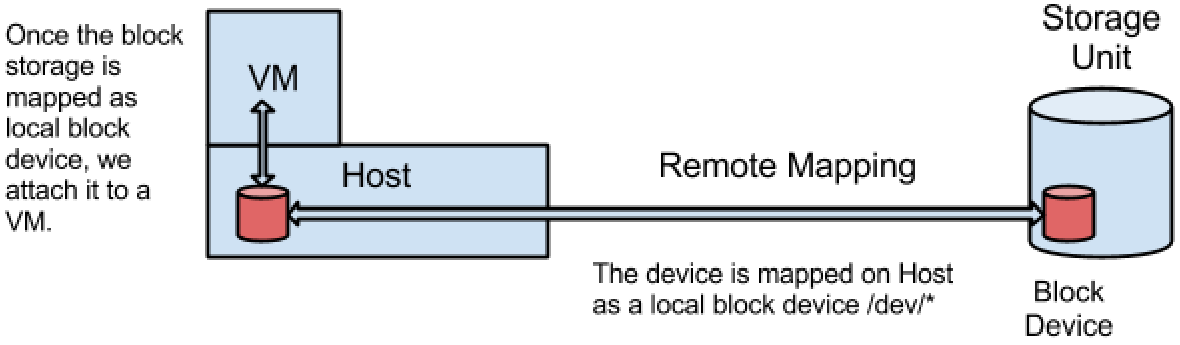
Block Storage for Cloud Orchestration Framework

Use Case

We have a small cloud facility running by now. We can boot and destroy VMs with simple API calls. Its all really good with just a minor flaw. The users still don’t have a way to store data permanently. As soon as the VM/instance is destroyed, the data is gone. The next challenge is to add a storage unit to our orchestration framework. The users should be able to create block storage devices and attach them to their respective instances on the fly.

Architecture


Once the block storage is mapped as local block device, we attach it to a VM.
VM
Host
Remote Mapping
Storage Unit
The device is mapped on Host as a local block device /dev/*
Block Device

Storage Unit

Storage Unit is going to be a separate server running an iSCSI[1] service. These services are capable of exporting block storage devices over network using iSCSI standard. There are plenty of solutions out there.

- Linux-ISCSI[2]
- TGT[3]
- Ceph[4]

We’ll go with Ceph. Its a robust storage solution which provides multiple interfaces to access data - block storage(RBD), network file system(similar to samba/NFS) and object storage. Its easy to setup and operate(remotely).

Note that, a ceph storage server will be provided to you during the evaluation. Hence, you needn’t setup one using your *install/setup* script. Although, you’ll need one on your local machine/VM for development.

Host

A host is going to run the VMs. This is where we'll map the iSCSI disks. In case of ceph, the ceph client(rbd command line tool) needs to be configured on the host. Steps involved are:

1. Use rbd to create a block device.
2. Use rbd to map it to your local system. The device will now appear as a local block storage device.
3. Attach it to your VM using libvirt.

I request you to go through the documents and try installing Ceph RBD[5] on your system or on a small VM before coming to tutorial.

References

1. https://en.wikipedia.org/wiki/ISCSI
2. http://linux-iscsi.org/wiki/Main_Page
3. http://stgt.sourceforge.net/
4. http://ceph.com/docs/next/rbd/rbd/
5. http://ceph.com/docs/next/start/quick-rbd/