RHOSP6 integration with Nimble Array. - Corporate IT - Confluence

Added by Paul Clifton

RHOSP6 integration with Nimble Array.

Overview
The integration of RHOSP 6 with a Nimble storage array (CS700 model) is no different from using any other cinder backend.

Networking
To setup cinder to access Nimble, first we had to add the iscsi vlan to the TOR switch that both the controller, and compute nodes were attached to. This is because the cinder volume service only passes the final iscsi target to the compute node for attachment to the VM, and does not actually handle any of the data flow.

Next we had to add an interface for the newly added iscsi vlan on to the bond we are using for Storage and tenant networks. In this case we were using the 1611 vlan. This has to be done on both controllers and compute nodes.

```
cat /etc/sysconfig/network-scripts/ifcfg-bond0.1611
BOOTPROTO=none
 IPADDR=172.16.11.160
 NETMASK=255.255.255.0
 DEVICE=bond0.1611
 ONBOOT=yes
 PEERDNS=no
 PEERROUTES=no
 VLAN=yes
 NM_CONTROLLED=no
 DEFROUTE=no
```

Cinder Setting
The block volume storage service in Openstack cinder, is designed to be modular and allow for multiple backends to be designated. The following changes to the the `/etc/cinder/cinder.conf` file allows for the addition of the nimble array, but still keeps the ceph/rbd volume type as the default for volume creation.

```
[default]
 default_volume_type=rbd
 enabled_backends=nfs,rbd,nimble
[nimble]
 san_ip=admin_interface_on_array
 san_login=admin
 san_password=admin_password
 volume_driver=cinder.volume.drivers.nimble.NimbleISCSIDriver
 volume_backend_name=nimble
 nimble_ito_enabled=true
```

There are some known bugs in the version of nimble.py the drivers for the Nimble that cinder uses. Under the direction of Nimble technical staff, nimble drivers 1.1 were downloaded and copied over to the cinder-volume server.

```
# mv nimble.py /usr/lib/python2.7/site-packages/cinder/volume/drivers/nimble.py
```

> This seems to be CLI bug caused by a common library change in Juno. To work around the issue, please specify the option "--display-description" during "cinder create" CLI.
>
> The bug has been fixed in upstream. Please see https://bugs.launchpad.net/cinder/+bug/1414247 for details.
>
> To fix the issue,
>
> Option #1: download the source from upstream, for example, https://github.com/openstack/cinder/blob/master/cinder/volume/drivers/nimble.py Replace the driver and restart cinder services.
>
> Option #2: Replace the driver with the attached nimble.py; restart cinder services.

Once the changes to cinder.conf have been made the cinder services need to be restarted. This will enable the changes that we made to cinder.conf as well as load in the new python driver library.

```
# openstack-service restart cinder
Once restarted we can see the volume service up and running.
[root@controller1 ~(keystone_admin)]# cinder service-list
 +------------------+------------------------------------------------+------+---------+-------+------------------------+------
 | Binary | Host | Zone | Status | State | Updated_at | Disabled Reason |
 +------------------+------------------------------------------------+------+---------+-------+------------------------+------
 | cinder-scheduler | controller1.juno-openstack.servicemesh.com | nova | enabled | up | 2015-05-08T13:04:12.000000 | None |
 | cinder-volume | controller1.juno-openstack.servicemesh.com@nfs | nova | enabled | up | 2015-05-08T13:04:16.000000 | None |
 | cinder-volume | controller1.juno-openstack.servicemesh.com@nimble | nova | enabled | up | 2015-04-27T16:26:31.000000 | None |
 | cinder-volume | controller1.juno-openstack.servicemesh.com@rbd | nova | enabled | up | 2015-05-08T13:04:18.000000 | None |
 +------------------+------------------------------------------------+------+---------+-------+------------------------+------
```

We need to setup service types. Adding key types allows us to designate what "type" uses which backend for cinder.

```
# cinder type-create nimble
```

```
# cinder type-key nimble set volume_backend_name=nimble

[root@controller1 ~(keystone_admin)]# cinder type-list
+------------------------------------+--------+
| ID | Name |
+------------------------------------+--------+
| 0a117370-4864-4387-803b-ddb1057c12b5 | nfs |
| 20306830-4739-4282-8e09-1a08fce24de1 | rbd |
| 2e8b559d-bf7e-449f-a90a-2f31edf55e6f | nimble |
+------------------------------------+--------+
```

Results

After sorting out a few minor issues in terms of setup, deployment of volumes from cinder was effortless. Using the configuration option:

```
nimble_ito_enabled=true
```

Launching multiple copies of the same instance were very quick as this option enables the cloning of the base image used in the volume, if it is already on the array, cutting down the need to copy the image from glance multiple times.

During our testing of this array we noticed that there was a bottleneck in the environment that limited IO to 120MB/s; this is due to the fact that the iSCSI connectivity goes from the Juno Lab ==> TOR ==>Nexus 9K ==> Cisco4506 (limiting factor) ==> Arista ==> Nimble array.   The same bottleneck was not seen when using RBD (Rados Block Device aka ceph) and the backed (however the storage for RBD is in the same rack as the Openstack environment).