

OpenStack Heat Workshop

Instructor Led | Duration: 3 Days | Course Number: NWV_413



OpenStack is a very popular open source cloud enablement system for creating private and public clouds. OpenStack software controls large pools of compute, storage, and networking resources in the cloud deployment. Heat is the service for orchestrating resources in an OpenStack deployment. It implements an orchestration engine to launch multiple composite cloud applications based on YAML or JSON templates. This workshop provides a thorough understanding of Heat Orchestration Templates (HOT). Using an OpenStack system, students will develop, enhance and analyze several heat templates of increasing complexity that will work together to understand and apply heat templates simulating practical deployments of OpenStack in a Telco environment.

Intended Audience

This session is intended for team members in technology organizations who want to work with OpenStack Heat and create and use Heat Templates (YAML format) for orchestration.

Learning Objectives

After completing this course, the student will be able to:

- Read and interpret Heat Orchestration Templates (HOT)
- Design and develop Heat Orchestration Templates
- Analyze, enhance and validate outcomes of HOT templates
- Create Heat Stacks in an OpenStack deployment
- Create nested Heat templates
- Articulate the affects of global environments as they relate to launching Heat Stacks
- Practice the scripting of VM actions and configuration at boot time

Suggested Prerequisites

- [NWV_304] SDN and NFV Architecture and Operations
- [NWV_405] OpenStack Workshop for SDN and NFV

Required Equipment

- Students will need a laptop with a web browser and Windows Remote Desktop installed
- It is recommended for the students to use an additional monitor

Course Outline

0. Prologue

- 0.1. OpenStack Heat high level review

1. Template Structure

- 1.1. Supported formats and syntax
- 1.2. Versions and compatibility
- 1.3. The What vs. the How

Exercise: Discover IaaS capabilities

2. Input and Intrinsic Functions

- 2.1. Input parameters
- 2.2. Parameter groups
- 2.3. Parameter constraints
- 2.4. Intrinsic functions

Exercise: Order and validate input,

Exercise: Use intrinsic functions

3. Heat Resources

- 3.1. Resource section
- 3.2. Resource types
- 3.3. Properties and attributes
- 3.4. Sample resource types

Exercise: Create servers, volumes, LB pool, VIP to server pool with floating IPs

4. Template Output

- 4.1. Output section
- 4.2. Use of parameters and attributes

Exercise: Output resource run-time values

5. Template Composition

- 5.1. Nested stacks in templates
- 5.2. Build, launch, manage nested HOTs
- 5.3. Resource dependencies
- 5.4. Information flow in nested stacks
- 5.5. Resource_facade

Exercises: Explore, analyze, launch and manage nested templates

6. Environments

- 6.1. Global and effective environments
- 6.2. Influencing parameters and parameter defaults
- 6.3. Resources registry and override

Exercise: Resource Groups, Policies, and triggers to scale Resources defined in external environments

7. Virtual Software Configuration

- 7.1. Consideration and mechanisms
 - 7.2. Various configuration tools options
 - 7.3. Config-init and config-drive
- Exercise: VM configuration using configuration scripts**

8. Dynamic Service Scaling

- 8.1. Autoscaling of virtual functions
- Exercise: A comprehensive example with scale out/in**