RED HAT®
ENTERPRISE LINUX®
OPENSTACK® PLATFORM

redhat.

WHITEPAPER

INSTALLING RED HAT ENTERPRISE LINUX OPENSTACK PLATFORM

Taste of Red Hat Training course excerpt

Taste of Red Hat Training course excerpts, labs, videos, and webinars give you a sample of the expert level of educational content you'd find in a full Red Hat course. This excerpt is from our Red Hat OpenStack® Administration (CL210) course.

Red Hat OpenStack Administration teaches system administrators how to implement a cloud computing environment using Red Hat Enterprise Linux® OpenStack Platform, including installation, configuration, and maintenance. Through hands-on labs, students will explore fault-tolerant and redundant configurations of Red Hat Enterprise Linux OpenStack Platform, and will also look at the future plans of the OpenStack development community. This course can help you prepare for the Red Hat Certified System Administrator in Red Hat OpenStack exam (EX210).

In this course excerpt, you'll find a guide to the basic architecture of Red Hat Enterprise Linux OpenStack Platform and the step-by-step process to install it.

Learn more at redhat.com/training.

INSTALLING RED HAT ENTERPRISE LINUX OPENSTACK PLATFORM

PART I: THE OPENSTACK ARCHITECTURE

OpenStack includes the following services:

Nova Compute: A service that manages networks of virtual machines running on nodes, providing virtual machines on demand. Nova is a distributed component and interacts with Keystone for authentication, Glance for images, and Horizon for web interface. Nova is designed to scale horizontally on standard hardware, downloading images to launch instances as required. Nova compute uses libvirtd, qemu, and kvm for the hypervisor.

Glance (Image): A service that acts as a registry for virtual machine images, allowing users to copy server images for immediate storage. These images can be used as templates when setting up new instances.

OpenStack Networking: A service that provides connectivity between the interfaces of other OpenStack services, such as Nova. Due to OpenStack Networking's pluggable architecture, users can create their own networks, control traffic, and connect servers to other networks. Various networking technologies are supported.

Cinder (Volume): A service that manages storage volumes for virtual machines. This is persistent block storage for the instances running in Nova. Snapshots can be taken for backing up data, either for restoring data, or to be used to create new block storage volumes.

Note: This content is taken directly from our Red Hat® OpenStack® Administration (CL210) course.

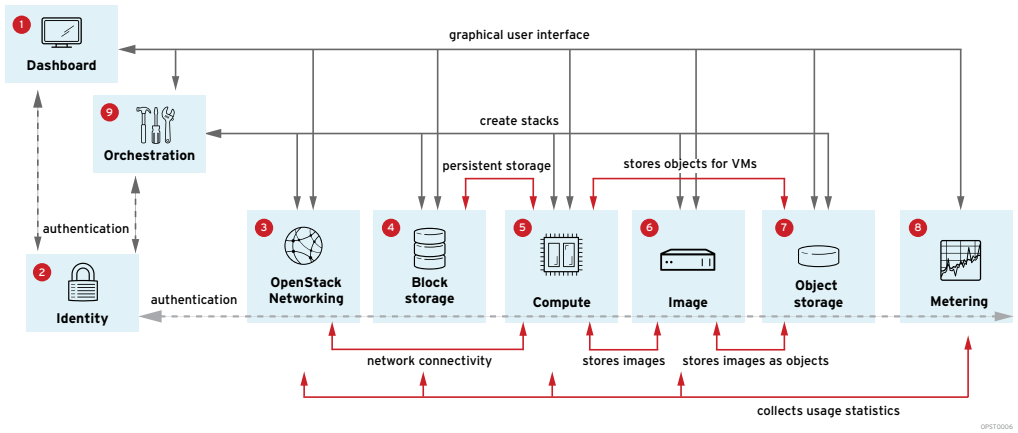
Swift (Object): A service providing object storage which allows users to store and retrieve files. Swift architecture is distributed to allow for horizontal scaling and to provide redundancy as failure-proofing. Data replication is managed by software, allowing greater scalability and redundancy than dedicated hardware.

Keystone (Identity): A centralized identity service that provides authentication and authorization for other services. Keystone also provides a central catalog of services running in a particular OpenStack cloud. It supports multiple forms of authentication including user name and password credentials, token-based systems, and Amazon Web Services style logins.

Horizon (Dashboard): A web-based interface for managing OpenStack services. It provides a graphical user interface for operations such as launching instances, managing networking and setting access controls.

Ceilometer (Metering): A centralized source for metering and monitoring data. This information will provide a means to bill the users of OpenStack.

Heat: A service to orchestrate multiple composite cloud applications using the AWS CloudFormation template format, through both a REST API and a CloudFormation-compatible Query API. The software integrates other core components of OpenStack into a one-file template system. The templates allow creation of most OpenStack resource types (such as instances, floating IPs, volumes, security groups, users, etc), as well as some more advanced functionality such as instance high availability, instance autoscaling, and nested stacks.



1. Horizon: Web browser user interface for creating and managing instances.
2. Keystone: Authentication and authorization framework.
3. OpenStack networking: Network connectivity as a service.
4. Cinder: Persistent block storage for runtime instances.
5. Nova: Scheduler for networks of virtual machines running on nodes.
6. Glance: Registry for virtual machine images.
7. Swift: File storage and retrieval.
8. Ceilometer: Metering engine for collecting billable meters.
9. Heat: Orchestration service for template based virtual machine deployments.

OPENSTACK TERMINOLOGY

OpenStack uses the following terminology:

- **Cloud controller**: The coordinating manager. All machines in the OpenStack cloud communicate with the cloud controller using the Advanced Message Queuing Protocol (AMQP). Red Hat Enterprise Linux OpenStack Platform uses the Qpid messaging daemon (qpidd) to provide AMQP.
- **Tenant:** Also known as a project in Horizon. A group of items (users, images, instances, network(s), volumes, etc.).
- **Compute node:** A hypervisor; any machine running the nova compute service. Most often, these only run the nova compute service.
- **Volume:** A persistent disk presented to the instance. Volumes can be attached to a single instance. These volumes are persistent, and can be attached to (or detached from) running instances. The cinder service is given an LVM volume group and volumes are created from this volume group (LVM logical volumes). A snapshot of the volume can be created, much as a snapshot is created of a logical volume.
- **Ephemeral disk:** A temporary disk used by an instance. When the instance is created, the ephemeral disk is created as a QCOW2 image in /var/lib/nova/instances/instance-00000000X/disk.local on the compute node. When the instance is terminated, this disk is removed. The first ephemeral disk normally appears as /dev/vdb.
- **Server or Instance:** A virtual machine.
- **Flavor**: The hardware associated with an instance. This includes, RAM, CPUs and disks.
- **Stack**: a group of instances built from a template. Template files are written in JSON. Stacks and the template files are used in the Heat orchestration service.
- **OpenStack Networking**: A software-defined networking service. Includes many plugins (e.g., Open vSwitch, Cisco UCS/Nexus) and allows software-defined network (SDN) and quality of service (QoS). The OpenStack Networking API uses the following abstractions to describe network resources:
  - **Network**: An isolated L2 segment, analogous to VLAN in the physical networking world.
  - **Subnet**: A block of v4 or v6 IP addresses and associated configuration state.
  - **Port**: A connection point for attaching a single device, such as the NIC of a virtual server, to a virtual network. Also describes the associated network configuration, such as the MAC and IP addresses to be used on that port.

PART II: INSTALL RED HAT ENTERPRISE LINUX OPENSTACK PLATFORM

Software requirements

- To deploy Red Hat Enterprise Linux OpenStack Platform, you need to have at least two machines with Red Hat Enterprise Linux 64-bit, at least version 6.4. One machine can act as a dedicated cloud controller node and the second machine can act as a Nova compute node. In the field, a minimum of two Nova compute nodes are recommended.
- Make sure your machines have their clocks synced via Network Time Protocol (NTP).

HARDWARE REQUIREMENTS

Red Hat Enterprise Linux OpenStack Platform Cloud Controller Node hardware requirements

PROCESSOR:

64-bit x86 processor with support for the Intel® 64 or AMD64 CPU extensions, and the AMD-V™ or Intel VT® hardware virtualization extensions enabled

MEMORY:

2 GB RAM

DISK SPACE:

50GB (1TB recommended for a realistic environment capable of hosting multiple instances of varying sizes)

NETWORK:

2x 1 Gbps Network Interface Card (NIC)

WORKSHOP

Note: This course excerpt is just the beginning of a full, deep dive into administrating Red Hat Enterprise Linux OpenStack Platform. Further chapters in this course show you how to:

- Deploy each Red Hat Enterprise Linux OpenStack platform service manually.
- Manage users and projects.
- Deploy instances and use Heat to deploy and customize instances.

INSTALLATION OF RED HAT ENTERPRISE LINUX OPENSTACK PLATFORM

Red Hat Enterprise Linux OpenStack Platform features a tool to help with the installation called `packstack`.

1. The openstack-packstack includes the `packstack` to quickly deploy Red Hat Enterprise Linux OpenStack Platform either interactively, or non-interactively by creating and using an answer file that can be tuned. Install the openstack-packstack package on `serverX`, using `yum`.
```
[root@serverX ~]# yum install -y openstack-packstack
```
2. Before we start with Red Hat Enterprise Linux OpenStack Platform deployment via `packstack`, SSH keys are generated for easy access to the Nova Compute Nodes from the Cloud Controller Node. We will not include a passphrase because the installation will require this passphrase hundreds of times during this installation.
```
[root@serverX ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): Enter
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
...
```
3. Explore some of the options of the packstack command.
```
[root@serverX ~]# packstack -h | less
```
4. The recommended way to do an installation is non-interactive, because this way the installation settings are documented. An answer file with default settings can be generated with the `packstack` command.
```
[root@serverX ~]# packstack --gen-answer-file /root/answers.txt
```
5. Before we can start the actual installation we should edit the /root/answers.txt and ensure the following items are configured:
```
CONFIG_SSH_KEY=/root/.ssh/id_rsa.pub
CONFIG_NTP_SERVERS=192.168.0.254
CONFIG_HORIZON_SSL=y
```
6. Answer file settings for the Cloud Controller Node:
   Setting: CONFIG_SSH_KEY=/root/.ssh/id_rsa.pub
   Purpose: Configure the SSH pubkey to be deployed on every machine that is related to Red Hat Enterprise Linux OpenStack Platform Cloud.

RED HAT ENTERPRISE LINUX OPENSTACK PLATFORM COMPUTE NODE HARDWARE REQUIREMENTS

PROCESSOR:

64-bit x86 processor with support for the Intel® 64 or AMD64 CPU extensions, and the AMD-V™ or Intel VT® hardware virtualization extensions enabled.

MEMORY:

2GB RAM minimum (add additional baased on amount of memory you intend to make available to virtual machine instances.)

DISK SPACE:

50GB (1TB recommended for a realistic environment capable of hosting multiple instances of varying sizes)

NETWORK:

2x 1 Gbps Network Interface Card (NIC)

Setting: CONFIG_NTP_SERVERS=192.168.0.254
Purpose: Configure the ntp servers for time synchronization.

Setting: CONFIG_HORIZON_SSL=y
Purpose: Enable use of SSL for Horizon.

7. Now it is time to do the actual deployment of the Red Hat Enterprise Linux OpenStack Platform Cloud Controller using the answer file we just prepared:

```
[root@serverX ~]# packstack --answer-file /root/answers.txt
Welcome to Installer setup utility
Installing:
Clean Up... [DONE]
Setting up ssh keys...root@192.168.0.X+100's password: redhat
...
```

USE THE HORIZON WEB INTERFACE

LOGGING INTO THE HORIZON WEB INTERFACE

The Horizon web interface allows for management of the entities for creating new projects with OpenStack. The web frontend is accessible at `https://serverX.example.com/dashboard`. You can login as admin with the password found in `/root/keystonerc_admin as the OS_PASSWORD` variable.

WORKING WITH TENANTS

A tenant describes a project with an assigned number of OpenStack users and resources. This enables multiple projects to use a single cloud without interfering with each other in terms of permissions and resources.

A quota of resources are already set when a new tenant is created. The quota includes the amount of VCPUs, instances, RAM, and floating IPs that can be assigned to instances. Tenants can be added, modified, and deleted in Horizon with a few clicks.

WORKSHOP

CREATE A TENANT IN HORIZON

Time to create a new project with the following requirements:

PROJECT PARAMETERS

PARAMETER	VALUE
Name	project1
Description	Project for project1
Quota	4 VCPUs, 4 instances, 4096 MB RAM, 2 floating IPs

1. On desktopX.example.com, open **Firefox** and browse to `https://192.168.0.X+100/dashboard`. Add an exception for the self-signed certificate.
2. Find the admin password in the `/root/keystonerc_admin` file on serverX.example.com, and use the password to login to Horizon.

3. To create the new tenant go to the **Admin** tab in the left pane and click on the **Projects** link.
4. Click on the **Create Project** button.
5. Add the name and description as above. Go to the Quota tab and set the **quotas** as above.
6. Click the **Create Project** button.

MANAGE FLAVORS

When a new instance gets deployed a flavor must be selected that outlines the virtual machine hardware specifications. The parameter includes the number of VCPUs, RAM, and disk space used by the instance.

WORKSHOP

CREATE A FLAVOR IN HORIZON

PROJECT PARAMETERS

PARAMETER	VALUE
Name	m2.tiny
VCPUs	1
RAM	1024 MB
Root disk	20 GB
Ephemeral disk	2 GB
Swap disk	512 MB

1. To create the new flavor go to the **Admin** tab in the left pane and click on the **Flavors** link.
2. Click on the **Create Flavor** button.
3. Enter the informations as given above.
4. Press the **Create Flavor** button.

USER MANAGEMENT IN HORIZON

Users can be added, modified, and deleted with the Horizon web interface. OpenStack permission management is role-based. By default there are two predefined roles: a member role that gets attached to a tenant and an administrative role to enable users other than the admin to administrate the cloud.

WORKSHOP

CREATE A USER IN HORIZON

Lets create a new user with the following parameters:

USER PARAMETERS

PARAMETER	VALUE
User name	user1
Email	root@desktopX.example.com
Password	redhat
Primary project	project1
Role	Member

1. Go to the Admin tab in the left pane and click on the **Users** link.
2. Click on the **Create User** button.
3. Enter the information as given above.
4. Click on the **Create User** button.
5. Logout as `admin` and login as the newly-created `user1` user. Recall that the password is `redhat`.

LAUNCH AN INSTANCE IN HORIZON

Horizon can be used to launch and manage instances. Before launching an instance, you will probably want to create and upload an image, configure a security group, create an SSH keypair and allocate floating IP addresses. The following workshop will walk you through the steps of preparing to launch an instance, and then creating an instance.

WORKSHOP

LAUNCH AN INSTANCE IN HORIZON

Create an instance using the following parameters:

INSTANCE PARAMETERS

PARAMETER	VALUE
Image name	small
Image location	`http://instructor.example.com/pub/materials/small.img`
Image format	`QCOW2`
Image settings	No minimum disk, minimum 1024 MB RAM, public
Private network name	net1
Private subnet information	• Subnet Name: `subnet1` • Network Address: `192.168.32.0/24` • IP Version: IPv4 • Gateway IP: Leave blank (not disabled)

PARAMETER	VALUE
Public network name	net2
Public subnet information	• Subnet Name: `subnet2` • Network Address: `172.24.X.0/24` • IP Version: IPv4 • Gateway `IP: 172.24.X.254`
Public subnet details	• Enable DHCP: deselected • Allocation Pools: `172.24.X.1,172.24.X.100`
Router name	router1
IP to allocate for the Instance	172.24.X.2
Security group name	sec1
Security group description	Web and SSH
Security group permissions	Allow TCP/22 and TCP/443 from CIDR 0.0.0.0/0, and TCP/80 from the `sec1` source group
SSH keypair name	key1
Instance image	small
Instance name	small
Instance flavor	m2.tiny
Instance keypair	key1
Instance security group	sec1
Instance floating IP address	172.24.X.2
Volume name	myvol1
Volume description	myvol1 volume
Volume size	2 gb
Volume snapshot name	myvol1-snap1
Volume snapshot description	myvol1-snap1 snapshot
Attached volume device	`/dev/vdd`

Note: Making the image "public" will allow users in other tenants to use this image.

1. The first thing to do to create a new machine is to import a new image. In the **Project** tab on the left pane select the **Images & Snapshots** link. Click on the **Create Image** button. Enter small for the name, enter `http://instructor.example.com/pub/materials/small.img` for the image location, and `QCOW2` for the image format. Leave the minimum disk blank, enter `1024` for the minimum RAM, and select the **Public** checkbox. Press the **Create Image** button.

2. Next, we configure networking. In the Project tab on the left pane, select the **Networks** link. Press the **Create Network** button. Enter `net1` for the network name. In the **Subnet** tab, enter `subnet1` for the name, `192.168.32.0/24` for the network address, and IPv4 for the IP version. Leave the other options as they are and press the **Create** button.
   Press the **Create Network** button again. Enter the public network name as `net2`. Browse to the **Subnet** tab. Enter the public subnet name as `subnet2`, the network address as `172.24.X.0/24`, IPv4 for the IP version and `172.24.X.254` as the gateway IP. Browse to the **Subnet Detail** tab. Deselect the **Enable DHCP** checkbox and enter `172.24.X.1,172.24.X.100` for the allocation pool. Press the **Create** button.
   In the **Project** tab on the left pane select the **Routers** link. Press the **Create Router** button. Enter the router name as `router1` and press the **Create** button.

3. Setting an external network can only be set as an administrator. Sign out as the user1 user and sign in as admin. In the Admin tab in the left pane, click on the Networks link. Click on the Edit Network button in the public (net2) network row. Select the External Network checkbox and click on the Save Changes button.

4. Sign out as the `admin` user and sign in as `user1`. In the **Project** tab in the left pane, click on the Routers link. Press the **Set Gateway** button in the `router1` row. In the **External Network** menu, choose `net2`. Press the **Set Gateway** button.

Note: The interfaces attached to the router have been assigned the first address in the range (192.168.32.1 and 172.24.X.1).

5. Click on the `router1` link. Press the **Add Interface** button. In the Subnet menu, select `net1: 192.168.32.0/24 (subnet1)`. Press the **Add Interface** button.
   Verify that both networks are attached to the router. In the **Project** tab in the left pane, click on the **Network Topology** link. `net1` and `net2` should both connect to the `router1` router and both network ranges should be displayed.

6. Next we allocate a floating IP. In the **Project** tab in the left pane, click the **Access & Security** link. Choose the **Floating IPs** tab and click on the **Allocate IP to Project** button. Use the `net2` pool and click on the **Allocate IP** button to allocate the above given floating IP address.

7. To determine the network access permissions we have to setup a security group for our instance. In the **Project** tab in the left pane, select the **Access & Security** link. Choose the **Security Group** tab and click on the **Create Security Group** button. Enter `sec1` for the name and `Web and SSH` for the description. Click on the **Create Security** Group button.
   Click on the **Edit Rules** button for the `sec1` security group. Press the **Add Rule** button. Choose **TCP** as the protocol, enter 22 as the port, and leave the **Open**, **Source Group** and **CIDR** as default. Click on the **Add** button. Click on the **Add Rule** button again. Enter 443 in the Port box and press the Add button. Click on the Add Rule button once more. Enter 80 in the **Port** box and choose `Security Group` in the **Source** menu. Choose sec1 (current) from the **Security Group** drop-down menu. Press the **Add** button.

8. The next step is to create a ssh key pair for the instance. In the **Project** tab in the left pane, select the **Access & Security** link. Choose the **Keypairs** tab and press the **Create Keypair** button. Enter ***key1*** for the name and press the **Create Keypair** button. Save the `key1.pem file` to the default location, which should be `in /home/student/Downloads/key1.pem on desktopX.example.com`.

9. Using Qemu within the virtual machine will likely cause the OpenvSwitch agent to overrun the polling time, so set the polling interval to 20 seconds and restart the `quantum-openvswitch-agent service.`

```
[root@serverX ~]# openstack-config --set
/etc/quantum/plugins/openvswitch/ovs_quantum_plugin.ini AGENT polling_
interval 20
[root@serverX ~]# service quantum-openvswitch-agent restart
```

10. Before attaching `eth0` to the `br-ex` bridge, configure the `br-ex` network device configuration file.

```
[root@serverX ~]# cp /etc/sysconfig/network-scripts/ifcfg-eth0 /root/
[root@serverX ~]# cp /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/syscon-
fig/network-scripts/ifcfg-br-ex
```

Remove everything but the `DEVICE, HWADDR` and `ONBOOT` settings from `/etc/sysconfig/network-scripts/ifcfg-eth0` so that it looks like the following:

```
DEVICE=eth0
HWADDR=52:54:00:00:00:XX
ONBOOT=yes
```

Leave the MAC address as it is. Do not change it. The XX is the desktopX number in hexidecimal. In the `/etc/sysconfig/network-scripts/ifcfg-br-ex` file, remove the `HWADDR` line and change the device name to `br-ex`. The `/etc/sysconfig/network-scripts/ifcfg-br-ex` file should contain the following:

```
DEVICE=br-ex
IPADDR=192.168.0.X+100
PREFIX=24
GATEWAY=192.168.0.254
DNS1=192.168.0.254
SEARCH1=example.com
ONBOOT=yes
```

11. Once you have verified the network files contain the correct information, add the `eth0` network device to the `br-ex` bridge and restart the network.

```
[root@serverX ~]# ovs-vsctl add-port br-ex eth0 ; service network restart
```

12. Now we are nearly there. Lets create the instance! Back in the Horizon dashboard, in the **Project** tab in the left pane, select the **Instances** link. Press the **Launch Instance** button. In the **Details** tab, select `small` as the image, enter `small` as the name and choose the `m2.tiny` flavor. In the **Access & Security** tab, enable the `key1` keypair and `sec1` security group, and deselect the `default` security group. In the **Networking** tab, press the + button next to the `net1` network. Press the **Launch** button.

Once the networking has been created for the instance, open the **Actions (More)** drop-down menu and select **Associate Floating IP**. Choose the `172.24.X.2` floating IP address, and choose the small instance, then press the **Associate** button.

13. Click on the **small** instance link. Click on the **Console** tab, then click on the **Click here to show only console** link. Watch the virtual machine boot. This may take several minutes to boot.

14. For a simple networking verification of our setup try to `ssh` to `172.24.X.2` from `desktopX.example.com`.

```
[student@desktopX ~]# chmod 600 /home/student/Downloads/key1.pem
[student@desktopX ~]# ssh -i /home/student/Downloads/key1.pem
root@172.24.X.2
The authenticity of host '172.24.X.2 (172.24.X.2)' can't be established.
RSA key fingerprint is aa:bb:cc:dd:ee:ff:00:11:22:33:44:55:66:77:88:99.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.24.X.2' (RSA) to the list of known hosts.
[root@small ~]#
```

15. Now we want to create a volume. Back in the dashboard, click on the **Volume** link on the left pane and then click on the **Create Volume** button. Enter `myvol1` as the volume name, `myvol1 volume` description, and 2 GB size. Press the **Create Volume** button.

Note: A snapshot of a volume can only be generated if it is not attached to a running instance.

16. Now we create a snapshot of the volume. While still in the **Volume** section on the left pane, click on the **More** drop down menu on the right side of the row that describes the new volume and select **Create Snapshot**. Enter `myvol1-snap1` as the snapshot name and `myvol1-snap1 snapshot` as the description. Press the **Create Volume Snapshot** button.

Note: While we provide the device name for the volume that is attached, this functionality currently does not work as expected. The device will appear like any newly attached scsi or sata device in the target instance and does not honor the provided device name.

17. Now we attach the volume to the running instance. Browse to the **Volume** section on the left pane and click on the **Edit Attachments** button. Select the `small` instance we just created in the **Attach to Instance** drop down menu, then enter the device `name /dev/vdd` in the input box. Press the **Attach Volume** button.

RELATED TRAINING AND CERTIFICATION

Want to learn more about how to install, configure, and maintain Red Hat Enterprise Linux OpenStack Platform? Our hands-on, lab-based course can teach you much more. Then prove your skills by achieving your certification.

COURSE

Red Hat OpenStack Administration[1]

- Classroom with exam (CL211)
- Classroom (CL210)
- Virtual classroom (CL210VT)*
- Red Hat Online Learning (CL210R)**
- On-site (private, team training)

CERTIFICATION

Red Hat Certified System Administrator in Red Hat OpenStack[2]

1 *http://www.redhat.com/training/paths/openstack*

2 *http://www.redhat.com/training/certifications/rhcoe-openstack-iaas/*

ABOUT THE AUTHORS

Forrest Taylor joined Red Hat in 2004 as an instructor in Denver, Colorado, and has since become a curriculum manager based in Raleigh, North Carolina. Forrest authored Red Hat's Red Hat Enterprise SELinux Policy Administration (RHS429) manual and the Red Hat Certificate of Expertise in SELinux Policy Administration (EX429) exam. He was also one of the first persons to achieve Red Hat Certified Security Specialist (RHCSS) status. He has taught all Red Hat Certified Engineer and Red Hat Certified Architect-level classes. Outside the computer world, he enjoys playing volleyball and basketball, and spending time with his kids.

Rudolf Kastl has worked at Red Hat since 2006 as an instructor and consultant and recently became a curriculum manager. When he started with Linux, he quickly became involved in the community by building third-party RPM packages and publishing them for his and for other users' convenience. His spare time is allocated with tasks around open source software and Fedora.

* Virtual classroom training is currently only available in North America and Europe

** Red Hat Online Learning is currently only available in North America, Europe, and certain Asia-Pacific countries

ABOUT RED HAT

Red Hat is the world's leading provider of open source solutions, using a community-powered approach to provide reliable and high-performing cloud, virtualization, storage, Linux, and middleware technologies. Red Hat also offers award-winning support, training, and consulting services. Red Hat is an S&P company with more than 70 offices spanning the globe, empowering its customers' businesses.

facebook.com/redhatinc
@redhatnews
linkedin.com/company/red-hat

redhat.com
#1821807_V2_0314

NORTH AMERICA
1 888 REDHAT1

EUROPE, MIDDLE EAST, AND AFRICA
00800 7334 2835
europe@redhat.com

ASIA PACIFIC
+65 6490 4200
apac@redhat.com

LATIN AMERICA
+54 11 4329 7300
info-latam@redhat.com