技 术 文 件

技术文件名称：Heat支持HA功能研究报告

技术文件编号：

版 本：V1.0

拟 制

审 核

会 签

标准化

批 准

中兴通讯股份有限公司

**修改记录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 文件编号 | 版本号 | 拟制人/  修改人 | 拟制/修改日期 | 更改理由 | 主要更改内容  （写要点即可） |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 注1：每次更改归档文件（指归档到事业部或公司档案室的文件）时，需填写此表。  注2：文件第一次归档时，“更改理由”、“主要更改内容”栏写“无”。 | | | | | |

目 录

[1 引言 4](#_Toc406426706)

[1.1 编写目的 4](#_Toc406426707)

[1.2 预期的读者和阅读建议 4](#_Toc406426708)

[1.3 文档约定 4](#_Toc406426709)

[2 术语、定义和缩略语 4](#_Toc406426710)

[2.1 术语、定义 4](#_Toc406426711)

[2.2 缩略语 4](#_Toc406426712)

[3 概述 5](#_Toc406426713)

[3.1 环境说明 5](#_Toc406426714)

[3.2 功能概述 5](#_Toc406426715)

[3.3 Heat简介 5](#_Toc406426716)

[4 组件说明 7](#_Toc406426717)

[4.1 Cloud-init 8](#_Toc406426718)

[4.2 Heat-cfntools 8](#_Toc406426719)

[4.3 Boto 9](#_Toc406426720)

[5 模板分析 10](#_Toc406426721)

[6 参考文献 10](#_Toc406426722)

# 引言

## 编写目的

本文描述的对象是Heat支持HA功能研究报告，通过对设计的思路、系统的构架、组成本系统的组件间的协作、组件的具体要求和组件间接口的详细描述，来满足上游需求，为集成测试设计、编写用户文档等工作提供依据。

## 预期的读者和阅读建议

本文档预期的读者和阅读建议见表1.1。

表1.1

| 读者分类 | 阅读重点 | 备注 |
| --- | --- | --- |
| 系统工程师（负责开发软件需求） | 构架说明、协作说明、组件说明、接口说明 |  |
| 系统工程师（负责软件概要设计） | 构架说明、协作说明、组件说明、接口说明 |  |
| 测试工程师 | 构架说明、协作说明、组件说明、接口说明 |  |

## 文档约定

无

# 术语、定义和缩略语

## 术语、定义

本文使用的专用术语、定义见表2.1。

表2.1 术语

| 术语/定义 | 英文 | 说 明 |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

## 缩略语

本文使用的专用缩略语见表2.2。

表2.2 缩略语

| 缩略语 | 原文 | 中文含义 |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 概述

## 环境说明

PC机一台，操作系统安装Red Hat 7，然后安装opencos i375版本，类型是all-in-one；使用fedora 20创建虚拟机；

## 功能概述

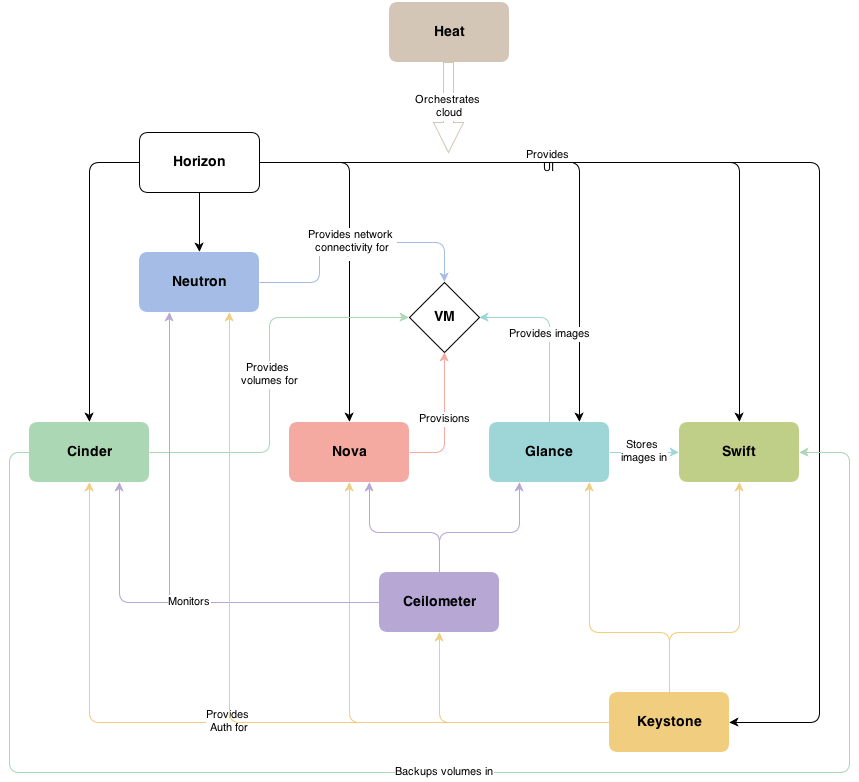
Heat中的虚拟机HA功能，是在虚拟机中的服务状态异常时，重新创建此虚拟机及其关联组件。

具体到实现上，是在HEAT模板创建的虚拟机中，通过传入的配置文件，设置监控其中的某些服务的状态，实现对虚拟机本身工作状态的监控；在虚拟机内部，当所监控的服务处于inactive状态时，启动对应的服务，使其恢复到active状态，并删除所在的整个虚拟机及其关联组件，然后重新创建虚拟机及其关联组件。

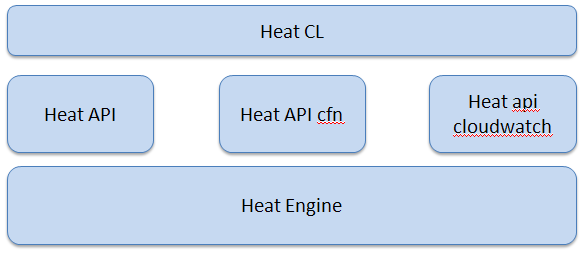
## Heat简介

1. Heat在系统中的位置

Heat是基于模板，方便部署openstack的一个模块；利用Heat，可以把nova、neutron、cinder、glance、swift、ceilometer等资源，集合到一个或多个模板中，实现用模板来部署openstack，以快速配置openstack，还可以在另外的环境上重复利用。



1. Heat模块结构



Heat CLI模块是heat命令行工具，与heat-api通信。

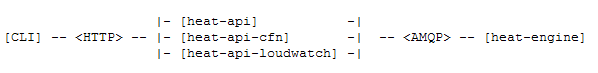
Heat-api模块提供openstack自研的REST API，接收其它模块的访问请求，并把这些请求通过RPC传递给heat-engine；

Heat-api-cfn模块是为了兼容AWS CloudFormation；用于接收Amazon CloudFormation API请求，并将其翻译成openstack能识别的请求，通过RPC传送给heat-engine；

Heat-api-cloudwatch模块是为了兼容AWS Cloudwatch；用于接收Amazon CloudFormation API请求，并将其翻译成openstack能识别的请求，然后通过RPC传送给heat-engine；

Heat-engine模块执行收到的API请求。

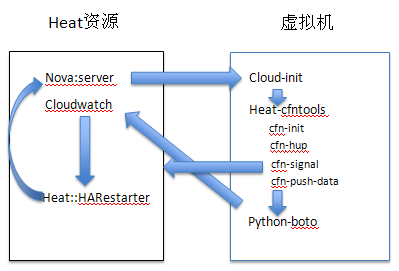
Heat工作流程：



# 组件说明

Heat的HA功能是OpenStack多模块配合实现的，涉及到Nova，neutron，Heat(需要启用Heat-cfn-api和heat-api-cloudwatch)，cloud-init，Heat-cfntools，python-boto等。

原理图简单介绍如下：



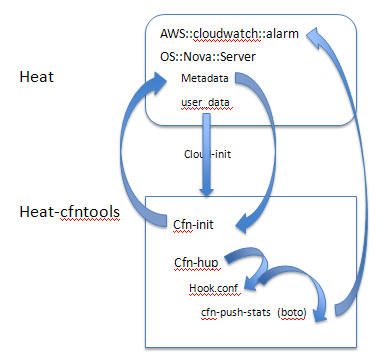
## Cloud-init

Cloud-init运行在虚拟机内部，需要打包到镜像中；用于在完成云实例创建后，做一些初始化配置。

## Heat-cfntools

Heat-cfntools包括了一些和Heat配合使用的小工具，它们运行在虚拟机内部，在制作虚拟机镜像的时候，需要将cfntools打包到镜像当中。

Heat的HA功能用Heat-cfntools中的cfn-init接收Heat注入的配置文件，并根据配置文件利用cfn-hup进行服务监控和触发异常情况的处理；cfn-push-stats对虚拟机或者服务的状态进行上报(调用python-boto)。



* cfn-init： 配置虚拟机，包括安装软件/服务，指定需要监控的服务，以及设置heat-cfntools的配置文件。
* cfn-hup：检测虚拟机内部的services的状态；如果检测到对应服务处于inactive状态，触发用户定义的hooks
* cfn-signal -> 发送执行命令成功或失败的信号
* cfn-push-stats -> 向heat-api-cloudwatch上报虚拟机服务状态

## Boto

Boto是提供AWS(Amazon Web Services)接口的Python包，运行在Python 2.6 and 2.7；Boto可以在openstack中使用；

# 模板分析

分析一下通过配置模板实现Heat的HA

"Resources" : {

"WebServerRestartPolicy" : { #执行删除后再创建虚拟机及其关联组件

"Type" : "OS::Heat::HARestarter",

"Properties" : {

"InstanceId" : { "Ref" : "WikiDatabase" }

}

},

"HttpFailureAlarm": { #接收虚拟机状态信息，到达门限时，执行OS::Heat::HARestarter

"Type": "AWS::CloudWatch::Alarm",

"Properties": {

"AlarmDescription": "Restart the WikiDatabase if sshd fails > 2 times in 1 minutes",

"MetricName": "ServiceFailure",

"Namespace": "system/linux",

"Statistic": "SampleCount",

"Period": "60",

"EvaluationPeriods": "1",

"Threshold": "2",

"AlarmActions": [ { "Ref": "WebServerRestartPolicy" } ],

"ComparisonOperator": "GreaterThanThreshold"

}

},

"WikiDatabase": { #虚拟机配置信息

"Type": "OS::Nova::Server",

"Metadata" : {

"AWS::CloudFormation::Init" : { #配置虚拟机，包括heat-cfntools和boto配置文件，以及安装软件及服务

"config" : { #配置文件

"files" : {

……

"/etc/cfn/notify-on-httpd-restarted" : { #发信息给cloudwatch

"content" : { "Fn::Join" : ["", [

"#!/bin/sh\n",

"/usr/bin/cfn-push-stats --watch ",

{ "Ref" : "HttpFailureAlarm" },

" --service-failure\n"

]]},

"mode" : "000700",

"owner" : "root",

"group" : "root"

},

#设置定时任务，执行cfn-hup，监控虚拟机服务状态；服务异常会重启服务，并触发hooks.conf

"/tmp/cfn-hup-crontab.txt" : {

"content" : { "Fn::Join" : ["", [

"MAIL=\"\"\n",

"\n",

"\* \* \* \* \* /usr/bin/cfn-hup -f\n"

]]},

"mode" : "000600",

"owner" : "root",

"group" : "root"

},

"/etc/cfn/hooks.conf" : { #

"content": { "Fn::Join" : ["", [

"[cfn-http-restarted]\n",

"triggers=service.restarted\n",

"path=Resources.WikiDatabase.Metadata\n",

"action=/etc/cfn/notify-on-httpd-restarted\n", #监控的服务发生异常时，触发相应处理

"runas=root\n"

]]},

"mode" : "000400",

"owner" : "root",

"group" : "root"

},

"/etc/boto.cfg" : { //python-boto配置文件

……

},

"mode" : "000400",

"owner" : "root",

"group" : "root"

}

},

"packages" : { #安装软件/服务

"yum" : {

"cronie" : []

}

},

"services" : { #设置heat-cfntools监控的服务

"systemd" : {

"sshd" : { "enabled" : "true", "ensureRunning" : "true" }

}

}

}

}

},

"Properties": {

……

"user\_data" : { "Fn::Base64" : { "Fn::Join" : ["", [ #在虚拟机创建完成后执行

"#!/bin/bash -v\n",

"/usr/bin/cfn-init -s ", { "Ref" : "AWS::StackName" }, #heat-cfntools进行初始化设置

" -r WikiDatabase ",

" --access-key ", { "Ref" : "WebServerKeys" },

" --secret-key ", {"Fn::GetAtt": ["WebServerKeys", "SecretAccessKey"]},

" --region ", { "Ref" : "AWS::Region" },

" || error\_exit 'Failed to run cfn-init'\n",

"# install cfn-hup crontab\n",

"crontab /tmp/cfn-hup-crontab.txt\n", #在虚拟机中设置定时任务，定期检测服务状态

]]}}

}

}

在虚拟机创建成功后，执行配置的UserData：UserData脚本执行的时候首先调用cfn-init，它会根据AWS::CloudFormation::Init段的配置，对heat-cfntools做一些初始化设置，例如生成相应的配置文件，安装软件、服务，执行脚本等等；然后设置一个定时任务(/tmp/cfn-hup-crontab.txt，是一个crontab的配置文件)，定期执行cfn-hup，实现监控某些服务。

如果发现监控的服务不是运行状态(active)，则会启动服务，并执行 /etc/cfn/hooks.conf 中的action，即调用cfn-push-stats向heat-api-cloudwatch（watch为HttpFailureAlarm）发送一个service-failure消息。

根据HttpFailureAlarm的配置，会触发AlarmActions OS::Heat::HARestarter；根据Heat对于资源OS::Heat::HARestarter的定义，Heat会删除原先的虚拟机WikiDatabase及其关联组件，然后重新创建一个虚拟机及其关联组件。

## 虚拟机初始化流程

## 虚拟机内部服务监控流程

## 虚拟机内部服务异常上报流程

## 虚拟机重生流程

# 参考文献