

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

This article explains how to deploy the Manila file sharing service in a Bright Openstack cluster. In this deployment CephFS will be used as the storage backend.

Prerequisites:

- Ceph and Bright Openstack already deployed
- A CephFS filesystem is already deployed. Step 1 of this article has a link to an article on how to deploy CephFS on Bright.
- The instances must have access to Ceph's public network in order to mount the shares.

Manila requires two types of nodes: the controller node (where the openstack-manila-api, and the openstack-manila-scheduler services run) and a share node (where the openstack-manila-share service runs) In this example one of the Openstack Controller nodes will be chosen to fulfill both roles. In the following procedure we will refer to the controller node in which Manila will be installed as the Manila Node.

This procedure was tested in Bright Cluster Manager 8.0 and Openstack Newton.

1 - Deploy CephFS (optional)

To deploy CephFS (if not already deployed), please refer to the KB article in <http://kb.brightcomputing.com/faq/index.php?action=artikel&cat=24&id=411>

2 - Configure a Ceph user for Manila

Run the following commands in the head node:

```
# MON_CAPS="allow r, allow command \"auth del\", allow command \"auth caps\", allow command \"auth get\", allow command \"auth get-or-create\""
```

```
# ceph auth get-or-create client.manila -o /etc/ceph/manila.keyring mds 'allow *' osd 'allow rw' mon "$MON_CAPS"
```

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

The above command will place the keyring file for the client.manila identity in `/etc/ceph/manila.keyring`.

Now it's necessary to modify the Ceph configuration to include specific configuration for client.manila.

Run the following commands in the head node:

```
# cmsg  
  
% ceph  
  
% append extraconfigparameters "[client.manila]client mount uid=0"  
  
% append extraconfigparameters "[client.manila]client mount gid=0"  
  
% append extraconfigparameters "[client.manila]log file=/var/log/manila/ceph-client.manila.log"  
  
% append extraconfigparameters "[client.manila]admin  
socket=/var/run/manila/ceph-$name.$pid.asok"  
  
% append extraconfigparameters "[client.manila]keyring=/etc/ceph/manila.keyring"  
  
% commit
```

After a minute, verify that the new values were written to `/etc/ceph/ceph.conf` in the head node.

Copy the `/etc/ceph/ceph.conf` and `/etc/ceph/manila.keyring` files from the head node to the Manila Node and also to its corresponding software image.

3 - Install Manila packages in the Manila node.

Run the following command in the Manila Node:

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

```
# yum install openstack-manila openstack-manila-share
```

Install the packages also in the software image. For example if the Manila Node uses the default-image software image, run the following command in the head node:

```
# yum install openstack-manila openstack-manila-share --installroot=/cm/images/default-image
```

4 - Configure the manila database in Galera.

In one of the controller nodes log into the local MariaDB database as root.

```
# mysql -u root -p
```

Create the database:

```
> CREATE DATABASE manila;
```

Assign privileges on the database to the manila user (substitute MANILA_DBPASS for a different password):

```
>GRANT ALL PRIVILEGES ON manila.* TO 'manila'@'localhost' \
IDENTIFIED BY 'MANILA_DBPASS';
>GRANT ALL PRIVILEGES ON manila.* TO 'manila'@'%'\
IDENTIFIED BY 'MANILA_DBPASS';
```

5 - Configure the Manila services and endpoints in Openstack

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

Run the following commands in the head node:

```
# cmsh

% openstack

% users

% add manila; set domain default; set password

% commit

% roleassignments

% add manila:service:admin; set user manila; set project service; set role admin

% commit

% services

% add manila; set description "Openstack Shared File Systems"; set type share

% add manilav2; set description "Openstack Shared File Systems V2"; set type sharev2

% exit; commit

% endpoints

% add manila:publicv1; set region defaultRegion; set service manila; set interface public; set url
http://<MANILA_IP>:8786/v1/%\(\tenant_id\)s

% add manila:internalv1; set region defaultRegion; set service manila; set interface internal; set
url http://<MANILA_IP>:8786/v1/%\(\tenant_id\)s

% add manila:adminv1; set region defaultRegion; set service manila; set interface admin; set url
http://<MANILA_IP>:8786/v1/%\(\tenant_id\)s

% add manila:publicv2; set region defaultRegion; set service manilav2; set interface public; set
url http://<MANILA_IP>:8786/v2/%\(\tenant_id\)s

% add manila:internalv2; set region defaultRegion; set service manilav2; set interface internal;
set url http://<MANILA_IP>:8786/v2/%\(\tenant_id\)s

% add manila:adminv2; set region defaultRegion; set service manilav2; set interface admin; set
url http://<MANILA\_IP>:8786/v2/%\\(\tenant\_id\\)s
```

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

```
% exit; commit
```

In the above commands, replace <MANILA_IP> for the IP of the Manila Node.

6 - Configure /etc/manila/manila.conf in the Manila node

The following changes have to be made to the /etc/manila/manila.conf file in the Manila node and also in the software image used by it.

6.1 - Database

In the [database] section, set the connection property as follows (replace <PASSWORD> for the password assigned in step 4:

```
connection = mysql+pymysql://manila:<PASSWORD>@oshaproxy:3308/manila
```

6.2 - RabbitMQ

In the [DEFAULT] section set the transport_url property with the same credentials and servers the rest of the Openstack services are using for RabbitMQ. For example, use the same transport_url value which is defined in /etc/cinder/cinder.conf for the controller nodes.

6.3 - Other properties

In the [DEFAULT] section, set these properties:

```
share_name_template = share-%s  
rootwrap_config = /etc/manila/rootwrap.conf  
api_paste_config = /etc/manila/api-paste.ini
```

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

6.4 - Authentication against Identity service (Keystone)

In the [DEFAULT] section, set this property:

```
auth_strategy = keystone
```

In the [keystone_authtoken] section, set these properties (substitute MANILA_PASS for the password defined for manila in step 5):

```
memcached_servers = 127.0.0.1:11211
```

```
auth_uri=http://localhost:5000/v3
```

```
auth_url=http://localhost:5000/v3
```

```
auth_type = v3password
```

```
project_domain_name = Default
```

```
user_domain_name = Default
```

```
project_name = service
```

```
username = manila
```

```
password = MANILA_PASS
```

6.5 - Configure lock path

In the [oslo_concurrency] section, set this property:

```
lock_path = /var/lib/manila/tmp
```

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

6.6 - Enable share protocols

In the [DEFAULT] section, set this property:

```
enabled_share_protocols = CEPHFS
```

6.7 - Add backend configuration

Add the following section at the end of the file:

```
[cephfs1]
```

```
driver_handles_share_servers = False
```

```
share_backend_name = cephfs1
```

```
share_driver = manila.share.drivers.cephfs.cephfs_native.CephFSNativeDriver
```

```
cephfs_conf_path = /etc/ceph/ceph.conf
```

```
cephfs_auth_id = manila
```

```
cephfs_cluster_name = ceph
```

```
cephfs_enable_snapshots = False
```

6.8 - Enable share backends

In the [DEFAULT] section, set this property:

```
enabled_share_backends = cephfs1
```

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

In the above line, cephfs1 is the name of the backend which was defined in step 6.7.

7 - Initialize manila database

Run the following command in the Manila node to initialize the manila database:

```
# manila-manage db sync
```

8 - Configure the Manila services

Run the following command in the head node (substitute <manila-node> for the name of the Manila node:

```
# cmsh
% device use <manila-node>
% services
% add openstack-manila-api
% set autostart yes; set monitored yes
% clone openstack-manila-api openstack-manila-scheduler
% clone openstack-manila-api openstack-manila-share
% exit; commit
```

9 - Verify operation

Run the following command in the head node:

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

```
# manila service-list
```

If the manila-scheduler and manila-share services are listed as up, then the services are running normally.

10 - Create share type

Run the following commands in the head node:

```
# manila type-create cephfstype false
```

```
# manila type-key cephfstype set share_backend_name='cephfs1'
```

11 - Set a default share type (optional)

To configure the cephfstype share type defined in step 9 as the default (it will be used when no share type is specified), add the following property to the [DEFAULT] of the /etc/manila/manila.conf file in the Manila node.

```
default_share_type = cephfstype
```

Then restart the manila services in the Manila node.

12 - Create and mount a share (optional)

The following is an example on how a share can be configured and mounted.

To create the share, run the following command in the head node:

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

```
# manila create --share-type cephfstype --name testshare1 cephfs 1
```

Run the following command in the head node to get the export location of the share (this will be needed to mount the filesystem later):

```
# manila share-export-location-list testshare1 --columns path
```

The export location returned by the previous command will look like this

example:

```
10.141.0.4:6789,10.141.0.5:6789:/volumes/_nogroup/38ae58bd-0f9b-43ef-9c9c-c79c0b4bbf5d
```

Configure access to testshare1 for user testuser by running the following command on the head node:

```
# manila access-allow testshare1 cephx testuser
```

The command above will create a client.testuser user in Ceph, with the necessary permissions to access the share testshare1.

Generate a keyring file for testuser by running the following command on the head node:

```
# ceph auth get client.testuser -o ceph.client.testuser.keyring
```

In the instance in which the share is to be mounted, create the /etc/ceph/ceph.conf file with the following contents (in the second line specify the hostnames of the Ceph monitors):

OpenStack: How do I configure Manila in Bright OpenStack 8.0/8.1?

[client]

```
client quota = true
```

```
mon host = <monitor1>:6789,<monitor2>:6789
```

Copy into `/etc/ceph/ceph.client.testuser.keyring` the keyring file generated previously.

Run the following command on the instance in order to mount the share (make sure that the `ceph-fuse` command is installed)

```
# ceph-fuse /mnt --id=testuser --client-mountpoint=<mountpoint>
```

In the above command, replace `<mountpoint>` for the path included in the share's export location (without the addresses of the monitors). For example, if the share location is

`10.141.0.4:6789,10.141.0.5:6789:/volumes/_nogroup/38ae58bd-0f9b-43ef-9c9c-c79c0b4bbf5d` as in the previous example, then the mount command has to be like this:

```
# ceph-fuse /mnt --id=testuser  
--client-mountpoint=/volumes/_nogroup/38ae58bd-0f9b-43ef-9c9c-c79c0b4bbf5d
```

Unique solution ID: #1413

Author: Carlos Pintado

Last update: 2018-02-28 18:13