2017.10.8

OpenStack 基本功能介绍

1451139

朱睿安

目录

一、简介

什么是 OpenStack？OpenStack 是一个开源的云计算管理平台项目，由几个主要的组件组合起来完成具体工作。OpenStack 支持几乎所有类型的云环境，项目目标是提供实施简单、可大规模扩展、丰富、标准统一的云计算管理平台。OpenStack 通过各种互补的服务提供了基础设施即服务（IaaS）的解决方案，每个服务提供 API 以进行集成。

本篇文章介绍了 OpenStack 的简单功能，以及执行这些功能的操作，主要是根据《OpenStack Operations Guide》的第八、九、十章翻译整理而成。

二、Projects（keystone 组件）

在 OpenStack 用户界面和一些文档中，有时候你会看到“project”是指一组用户，而有时候你会也看到用来替代“tenant”，这两种术语是可以通用的。

这是因为最初 OpenStack 计算服务(nova)有着自己的身份验证系统，并使用的术语“project”。当认证系统独立成为 OpenStack 身份识别服务(Keystone)项目后，新项目中使用的术语“tenant”代指一个用户组。由于这一问题，一些 OpenStack 工具是指“project”，有些是指“tenant”。

一个用户必须至少属于一个项目，也可以属于多个项目。因此,至少添加一个项目后，才可以添加用户。

我们可以通过仪表盘来创建项目，具体过程不再赘述，或者通过命令行来创建项目。

```
# keystone tenant-create --name=demo
```

这将创建一个新项目命名为“demo”。可以用 –description <tenant-description> 参数添加一些描述。也可以用 –enable false 参数创建一个禁用状态的租户，不指定是默认开启状态。

三、Quotas

为了防止系统性能在没有通知的情况下消耗殆尽，我们需要 quota 来防止这一情况我们可以通过命令行来管理 OpenStack 计算服务(nova)、OpenStack 对象存储服务(swift)和 OpenStack 块存储服务(cinder)的 quota 即配额。

1、Image Quotas

镜像的配额是应用于全云端的，所以一旦将镜像的配额设置为例如 5GB，那么在你云中的所有项目都将只被允许设置 5GB 的镜像或者快照。

修改镜像配额需要在 /etc/glance/glance-api.conf 文件下的[DEFAULT]选项中增加

user_storage_quota = <bytes>

2、Compute Service Quotas（nova 组件）

通过命令行可以查看或修改计算服务默认的配额或是某个项目的配额。

查看所有租户的默认配额

$ nova quota-defaults

修改一个新租户的默认配额

$ nova quota-class-update default key value

将租户 ID 赋值

$ tenant=$(keystone tenant-list | awk '/tenantName/ {print $2}')

显示该租户的所有配额

$ nova quota-show --tenant $tenant

更新该租户的配额

 # nova quota-update --quotaName quotaValue tenantID

查看帮助

$ nova help quota-update

3、Object Storage Quotas（swift 组件）

对象存储配额有两类，Container quotas 和 Account quotas ，无论是需要修改哪一类的配额，都需要在 proxy-server.conf 文件中加上

[pipeline:main]

pipeline = healthcheck [...] container_quotas account_quotas proxy-server

[filter:account_quotas]

use = egg:swift#account_quotas

[filter:container_quotas]

use = egg:swift#container_quotas

查看一个项目的账户配额

$ swift stat

应用或修改一个项目的账户配额

$ swift post -m quota-bytes:

<bytes>

4、Block Storage Quotas（cinder 组件）

查看所有项目的默认配额

$ cinder quota-defaults

在/etc/ cinder/cinder.conf 文件中修改新租户的默认配额

查看一个租户的配额

 # cinder quota-show tenantName

将租户 ID 赋值

$ tenant=$(keystone tenant-list | awk '/tenantName/ {print $2}')

更新配额的值

 # cinder quota-update --quotaName NewValue tenantID

四、User Management

在命令行用户管理用户非常不方便。需要多条命令才能完成一个任务，并且要是用 UUID，而不是象征性的名字。在实践中，人们通常不会使用命令行管理用户。幸运的是，OpenStack 仪表盘提供了一个合理的接口。

1、Creating New Users

要创建一个新的用户，您将需要以下信息：

- 用户名
- 邮箱
- 密码
- 所属主要项目
- 角色

用户名和电子邮件都是不言而喻的，虽然我们的网站可能有本地习惯，但是这样便于观察。设置和更改密码的认证服务，需要管理员权限。

角色就是一个”会员”，可以直接使用：

- “member”：一个典型的用户。
- “admin”：超级管理员用户，在所有项目中，你应该谨慎使用它。

我们也可以定义其它角色，但很少这样做。

2、Associating Users with Projects

许多网站运行与用户相关的只有一个项目。这是一种较为保守和简单的管理用户选择。在管理上，一个用户报告出现很明显问题的一个实例或配额，如果它们在一个项目中，用户不必担心它们的行为是哪个项目。然而，需要注意在默认情况下，任何用户都可以影响到这个项目下其他用户资源的使用额度。也可以让用户关联多个项目，这样的组织比较有意义。在仪表盘中可以进行这一操作。

按照惯例，典型的应用是在一个单一的项目里，该项目创建默认设置云管理用户。如果您的管理用户使用云资源来启动和管理，强烈建议您使用单独的用户账户来管理访问权限和云正常运作，它们在不同的项目里。

五、Images（glance 组件）

OpenStack 镜像通常可以被理解为 “virtual machine templates（虚机模板）” 。镜像也可以被认为是标准安装介质例如 ISO 镜像. 基本上，它们都含有能启动实例的启动系统文件。

glance image-create 命令有很多选项

min-disk 选项对启动分区有大小要求的镜像(象 windows 需要比较大的分区)非常有用。

location 选项需要特别注意。它并不复制整个镜像到 Glance，而是提供镜像的原始路径。当启动一个实例的时候，Glance 会到该路径加载镜像。

copy-from 选项从指定路径复制镜像到 /var/lib/glance/images 。

查看这些选项

$ glance help image-create

运行下述命令来查看已有镜像的详细信息

$ glance details

删除镜像

$ glance image-delete <image uuid>

注意：删除镜像不影响基于此镜像的虚机实例或快照。

六、Flavors（nova 组件）

在 Openstack 中，Virtual hardware templates（虚机硬件模板）被称为类型模板(flavor)，包括 RAM 和硬盘大小，CPU 核数等。标准安装后有 5 个缺省的类型。类型模板可以被有管理员权限的用户修改(修改的权限也可以被编辑，通过在 nova-api 服务器上的/etc/nova/policy.json 文件中修改访问控制：

compute_extension:flavormanage)。Flavor 定义了很多元素，可以在官网的表中查看。

用户往往会为了项目创建一个新的私人类型模板，通过命令行可以连接模板

$ nova flavor-access-add <flavor-id> <project-id>

那如何修改一个已有的类型模板？不幸的是，OpenStack 没有提供修改模板的接口，只有增加和删除。Dashboard 里的模板修改的工作模式其实是删除旧模板并增加一个同名模板。

七、Security Groups（nova 组件）

对新用户，Openstack 最常见的问题是当启动一个实例，未能设置适当的安全组，之后后无法访问网络上的实例。 安全组是一组应用于一个实例的网络的 IP 过滤规则，是基于具体项目的，项目成员可以编辑默认的规则，或对他们组添加新规则。所有的项目如果没有其他安全组定义，都有一个“默认”安全组，并将其应用于实例。

nova.conf 文件中的选项 allow_same_net_traffic (缺省为 true) 是在全局范围内，控制规则是否适用于共享一个网络的主机群。当设置为 true，在同一子网主机之间可以相互传送所有类型数据，没有经过过滤。在 Flat 模式网络，这使得所有项目中的所有实例可以未过滤的相互通信。在 VLAN 模式网络,只有在同一个项目允许实例相互访问。增加新规则

$ nova secgroup-add-rule <secgroup> <ip-proto> <from-port> <to-port><cidr>

这里的 ‘from-port’ 和 ‘to-port’ 并非源端口和目的端口，而是端口的范围。复杂的规则可以通过多条规则实现。反方向操作是 secgroup-delete-rule，同样的格式。如果想删除整个安全组可以用：secgroup-delete.

一个用户设置一个 SourceGroup (有安全组名字)， 所有此用户在其他的实例上可以动态选择使用这个 SourceGroup。 这种方式减轻了每个新用户需要一个新的规则的麻烦。

$ nova secgroup-add-group-rule <secgroup> <source-group> <ip-proto> <from-port> <to-port>

八、Block Storage（cinder 组件）

OpenStack 卷是持久的块存储设备，可以附加到实例或分离，但同时只能连接到一个实例，类似于一个外部硬盘，不象网络文件系统或对象存储那样提供共享存储方式。块存储让实例中的操作系统把文件系统加载和安装到块设备。类似于其他可移动磁盘技术，操作系统不能利用磁盘，然后马上移掉它，数据容易出问题。

OpenStack 没有涉及实例操作系统在访问块设备时需要的步骤等等相关规则。 所涉及到的只有如何创建卷并将其挂载到实例或卸载。这些操作可以在 dashboard 的 ‘卷（volume）’ 页面中找到，或用 cinder 命令行。为了增加一个卷只需要名字和卷大小（GB），将其输入到 ‘创建卷’ 菜单，或者用命令行方式，创建了一个人名为 ‘test-volume’ 、10GB 的卷

$ cinder create –display-name test-volume 10

列出已存在的卷和所连接的实例

$ cinder list

块存储服务还允许创建快照。值得注意的是这是块级别的快照，最好在卷没有连接到实例的时候进行快照，或者卷没有被加载或使用的时候做。 如果在卷被频繁使用的时候做，可能得到的是一个不一致的文件系统。

九、Instances(nova 组件)

实例是在一个 OpenStack 云运行的虚拟机。为了发起一个实例，需要选择一个镜像，一个类型模型，和一个名字。 名字不一定需要唯一，但如果是唯一的话，你的日子会好过很多，因为很多工具用到名字代替 UUID。

发起实例可以通过 dashboard 的 “实例” 页面的 “发起实例(launch instance)” 按钮，然后到选择镜像和快照页面。 也可以选择命令行模式

$ nova boot –flavor <flavor> –image <image> <name>

删除实例

$ nova delete <instance-uuid>

要注意， 将一个实例关闭电源(关机)并不代表这个实例在 openstack 中被去除。

查看实例测试数据

$ nova show test-instance

有各种各样的方法来注入定制化的数据包括用户数据，元数据服务，授权密钥(authorized_keys)注入，和文件注入。 澄清用户数据与元数据的区别：“用户数据”是一部分数据，在实例没有运行时设置。这个用户数据可以在实例运行中存取和使用。人们使用这个用户数据来存储配置，脚本,或其他的数据，如果租户想要的话。 而元数据，是与实例相关联的一组 键/值（key/value）。在实例存在期间，当用户通过 Compute API 发出指令时，nova-compute 从实例的内部或外部来读写这些键/值。

用户可以通过 nova 命令生成和注册 ssh 密钥：

$ nova keypair-add mykey > mykey.pem

这生成了一个名字为 mykey 的密钥，可以关联到实例上。mykey.pem 文件是私钥，需要保存到一个安全的地方，因为它允许以 root 的用户访问与其关联的实例。我们可以用以下命令注册一个公钥：

$ nova keypair-add –pub-key mykey.pub mykey

同时我们必须要有相对应的私钥来访问和此公钥相关联的实例，在一个实例启动时关联密钥

$ nova boot –image ubuntu-cloudimage –flavor 1 -

key_name mykey

十、实例启动图