

# OpenStack: How can OpenStack be deployed on a Bright 7.0 HA Cluster?

## Notes on deploying OpenStack on HA Clusters

We start with a single headnode cluster:

- deploy openstack
- add second headnode

## Requirements:

Bright Cluster Manager 7.0 (OpenStack Icehouse)

One headnode is configured

Second headnode is not configured yet

At least two compute nodes (in this example those are named “networknode” and “node001”)

Both compute nodes are down (or CMDaemon is stopped on them)

Shared storage for the headnodes /cm/shared is not configured yet

## Introduction

These instructions show how to configure a Bright 7.0 HA deployment composed of two headnodes (which host some OpenStack roles and no Ceph roles), and two compute nodes. In this example both compute nodes participate both in the OpenStack deployment as well as in the Ceph deployment. (a openStack network + Ceph Monitor, and a OpenStack compute host + Ceph OSD). In practice, in production deployments it's of course possible and advised to have separate nodes dedicated for OpenStack, and separate nodes dedicated for Ceph.

The end result of these instructions will be using Ceph for all the storage needs of OpenStack.

## The instructions

**# make sure the slave nodes which will participate in the OpenStack and/or Ceph deployment are powered off.**

**# run cmha-setup -> setup -> configure**

(this will create a second headnode object inside bright's configuration)

**# Prepare for running cm-ceph-setup.** Ceph deployment process at least one of the nodes which will be later specified as the ceph monitor nodes to be UP during the deployment. This node must have 'ceph' rpm preinstalled:

- Configure the Software image used by the slave nodes which will be later specified as the ceph monitor nodes. Just do “yum install ceph” for the software image of those nodes. There's no need to install this rpm to software images used by OSDs (cm-ceph-setup will do that).
- power on all (or at least one) of the slave nodes which will be later specified as the Ceph monitor nodes, and let them be provisioned with the software image mentioned in the previous step. In the case of the minimum example Ceph deployment that would be the “networknode”. Not all monitors have to be up, so even if you will use multiple monitor nodes later on, powering one at this stage is enough.

# OpenStack: How can OpenStack be deployed on a Bright 7.0 HA Cluster?

## # run cm-ceph-setup

This will configure ceph. The Headnode should NOT be used either as the Ceph OSD or as the Ceph monitor node. (However, the headnode can be used as the monitor node if needed (but since there must be a odd number of monitor nodes, this means that both headnodes would have to be monitor nodes + an odd number of additional monitor nodes).

A minimum functioning Ceph deployment is composed of two nodes. Since in the case of HA clusters the Ceph nodes cannot be the headnodes, the minimal working ceph-based bright HA deployment is composed of two headnodes, and two slave nodes. One slave node being the Ceph monitor, the other being the Ceph OSD. Note, that with such a minimal deployment, data stored on the OSD is not replicated across other OSDs, so this deployment is only recommended for tests. Note that at least one future monitor node must be up when running cm-ceph-setup and have ceph packages must installed (as described in the previous step).

In the case of the minimum example deployment discussed in this document, the user should configure “networknode” as the monitor node, and “node001” as the OSD node. and power on the remaining ceph nodes. After running cm-ceph-setup power on the nodes which participate in the Ceph cluster. This will effectively enable Ceph. This is needed because a functioning Ceph cluster is required for the Ceph-enabled OpenStack deployment process to work. (Ceph is used by cm-openstack-setup).

In the case of the minimum example deployment discussed in this document, the user would have to power on node001 (node which had the OSD role).

At this point, “ceph -s” should report the monitor and OSD nodes to be up

## # Deploy OpenStack.

In the case of the minimum example deployment discussed in this document, node001 (which is now already Ceph OSD) should be picked as the Nova compute host, and the “networknode” (which is not already the Ceph monitor) as the OpenStack networking node.

## # Reboot all OpenStack slave nodes

The OpenStack deployment process will offer to automatically reboot all slave nodes participating in the OpenStack deployment. This has to be done for the interfaces on those nodes to be reconfigured by the node-installer.

In the case of the example deployment, both network node and node001 must be rebooted and SYNC-reinstalled.

It's important to NOT do a FULL install on the OpenStack nodes which are also Ceph nodes at this point. It's ok ,at this stage, to do a FULL install for OpenStack nodes which are NOT Ceph nodes.

Wait for the nodes to come back up.

Headnodes always, by default, participate in the OpenStack deployment. However they don't

# OpenStack: How can OpenStack be deployed on a Bright 7.0 HA Cluster?

have to be rebooted at this stage.

**# At this stage both OpenStack and Ceph deployments are functional.** However HA is still missing.

**# Verify that the following file exists on the headnode** [it should have been written out during cm-openstack-setup]:

```
/cm/local/apps/cluster-tools/ha/conf/extradbclone_openstack.xml
```

This file contains the information required by the cmha-setup script to configure database cloning for OpenStack databases.

**# disable OpenStack:**

```
[openstack1->openstack[default]]% set enabled 0; commit
```

This will prevent CMDaemon from reconfiguring openstack while the HA setup process takes place.

**# stop all OpenStack services via cmsh or cmgui**

Both on the headnode, and on all the compute nodes

**# cmha-setup -> setup -> clone install**

this will clone the headnode to the secondary headnode

(Including the openstack packages which were just installed)

After the clone is done, when asked by the rescue environemnt, reboot the node

wait for the node to power on (it will be DOWN in Bright, as CMDaemon is not configured yet, see next step)

**# cmha-setup -> setup -> finalize**

this will clone the databases to the failover node, and start the cmdaemon. Note that the OpenStack services must be down, otherwise they may be writing data to the database at this moment.

Once asked, agree for the reboot of the failover headnode.

**# Create rabbitmq user on the second headnode and assign permissions to it**

```
ssh master2
```

```
rabbitmqctl add_user <username> <password>
```

To get username and password, use CMSH:

```
[openstack1]% openstack use default
```

```
[openstack1->openstack[default]]% credentials
```

```
[openstack1->openstack[default]->credentials]% get messagequeueusername
```

```
openstack
```

```
[openstack1->openstack[default]->credentials]% get messagequeuepassword
```

# OpenStack: How can OpenStack be deployed on a Bright 7.0 HA Cluster?

7AUrVpdy8DWYIqPKGooJDyda

Now set the permissions (again, on the secondary headnode)

```
rabbitmqctl set_permissions -p / <username> '.*' '.*' '.*'
```

## # cmha-setup -> shared Storage

This will configure shared storage for the nodes. in the case of NAS, this will copy local /cm/shared to an external share. Note, that at this point /cm/shared might already contain some OpenStack-related file. That's why it's important to deploy OpenStack before cloning /cm/shared to the external storage

## # Enable OpenStack

```
[openstack1->openstack[default]]% set enabled 1; commit
```

## # start openstack services using cmsh or cmgui

## # Done!

Unique solution ID: #1271

Author: Piotr Wachowicz

Last update: 2017-03-17 18:20