi node OpenStack cluster. A high level overview of the RDO installation procedure is summarized as follows:

1. Determine what hardware will become the controller, networking, and compute nodes. Cable the management and neutron network connections accordingly.
2. Install RedHat or CentOS (version 7 recommended) on all nodes and ensure installed software and services are up to date via *yum update*.
3. Configure network and document interfaces planned for use in OpenStack.
4. On a dedicated *controller* management node, install the RDO package repository and *packstack* package:
  - a. ``yum install -y https://rdoproject.org/repos/rdo-release.rpm``
  - b. ``yum install -y OpenStack-packstack``
5. For a multi-node installation, generate and modify the default *packstack* answers file to your environment:
  - a. ``packstack --gen-answer-file=packstack-answers-file.txt``
6. Run the *packstack* command with the generated answer file:
  - a. ``packstack --answer-file=packstack-answers-file.txt``

Upon running *packstack*, it will request the root password for each node and enable passwordless SSH with them using *publickey* authentication. Puppet manifest files generated by *packstack* ensure that all of the OpenStack components are installed and configured consistently.

## G. Usage Information

Administrators and users have multiple ways to interact with OpenStack. Two of the most common methods are the *Horizon* web dashboard and the OpenStack CLI tools. Restful API endpoints are also available for advanced users to write their own custom user interfaces. *Horizon* dashboard access is well documented and will not be covered in this section.

CLI access requires the installation the OpenStack client tools and creation of a *keystone\_rc* file, which is sourced in the user's shell to set up the environment and provide login credentials to the client tools. Each OpenStack project has its own CLI tools which are installed via the python *pip* package management system. For example, the *Nova* client may be installed via ``pip install python-novaclient``. The naming convention for the client tools available via *pip* is "*python-<OpenStack-Project-Name>client*".

## H. Evaluation Result

We attribute some degree of the difficulties encountered in getting the OpenStack installations operating for test purposes to our own inexperience with the product. The RedHat RDO distribution of OpenStack does much to enable a rapid initial deployment of OpenStack, but does not relieve us of the responsibility to master the essential configuration knowledge and skills required to operate an OpenStack ecosystem. It remains essential for operators to fully familiarize themselves with the components and methods of installation detailed in the OpenStack documentation matching the release being used (e.g., the OpenStack *Juno* documentation for RedHat and related Linux distributions available at <http://docs.OpenStack.org/juno/install-guide/install/yum/content/>).

The OpenStack learning curve is fairly steep, and the rapid pace of OpenStack development, with major releases every 6 months, requires OpenStack operators to continuously dedicate time and resources to staying abreast of OpenStack developments. It is therefore prudent to consider OpenStack as a software-based operations ecosystem that requires a sufficiently large team of operators to design, deploy, operate, monitor, support and maintain, much as one might do for a conventional HPC systems machine room; the team should include members with knowledge, skills and assigned responsibility for design, implementation, operation and maintenance in areas including:

- infrastructure acquisition and scaling for the OpenStack software itself
- OpenStack software administration and maintenance
- networking, network virtualization techniques, and software defined networking (SDN)
- risk assessment and policy definition
- security instrumentation, monitoring, and operations
- virtual machines (KVM), (Docker) containers and applications, and “bare metal” system building
- library stewardship for VMs, applications, and system images
- user experience design and user interface design

We recommend adoption of OpenStack for larger-scale implementations with enough staff and resources dedicated to it for long term operation. Projects interested in the services and capabilities provided by OpenStack are well-advised to identify and work with an existing production-quality operator of OpenStack infrastructure rather than to “roll their own.”

## I. Appendix 1: Suggestions for Evaluation Procedure Revisions

1. Given the scope and complexity of infrastructure management ecosystems such as OpenStack, progress may be facilitated by having the entire evaluation team work through and document implementation stages and tests together in a series of closed workshop sessions conducted at a single site, with site subject matter experts available for consultation as needed (e.g., site network administration staff).