

从本节开始我们学习 OpenStack 的 Block Storage Service，Cinder

**理解 Block Storage**

操作系统获得存储空间的方式一般有两种：

1. 通过某种协议（SAS,SCSI,SAN,iSCSI 等）挂接裸硬盘，然后分区、格式化、创建文件系统；或者直接使用裸硬盘存储数据（数据库）
2. 通过 NFS、CIFS 等 协议，mount 远程的文件系统

第一种裸硬盘的方式叫做 Block Storage（块存储），每个裸硬盘通常也称作 Volume（卷） 第二种叫做文件系统存储。NAS 和 NFS 服务器，以及各种分布式文件系统提供的都是这种存储。

**理解 Block Storage Service**

Block Storage Servicet 提供对 volume 从创建到删除整个生命周期的管理。

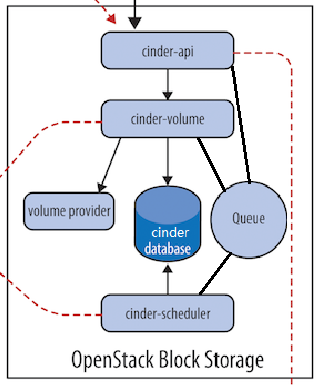
从 instance 的角度看，挂载的每一个 Volume 都是一块硬盘。

OpenStack 提供 Block Storage Service 的是 Cinder，其具体功能是：

1. 提供 REST API 使用户能够查询和管理 volume、volume snapshot 以及 volume type
2. 提供 scheduler 调度 volume 创建请求，合理优化存储资源的分配
3. 通过 driver 架构支持多种 back-end（后端）存储方式，包括 LVM，NFS，Ceph 和其他诸如 EMC、IBM 等商业存储产品和方案

**Cinder 架构**

下图是 cinder 的逻辑架构图



Cinder 包含如下几个组件：

**cinder-api**  
接收 API 请求，调用 cinder-volume 执行操作。

**cinder-volume**  
管理 volume 的服务，与 volume provider 协调工作，管理 volume 的生命周期。运行 cinder-volume 服务的节点被称作为存储节点。

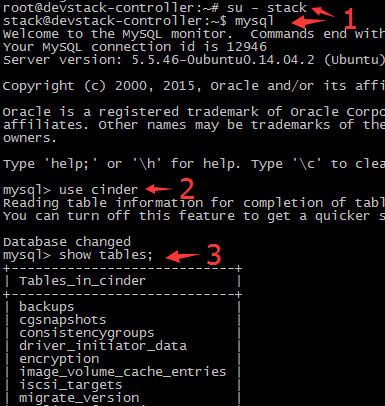
**cinder-scheduler**  
scheduler 通过调度算法选择最合适的存储节点创建 volume。

**volume provider**  
数据的存储设备，为 volume 提供物理存储空间。

cinder-volume 支持多种 volume provider，每种 volume provider 通过自己的 driver 与cinder-volume 协调工作。

**Message Queue**  
Cinder 各个子服务通过消息队列实现进程间通信和相互协作。因为有了消息队列，子服务之间实现了解耦，这种松散的结构也是分布式系统的重要特征。

**Database** Cinder 有一些数据需要存放到数据库中，一般使用 MySQL。数据库是安装在控制节点上的，比如在我们的实验环境中，可以访问名称为“cinder”的数据库。



**物理部署方案**

Cinder 的服务会部署在两类节点上，控制节点和存储节点。

我们来看看控制节点 devstack-controller 上都运行了哪些 cinder-\* 子服务。

http://7xo6kd.com1.z0.glb.clouddn.com/upload-ueditor-image-20160610-1465514630477045453.jpg

cinder-api 和 cinder-scheduler 部署在控制节点上，这个很合理。

至于 cinder-volume 也在控制节点上可能有些同学就会迷糊了：cinder-volume 不是应该部署在存储节点上吗？

要回答这个问题，首先要搞清楚一个事实：

OpenStack 是分布式系统，其每个子服务都可以部署在任何地方，只要网络能够连通。

无论是哪个节点，只要上面运行了 cinder-volume，它就是一个存储节点，当然，该节点上也可以运行其他 OpenStack服务。

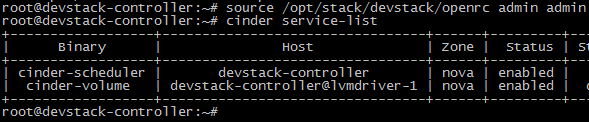
cinder-volume 是一顶存储节点帽子，cinder-api 是一顶控制节点帽子。在我们的环境中，devstack-controller 同时戴上了这两顶帽子，所以它既是控制节点，又是存储节点。当然，我们也可以用一个专门的节点来运行 cinder-volume。

这再一次展示了 OpenStack 分布式架构部署上的灵活性：

可以将所有服务都放在一台物理机上，用作一个 All-in-One 的测试环境；而在生产环境中可以将服务部署在多台物理机上，获得更好的性能和高可用。

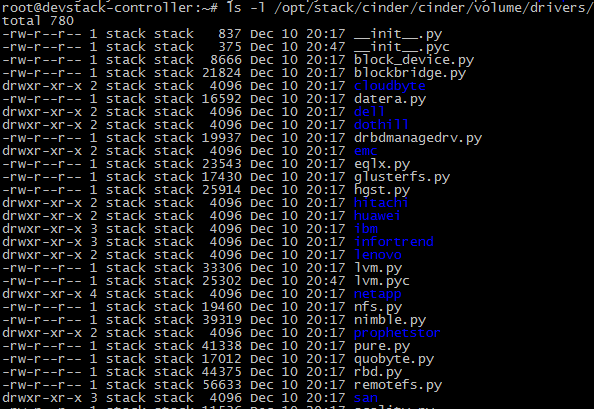
RabbitMQ 和 MySQL 通常是放在控制节点上的。

另外，也可以用 cinder service list 查看 cinder-\* 子服务都分布在哪些节点上



还有一个问题：volume provider 放在那里？

一般来讲，volume provider 是独立的。cinder-volume 使用 driver 与 volume provider 通信并协调工作。所以只需要将 driver 与 cinder-volume 放到一起就可以了。在 cinder-volume 的源代码目录下有很多 driver，支持不同的 volume provider。



后面我们会以 LVM 和 NFS 这两种 volume provider 为例讨论 cinder-volume 的使用，其他 volume provider 可以查看 OpenStack 的 configuration 文档。

下一节我们将讨论 Cinder 的这些组件如何协调工作。