11. Comprehensive Practice

Let's put our knowledge together from previous chapters and solve practical use cases! This chapter will not explicitly lay out the details and commands for you, it will only provide the high level workflow for you to achieve each task.

If you have questions, or accidentally sabotage your environment ask the instructor for help.

Chapter Details	
Chapter Goal	Put your knowledge and prowess to the test
Chapter Sections	*11.1. Comprehensive Task Overview* *11.2. Create a New SSH User* *11.2.1. Hints for Creating a new SSH User* *11.3. Create an OpenStack User and Project (Tenant)* *11.3.1. Hints for creating OpenStack user and project* *11.4. Adding a Compute Node* *11.4.1. Hints for Adding a Compute Node* *11.5. Stacklight Monitoring* *11.6. Stacklight Monitoring Hints*

11.1. Comprehensive Task Overview

In this chapter, you will dive in to some of the key operations tasks when managing your infrastructure. If you are unsure about how to approach a certain step, see if you can solve it by viewing the hint and references first. The instructor will provide you with answers and go over any questions you have once everyone has had the chance to work through the exercise.

11.2. Create a New SSH User

A new member has joined the infrastructure management team (congratulations!) it's your task to create a new SSH User for this member. Follow the general directions below and create a SSH user for *sally*.

Step 1 Generate an SSH keypair for Sally on the host (*kvm01*) The keypair files should be called *sally* and *sally.pub*

Step 2 Create a new user in the cluster model called *sally.yml* using the *sally.pub* key generated from the first step. Use *sally@domain.tld* for the email address.

Step 3 Apply the change to the model to create the SSH user

Step 4 Test your change by logging in to *cfg01* node as *sally* user with the *sally* key from the *kvm01* node

11.2.1. Hints for Creating a new SSH User

Hint 1 To create a keypair, use the linux command *ssh-keygen -f <filename>*

Hint 2 View the Salt Formula for *openssh* for model formatting: `https://github.com/salt-formulas/salt-formula-openssh`

Hint 3 Examples of SSH user model can be found in the following directory:

```
root@cfg01:~# cd /srv/salt/reclass/classes/system/openssh/server/team/members
```

Hint 4 The process of adding a new change to the model is to 1) run the *sync_all* saltutil, 2) Apply the *reclass* state to the *cfg01* node, and 3) apply the state which is relevant to your changes

Hint 5 The states to apply are *linux.system.user* and *openssh.server* on all nodes

Hint 6 To log-in to *cfg01* using a SSH key, use the following command:

```
root@kvm01:~# ssh -i <private-key> <user>@cfg01
```

11.3. Create an OpenStack User and Project (Tenant)

Chandler is one of your new OpenStack customer who wishes to be part of a project called *Friends*. Create a new project called *Friends* and assign *Chandler* as the user with *admin* role.

Step 1 Add a new Reclass cluster model for Keystone with the following information:

- Project name: *Friends*
  - description: *Project for friends*
- User name: *Chandler*
  - password: *stack*
  - email: *chandler@bing.com*

Step 2 Apply your changes and check that you can log-in to OpenStack UI as *Chandler* user with *stack* password

11.3.1. Hints for creating OpenStack user and project

Hint 1 Refer to the parameters of the *keystone:client* pillar to formulate your yaml file:

```
root@cfg01:~# salt 'ctl01*' pillar.items keystone:client
# `identity` is the name of our Keystone server. Under the identity
# pillar definition, you will find the existing admin project
# format
```

Or refer to the following file

```
root@cfg01:~# cat /srv/salt/reclass/classes/system/keystone/client/core.yml
```

Hint 2 Horizon UI is located at `https://192.168.2.80/auth/login/`

11.4. Adding a Compute Node

Your IT team has decided to expand tenant capcity and it's up to you to add a new compute node! Take the following information and add a new compute node to the lab environment.

Keep in mind that a *compute node* in your environment is a nested virtual machine. Adding a new node means defining a new machine for *kvm* model.

Do your work in your *developer* directory located at `/root/mymodel`. Then follow the commit, approve, submit process as you would in production.

Step 1 Edit the *classes/cluster/lab28/infra/kvm.yml* model to define a new compute node VM: *cmp002.trainings.local*.

Step 2 Edit the necessary model to define the infrastructure configuration for your compute node in the *classes/cluster/lab28/infra/config.yml* model.

Step 3 Submit your changes to Gerrit for review and approve it to merge with master

Step 4 Pull the latest submitted changes in the /srv/salt/reclass directory

Step 5 Run the *reclass.storage* state on the *cfg01* node to generate the newly pulled Reclass hierarchy

Step 6 Run the *salt.control* state on the *kvm01* node to create and initialize your compute node VM

Step 7 Run the *Deploy - OpenStack Compute node* build in Jenkins targeting the *cmp002.trainings.local* node you created.

Step 8 Check the list of hypervisors in OpenStack and ensure that you can see the new compute node

Step 9 Create an OpenStack VM and schedule it on the *cmp002* node

11.4.1. Hints for Adding a Compute Node

Hint 1: If you are not sure of the value of an existing *_param:* do a pillar item list of that parameter:

```
root@cfg01:~# salt 'cfg01*' pillar.items _param:openstack_compute_rack01_single_subnet
cfg01.trainings.local:
    ----------
    _param:openstack_compute_rack01_single_subnet:
        192.168.2
```

Hint 2: Reference the existing compute node definition to get an idea of how to format a new node, or check the official Mirantis github at `github.com/Mirantis` and navigate to the reclass model repository. (Adding a new node is an *infrastructure* operation. Thus, look for *infra* related directory)

Hint 3: Official documentation on adding a compute node can be found here: https://docs.mirantis.com/mcp/q2-18/mcp-operations-guide/openstack-operations/manage-compute-nodes/add-compute-node.html

Hint 4: Make sure to run *salt '*' saltutil.sync_all* before running the pipeline

Hint 5: OpenStack UI (Horizon) is located at `192.168.2.80/auth/login` for your use if you are more comfortable using Horizon. Log-in credentials are *admin* / *r00tme*

Hint 6: *–availability-zone* option in OpenStack CLI will allow you to choose the host to launch the instance

11.5. Stacklight Monitoring

Step 1 View your current Nova Hypervisor setup and note down the CPU utilization of *cmp002*

Step 2 Configure the alarm to trigger CPU utilization of over 80% on *cmp002* with a label to route to email

Step 3 Run a busy command (*$ yes > /dev/null &*) twice on the compute node VM to generate high CPU utilization

Step 4 Check that the notification trigger has worked by navigating to Prometheus UI and viewing the Alerts tab

Step 5 Run *# killall yes* to kill the busy command on the new compute node

11.6. Stacklight Monitoring Hints

Hint 1 The UI for Grafana is located at `https://192.168.2.80:8084` and can be accessed with *admin* user and *r00tme* password

Hint 2 The Prometheus UI is located at `http://192.168.2.70:15010` and can be accessed with *admin* user and *r00tme* password

Hint 3 To search for a metric query, start typing in a keyword in the search bar of Prometheus UI

Hint 4 The Alertmanager UI is located at `http://192.168.2.70:15011`

Hint 5 By selecting a graph on Grafana and clicking *Edit* you can view the query used to generate that graph. Select a CPU utilization graph and edit it to get an idea of how to formulate your query.

Congrautlations, you have finished the *11. Comprehensive Practice* chapter!

Checkpoint:

- Create a new SSH user using Reclass model
- Create an OpenStack tenant using Reclass
- Add a new Compute Node using Reclass, Salt, and DriveTrain
- Monitor your creation using Stacklight