Installing and using the Netapp Docker Plugin (nDVP) with DC/OS for Persistent Storage

Table of Contents

Document Purpose

The purpose of this document is to outline how to use the Netapp Docker Volume Plugin (nDVP) for use in a DC/OS cluster. In the proceeding sections, we will cover the prerequisites required for you NetApp system, as well as those for DC/OS. The nVDP leverages standard DVDI and DVDCLI commands for installation, setup, and administration for Volumes created via the plugin

Background

When considering the storage resources required to run micro services applications, containers, and stateful data frameworks, many organizations prefer to leverage existing storage backends re-factored for container-specific resource requirements across deployment and Day 2 Operations. The NetApp Docker Volume Plugin (nDVP) provides direct integration with the Docker ecosystem for NetApp's ONTAP, SolidFire, and E-Series storage platforms. The nDVP package supports the provisioning and management of storage resources from the storage platform to Docker hosts, with a robust framework for adding additional platforms in the future.

Multiple instances of the nDVP can run concurrently on the same host. This allows simultaneous connections to multiple storage systems and storage types, with the ability to customize the storage used for the Docker volume(s). When used on DC/OS, and in conjunction with External Persistent Storage, nDVP provides a robust way to provide large amount of storage backend, while maintain a high level of fault tolerance without human intervention

Prerequisites

nDVP is supported on the following operating systems:

- Debian , Ubuntu, 14.04+ (if not using iSCSI multipathing, 15.10+ with iSCSI multipathing). CentOS, 7.0+, and RHEL, 7.0+

Verify your storage system meets the minimum requirements:

- ONTAP: 8.3 or greater, SolidFire: ElementOS 7 or greater, or E-Series: Santricity

DC/OS

- SSH must be enabled on all agent nodes

- Root access to agent nodes for enabling and editing configurations on the nDVP

Installing the Netapp Docker Plugin

SSH to each agent node

a. Run 'dcos node' command, and note the "mesos-id" for each agent
b. SSH to each node by issuing the following command:

```
dcos node ssh --master-proxy --mesos-id=<mesos-id>
```

Installing the Managed Plugin (For each Agent in the cluster):

1. Ensure you have Docker Engine 17.03 (nee 1.13) or above installed.

   docker –version (*If your version is out of date, follow the instructions for your distribution to install*

2. Download and unpack the plugin package

```
# download and unpack the application
wget
https://github.com/NetApp/netappdvp/releases/download/v17.07.0/netappdvp-17.07.0.tar.gz
tar zxf netappdvp-17.07.0.tar.gz

# move to a location in the bin path
sudo mv netappdvp /usr/local/bin
sudo chown root:root /usr/local/bin/netappdvp
sudo chmod 755 /usr/local/bin/netappdvp

# create a location for the config files
sudo mkdir -p /etc/netappdvp
```

3. Create a configuration file. The config file must be located in the /etc/netappdvp directory. The default filename is config.json, however you can use any name you choose by specifying the config option with the file name

```
{
        "version": 1,
        "storageDriverName": "ontap-nas",
        "managementLIF": "10.0.0.1",
        "dataLIF": "10.0.0.2",
        "svm": "svm_nfs",
        "username": "vsadmin",
        "password": "netapp123",
        "aggregate": "aggr1",
        "defaults": {
        "size": "10G",
        "spaceReserve": "none",
        "exportPolicy": "default"
}
```

Note: the 'exportPolicy' must be specified for use with DC/OS. This designates the path to find backend storage resources on the aggregate associated with a NetApp array. It is set to a value of 'default' typically but may have been changed by your storage administrator. If so, please find the correct path before starting. For examples of sample config options please see NetApp's Configuration File Options

```
docker plugin install --grant-all-permissions --alias netapp netapp/ndvp-plugin:17.07
config=myConfigFile.json
```

5. Test to ensure nDVP can consume storage from the configured system.

```
# create a volume named "firstVolume"
docker volume create -d netapp --name firstVolume

# create a default volume at container instantiation
docker run --rm -it --volume-driver netapp --volume secondVolume:/my_vol alpine ash

# remove the volume "firstVolume"
docker volume rm firstVolume
```

Creating an External Persistent Volume Mount in DC/OS

Next, we will walk through an example JSON configuration for a Docker application that will be deployed through Marathon on DC/OS. This docker container will specify the use of a mounted volume on the nDVP to be used upon creation. If installed correctly in the prior steps, the nDVP will mount the volume to the backend NetApp aggregate successfully and the container will show a 'healthy" running status inside DC/OS

1. Log into your DC/OS UI and select Services

   'Add' a single container option

2. Copy and paste the following JSON configuration in the JSON Editor option in the top right. (*You can also save the json file to your local machine and run the 'dcos marathon app add name.json through the DC/OS CLI*)

   Note: *you will see under the options section that there is a parameter for 'dvdi/driver'. By default, the nDVP driver will be labeled 'netapp'. This must be specified. The 'name' is the name of The volume that will be mounted through the nDVP on the corresponding aggregate. This can validated with a DVDCLI command 'docker volume inspect'*

```
{
 "id": "/test-docker",
 "instances": 1,
 "cpus": 0.1,
 "mem": 32,
 "cmd": "/usr/bin/tail -f /dev/null",
 "container": {
  "type": "DOCKER",
  "docker": {
   "image": "alpine:3.1",
   "network": "HOST",
   "forcePullImage": true
  },
  "volumes": [
   {
    "containerPath":  "/data/testnetappvolume",
    "external": {
     "name": "my-test-vol",
     "provider": "dvdi",
     "options": { "dvdi/driver": "netapp" }
    },
    "mode": "RW"
   }
  ]
 },
 "upgradeStrategy": {
  "minimumHealthCapacity":  0,
  "maximumOverCapacity": 0
 }
}
```

Deploying and Troubleshooting

1. Go to the ‘Services” Tab and click on the container image that you just deployed

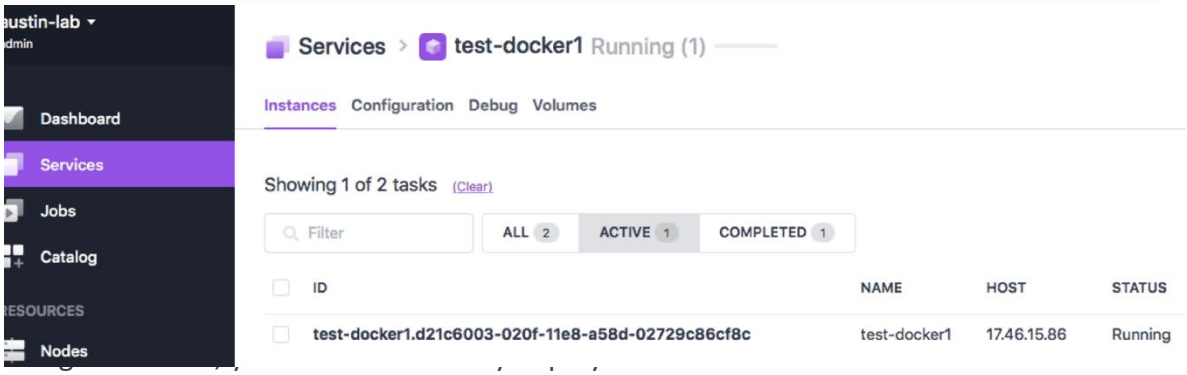
2. Note the status of the service task (Staging, Delayed, Waiting, or Running). It should say running relatively quickly. If it does not say this, please do the following:

    a. Go to Debug – Logs and note any errors

    b. If you see a Volume mount error, the most likely cause of this is due to your export path not being specified

        i. If you configured a new config.json which was updated, then you will need to remove all existing volume mounts, disable the plugin, update the config.json and update the plugin on each of your nodes so that it uses the correct configuration. Docker will also need to be restarted

3. If you see no errors, you should now see a ‘Volumes’ section under your container service deployment listing the name and path of the volume that is mounted. On the nDVP, you should see the DVDICLI list show the volume mounted. On NetApp, you should be able to verify the same thing.



persistent volume tied to your NetApp storage!

4. If you cannot verify success, please see the Troubleshooting section of the Appendix for more troubleshooting guides

Useful DVDI Commands

Using these options during the docker volume create operation is super simple, just provide the option and the value using the `-o` operator during the CLI operation. These override any equivalent vales from the JSON configuration file.

```
# create a 10GiB volume
docker volume create -d netapp --name demo -o size=10G -o encryption=true

# create a 100GiB volume with snapshots
docker volume create -d netapp --name demo -o size=100G -o snapshotPolicy=default

# create a volume which has the setUID bit enabled
docker volume create -d netapp --name demo -o unixPermissions=4755

# reload systemd for it to ingest changes
systemctl daemon-reload

# enable the service, note this name will change depending on what you named the
# file in the /usr/lib/systemd/system directory
systemctl enable netappdvp

# start the service, see note above about service name
systemctl start netappdvp

# view the status
systemctl status netappdvp

# stop all running instances
pkill /usr/local/bin/netappdvp

# restart docker
systemctl restart docker
```

Minimum ONTAP User Permissions required for nDVP

nDVP does not need full permissions on the ONTAP cluster and should not be used with the cluster-level admin account. Below are the ONTAP CLI comands to create a dedicated user for nDVP with specific permissions.

```
# create a new nDVP role
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname DEFAULT -access none

# grant common nDVP permissions
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "event
generate-autosupport-log" -access all
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "network interface"
-access readonly
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "version" -access
readonly
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "vserver" -access
readonly
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "vserver nfs show"
-access readonly
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "volume" -access all
```

```
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "snapmirror" -access
all

# grant ontap-san nDVP permissions
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "vserver iscsi show"
-access readonly
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "lun" -access all


# grant ontap-nas-economy nDVP permissions
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "vserver
export-policy create" -access all
security login role create -vserver [VSERVER] -role ndvp_role -cmddirname "vserver
export-policy rule create" -access all

# create a new nDVP user with nDVP role
security login create -vserver [VSERVER] -username ndvp_user -role ndvp_role -application
ontapi -authmethod password
```

Appendix

Quick start
http://netappdvp.readthedocs.io/en/latest/quick_start.html

Host Prerequisites
http://netappdvp.readthedocs.io/en/latest/install/host_config.html

Configuration Options
http://netappdvp.readthedocs.io/en/latest/install/ndvp_ontap_config.html#

User Permissions
https://github.com/NetApp/netappdvp/blob/master/docs/install/ndvp_ontap_config.rst

Using nDVP
http://netappdvp.readthedocs.io/en/latest/use/index.html

Troubleshooting
http://netappdvp.readthedocs.io/en/latest/support.html