Module 8: Understanding Orchestration

Agenda

* What is Orchestration
* Heat Components
* Terminologies of Heat
* Writing HOT template

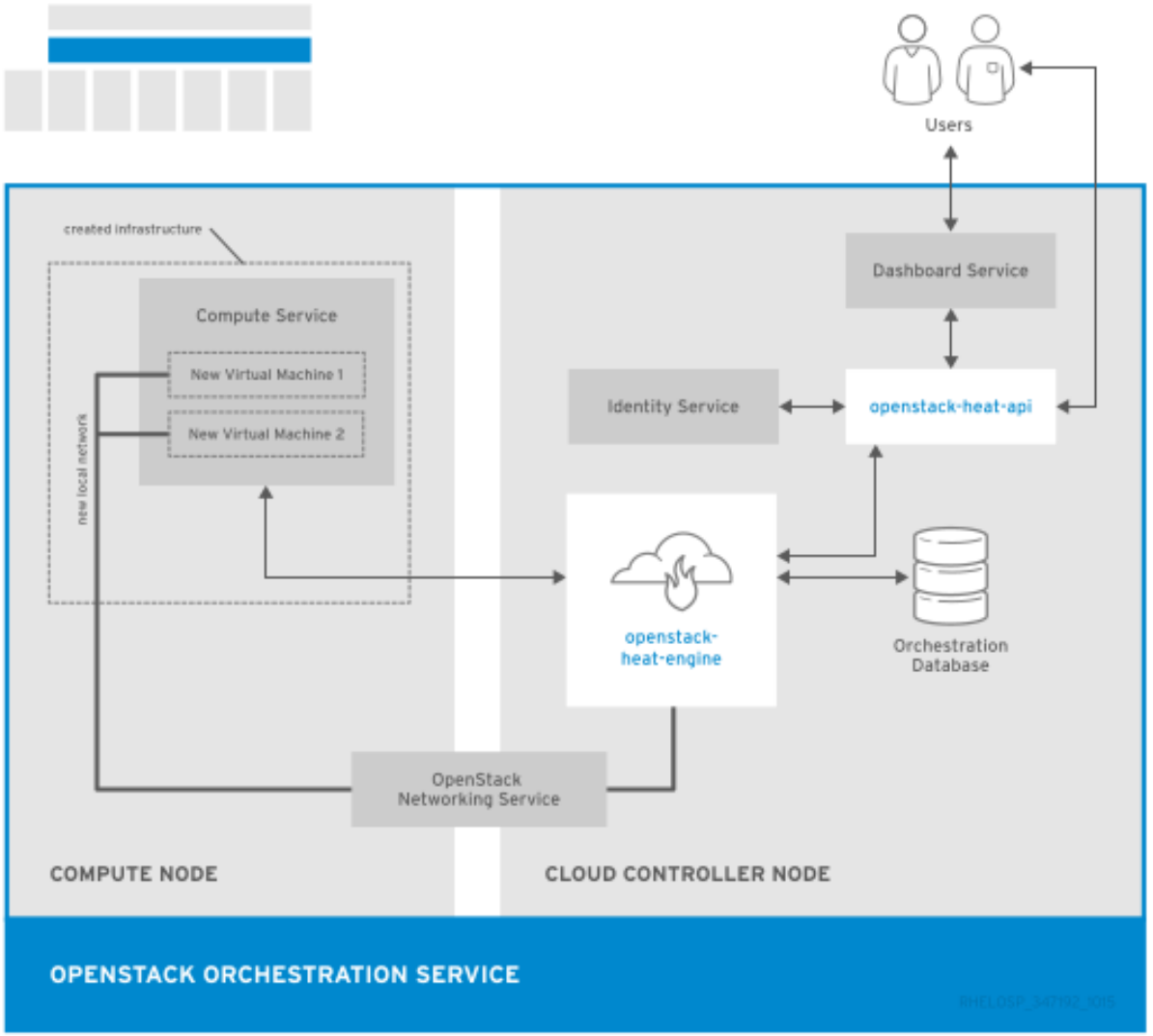
Orchestration

Orchestration automates provisioning of cloud resources, including virtual networks, storage, and servers.

It gives developers and system administrators an easy way to create and manage cloud resources in an orderly and predictable fashion.

The Orchestration service (heat) provides OpenStack with orchestration functionality.

Heat components



Heat Components

* The heat-api component implements an OpenStack-native RESTful API. This components processes API requests by sending them to the Heat engine via AMQP.

* The heat-api-cfn component provides an API compatible with AWS CloudFormation, and also forwards API requests to the Heat engine over AMQP.

* The heat-engine component provides the main orchestration functionality.

Heat Terminologies

Stack: In Heat parlance, a stack is the collection of objects—or resources—that will be created by Heat. This might include instances (VMs), networks, subnets, routers, ports, router interfaces, security groups, security group rules, auto-scaling rules, etc.

Template: Heat uses the idea of a template to define a stack. If you wanted to have a stack that created two instances connected by a private network, then your template would contain the definitions for two instances, a network, a subnet, and two network ports. Since templates are central to how Heat operates, I'll show you examples of templates in this post.

Heat Terminologies

Parameters : Input parameters provided when deploying from the template.

Resources : The infrastructure elements to deploy, such as virtual machines or network ports.

Outputs : Output parameters dynamically generated by Heat, available through the command-line interface, as well as the dashboard.

Heat Terminologies

HOT: Short for Heat Orchestration Template, HOT is one of two template formats used by Heat. HOT is not backwards-compatible with AWS CloudFormation templates and can only be used with OpenStack. Templates in HOT format are typically YAML

CFN: Short for AWS CloudFormation, this is the second template format that is supported by Heat. CFN-formatted templates are typically expressed in JSON