



Red Hat OpenStack Platform 16.0

NetApp Back End Guide for the Shared File System Service

Deploying Multiple NetApp Back Ends for the Shared File System Service in a Red Hat OpenStack Platform Overcloud

Red Hat OpenStack Platform 16.0 NetApp Back End Guide for the Shared File System Service

Deploying Multiple NetApp Back Ends for the Shared File System Service in a Red Hat OpenStack Platform Overcloud

OpenStack Team
rhos-docs@redhat.com

Legal Notice

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This document contains information about configuring and deploying the OpenStack Shared File System Service using a NetApp storage controller (running Data ONTAP) as a back end. This scenario uses the ``manila.share.drivers.netapp.common.NetAppDriver`` driver in a custom environment file to enable the NetApp back end and allow it to provision and manage shared file system storage.

Table of Contents

CHAPTER 1. INTRODUCTION TO NETAPP BACK ENDS	3
CHAPTER 2. REQUIREMENTS FOR NETAPP BACK ENDS	4
CHAPTER 3. CREATING THE NETAPP BACK END ENVIRONMENT FILE	5
CHAPTER 4. DEPLOYING THE SHARED FILE SYSTEMS SERVICE WITH NETAPP BACK ENDS	7
CHAPTER 5. CREATING A BASIC SHARE TYPE	8

CHAPTER 1. INTRODUCTION TO NETAPP BACK ENDS

With the OpenStack Shared File Systems service (manila) you can provision shared file systems that can be consumed by multiple compute instances.

This release supports the NetApp unified driver (**manila.share.drivers.netapp.common.NetAppDriver**). This driver allows the Shared File System service to use NetApp storage controllers (running Data ONTAP) as a back end.

The recommended method for configuring a Shared File System back end is through the director. Doing so involves writing a custom environment file.

With the director you can deploy the Shared File System with a NetApp back end on the overcloud.

CHAPTER 2. REQUIREMENTS FOR NETAPP BACK ENDS

Prerequisites

- A NetApp storage controller is deployed and ready to be used as a back end.
- You intend to use only one NetApp storage controller as a back end for your Shared File System service.
- You can use the director installation user account, which you create as part of the overcloud deployment. For more information about the director user account, see [Preparing for director installation](#) in the [Director Installation and Usage](#) guide.
- You want to install the Shared File System service on the Controller nodes. This is the default installation method.

This document does not contain information about the different deployment configurations possible for your NetApp back end. For more information about possible NetApp storage deployment configurations suitable for the Shared File System service, see the upstream [NetApp documentation](#) in particular, the [Theory of Operation and Deployment Choices](#).

After you complete the configuration for each NetApp back end, you can translate your configuration to a custom environment file and include this environment file in the deployment command. The director uses this file during deployment to orchestrate the configuration of your back end configuration and persists across overcloud updates.

CHAPTER 3. CREATING THE NETAPP BACK END ENVIRONMENT FILE

The director already includes default heat templates that configure most of the necessary settings to integrate a NetApp back end. You can also use a custom environment file to define settings specific to your deployment.

To start, log in as the **stack** user on the undercloud and create an environment file with the following contents:

`/home/stack/templates/netapp-config.yaml`

```
parameter_defaults:
  ManilaNetappLogin: 'NETAPP_USER' # 1
  ManilaNetappPassword: 'NETAPP_USER_PASSWORD'
  ManilaNetappServerHostname: 'HOSTNAME' # 2
  ManilaNetappVserver: 'SVM' # 3
  ManilaNetappRootVolumeAggr: 'ROOTVAGGR' # 4
  ManilaNetappTraceFlags: 'TRFLAGS' # 5
  ManilaNetappDriverHandlesShareServers: 'false' # 6
```

- 1 Replace `NETAPP_USER` and `NETAPP_USER_PASSWORD` with the credentials of the administrative account used to access the storage system (specifically, `HOSTNAME`).
- 2 Replace `HOSTNAME` with the storage system or proxy server. This value must be the IP address or hostname of either the cluster management logical interface (LIF) or Storage Virtual Machine (SVM) LIF.
- 3 `SVM` specifies the storage virtual machine, previously called a vserver, on the storage cluster where you want to provision shared file systems. This parameter is required if you want the driver to operate without managing share servers. That is, to be limited to the scope of a single SVM.
- 4 `ROOTVAGGR` specifies the name of the aggregate where you want to place the root volume when a new storage virtual machine (SVM) is created to correspond to a manila share server. This parameter is required if you set the value of **ManilaNetappDriverHandlesShareServers** to **true**, which means the driver manages the life cycle of share servers. This value is not required if the value of **ManilaNetappDriverHandlesShareServers** is **false**.
- 5 Replace `TRFLAGS` with a comma-separated list of options that control which trace info is written to the Shared File System service logs when the debug level is set to **True**. Supported values include **method** and **api**.
- 6 Set the **ManilaNetappDriverHandlesShareServers** parameter to **true** or **false** to enable or disable lifecycle management of the share server.

For example:

`/home/stack/templates/netapp-config.yaml`

```
parameter_defaults:
  ManilaNetappLogin: 'netapp_user'
  ManilaNetappPassword: 'netapp_user_password'
  ManilaNetappServerHostname: '10.8.18.108'
```

ManilaNetappVserver: 'vserver_1'
ManilaNetappRootVolumeAggr: 'aggr1_n1'
ManilaNetappTraceFlags: 'method,api'
ManilaNetappDriverHandlesShareServers: 'false'

CHAPTER 4. DEPLOYING THE SHARED FILE SYSTEMS SERVICE WITH NETAPP BACK ENDS

This section describes how to use the `/home/stack/templates/netapp-config.yaml` environment file to orchestrate the configuration of your NetApp back end.

After you create the `/home/stack/templates/netapp-config.yaml`, environment file, log in as the **stack** user on the undercloud and deploy the configured back end by running:

```
$ source ~/stackrc
$ openstack overcloud deploy --templates -e /usr/share/openstack-tripleo-heat-templates/environments/manila-netapp-config.yaml -e /home/stack/templates/netapp-config.yaml
```

The `/usr/share/openstack-tripleo-heat-templates/environments/manila-netapp-config.yaml` file is the default heat template provided with the director to deploy NetApp back ends for the Shared File System service. The `/home/stack/templates/netapp-config.yaml` file that you create in [Chapter 3, *Creating the NetApp back end environment file*](#) applies your custom configuration on top of the configuration in the default heat template.



IMPORTANT

If you passed any extra environment files when you created the overcloud, pass them again here using the `-e` option to avoid making undesired changes to the overcloud. For more information, see [Modifying the Overcloud Environment](#) in the [Director Installation and Usage](#) guide.

CHAPTER 5. CREATING A BASIC SHARE TYPE

Whenever you create a new share, you must specify a share type. If you do not specify a share type, the share creation process fails.

Director does not support automatic configuration or creation of the **default** share type during installation. However, director does set the **manila.conf** configuration option **default_share_type** to **default**. You must create the **default** share type after the overcloud has been deployed.

To create a basic share type named **default**, run the following commands as the **stack** user on the undercloud:

```
$ source ~/overcloudrc
$ manila type-create default false
```

In [Chapter 3, *Creating the NetApp back end environment file*](#), you set `ManilaNetappDriverHandlesShareServers` to **false**. In this example, `manila type-create default` is therefore also **false**, because the NetApp driver is not required to handle the life cycle of the share servers. If you set the `ManilaNetappDriverHandlesShareServers` parameter to **true**, you must also set the default share type to **true**. For more information about share types, see [Creating and Managing Share Types](#) in the [Red Hat OpenStack Platform Storage Guide](#).