



Configuring Pure Storage as a Cinder storage provider for IBM PowerVC

Simon Dodsley, Technical Director

Version: 2.2 26 February 2019

Contents

Notices	3
Summary.....	4
Prerequisites	4
Registering the Pure Storage FlashArray pluggable driver	4
Install the Pure Storage Python SDK	4
Obtain the pluggable driver for PowerVC.....	4
Create the Cinder configuration file.....	5
Register the pluggable driver	6
Provisioning Storage Volumes	9
Adding Multiple Pure Storage FlashArrays	10
About the Author	11

Notices

© 2018 Pure Storage, Inc. All rights reserved. Pure Storage, Pure1, and the P Logo are trademarks or registered trademarks of Pure Storage, Inc. in the U.S. and other countries. OpenStack is a registered trademark of the OpenStack Foundation, registered in many jurisdictions worldwide. IBM and PowerVC are registered trademarks of the IBM Corporation, registered in many jurisdictions worldwide. All other trademarks are registered marks of their respective owners.

The Pure Storage products and programs described in this documentation are distributed under a license agreement restricting the use, copying, distribution, and decompilation/reverse engineering of the products. No part of this documentation may be reproduced in any form by any means without prior written authorization from Pure Storage, Inc. and its licensors if any. Pure Storage may make improvements and/or changes in the Pure Storage products and/or the programs described in this documentation at any time without notice.

THIS DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. PURE STORAGE SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

Pure Storage, Inc.
650 Castro Street
Mountain View, CA 94041

Summary

This document covers the configuration process required to enable a Pure Storage® FlashArray to be used as a pluggable driver providing Cinder Block Storage for IBM® PowerVC® distributions.

Prerequisites

- Ensure your IBM PowerVC environment is up and running.
- Your Pure Storage Array should be available in the management network or routed to the management network.
- Ensure the FlashArray fibre channel ports are correctly configured.
- The Pure Storage management IP ports must have connectivity from all the controller and compute nodes.

Registering the Pure Storage FlashArray pluggable driver

There are only three things required by PowerVC to register the Pure Storage FlashArray:

- The Pure Storage Python SDK
- The pluggable driver supported by OpenStack
- The Cinder configuration file

All these steps require you to be logged on as the `root` user, or to have `root` privileges.

Install the Pure Storage Python SDK

It is necessary to install the Pure Storage Python SDK kit as the Pure Storage Cinder driver utilizes this. The kit is publicly available and is installed using the following command:

```
pip install purestorage
```

Ensure that you have the `python-pip` package installed prior to running this command, using the appropriate package installation procedures for your Red Hat Linux distribution.

Obtain the pluggable driver for PowerVC

This section is required as IBM do not incorporate in-tree OpenStack Cinder drivers from 3rd party arrays and therefore you must download the Pure Storage FlashArray driver.

The Pure Storage FlashArray driver's compatible with different versions of PowerVC are listed in the following table:

PowerVC version	Driver location
1.3.0	https://raw.githubusercontent.com/openstack/cinder/stable/liberty/cinder/volume/drivers/pure.py
1.3.1	https://raw.githubusercontent.com/openstack/cinder/stable/mitaka/cinder/volume/drivers/pure.py
1.3.2	https://raw.githubusercontent.com/openstack/cinder/stable/newton/cinder/volume/drivers/pure.py
1.3.3	https://raw.githubusercontent.com/openstack/cinder/stable/ocata/cinder/volume/drivers/pure.py
1.4.0	https://raw.githubusercontent.com/openstack/cinder/stable/pike/cinder/volume/drivers/pure.py
1.4.1 & 1.4.2	https://raw.githubusercontent.com/openstack/cinder/stable/queens/cinder/volume/drivers/pure.py

Download the appropriate file and move it to the following directory:

```
/usr/lib/python2.7/site-packages/cinder/volume/drivers/
```

and ensure that the file is executable.

Create the Cinder configuration file

The properties the pluggable driver registration will use are identified in a file that will be specified in the registration command later in this document.

For the Pure Storage FlashArray, the required properties that need to be added are listed below. These are added to a file, which is passed as a parameter when registering the pluggable driver.

The format for the configuration file for PowerVC v1.3.x is as follows:

```
san_ip = <Management VIP address/FQDN of Pure Storage FlashArray>
pure_api_token = <API token obtained from FlashArray GUI>
use_multipath_for_image_xfer = True
volume_pool_name = FlashArray
default_volume_type = FlashArray
volume_name_template = pvcpure-%s
```

For PowerVC v1.4.x the format is slightly different:

```
[backend_defaults]
san_ip = <Management VIP address/FQDN of Pure Storage FlashArray>
pure_api_token = <API token obtained from FlashArray GUI>
use_multipath_for_image_xfer = True
volume_pool_name = FlashArray
```

```
default_volume_type = FlashArray
volume_name_template = pvcpure-%s
```

Note: The `volume_pool_name` and `default_volume_type` parameters are just names used for the registration.

The values in brackets are specific to your Pure Storage FlashArray. For more details on obtaining the API token refer to the Pure Storage OpenStack Best Practices guides available from the [Pure Storage Support website](#).

If you have a FlashArray that is running Purity v5.1 or higher it is recommended that you create a new local user on the array with `array_admin` privileges to be used just for PowerVC. Ensure that you have created an API token for this user and use that as the value for the `pure_api_token` parameter.

When all the values required have been obtained, open a new file for editing on your PowerVC management server. The file name should be in the form `.conf`, with the values on separate lines in the file.

This file is temporary and therefore may be stored anywhere you like on the management server. For this example, we will create the file at `/root/flasharray.conf`. The file is only used to configure the `cinder.conf` file when registering the plugin. Once the plugin registered with the command this file isn't needed anymore and may be deleted.

Register the pluggable driver

Run the `powervc-register` command to register the plugin and create the Pure Storage FlashArray as a storage provider to PowerVC. This might take a few minutes to complete.

Note that this command must be run as `root` and will prompt for the `root` password.

The command and parameters look like this:

```
powervc-register -o add -r storage -d <driver> -n <display name> -p <property file>
```

- o is the action option;
- r is the registration type;
- d is the driver name for Pure Storage FlashArray plugin;
- n is the storage provider plugin name;
- p is the command parameter conf file from above.

For more information on the `powervc-register` command, you can run `powervc-register -h` to display additional options and help.

To register the Pure Storage FlashArray driver use the following command:

```
powervc-register -o add -r storage -d cinder.volume.drivers.pure.PureFCDriver -n  
PURE -p /root/flasharray.conf
```

When the command completes, you can list the registered plugins:

```
# powervc-register -o list -r storage
Enter password for root:

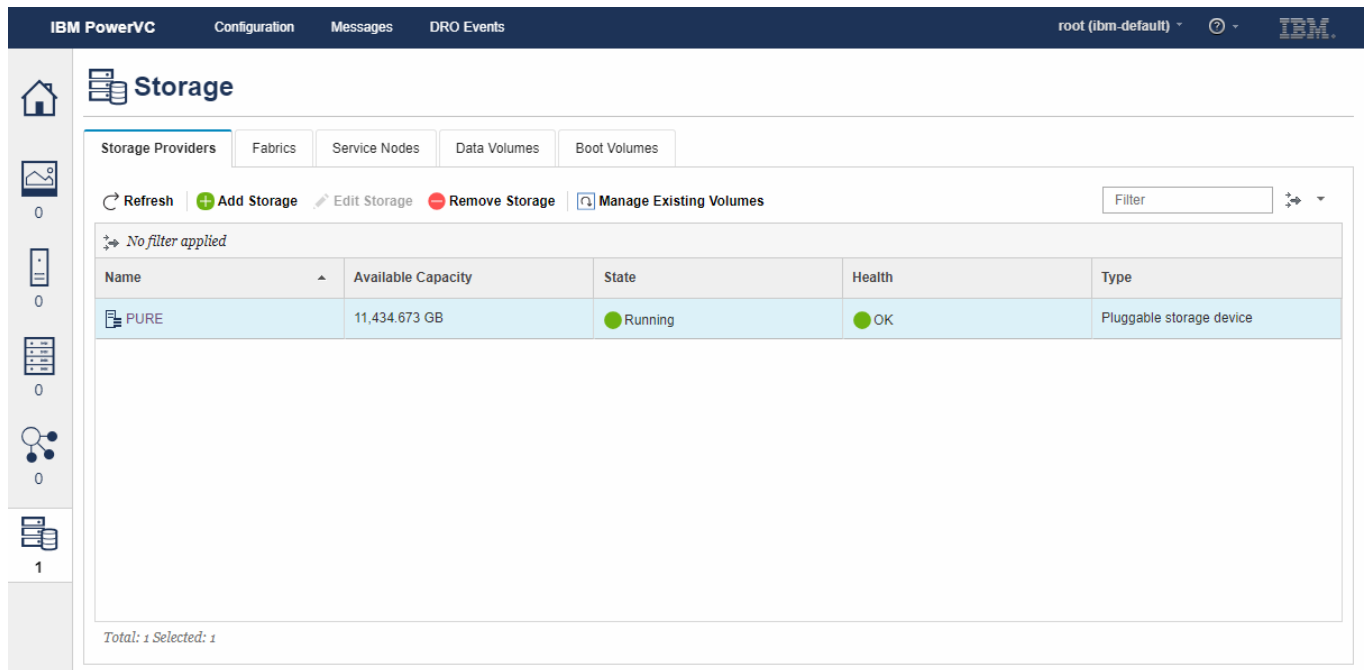
Name                Display Name          Type
====                =
generic0            PURE                  generic
```

When the command returns successfully, your Pure Storage FlashArray is now registered as a pluggable driver and you can begin management operations through the PowerVC user interface.

The new device will be listed under the Storage Providers tab with its type identifying it as a Pluggable storage device.

Note that the Pure Storage FlashArray plugin is available to all PowerVC projects.

Below is a screenshot of the PowerVC management interface after the Pure Storage pluggable driver has been successfully installed.



And by selecting the Storage Provider we see some more details, including any volumes created by PowerVC:

IBM PowerVC Configuration Messages DRO Events root (ibm-default)

Storage > Storage Provider: PURE

Storage Provider: PURE

Refresh Edit Remove

Information

Name:	PURE
State:	Running
Health:	OK
Available capacity:	11,434.7 GB
Total capacity:	11,434.7 GB
Type:	Pluggable storage device

Volumes

Refresh Create Delete Manage Existing Unmanage Filter

No filter applied

Name	Size (GB)	State	Health	Storage Template	Description
test	15	Available	OK	PURE base template	

Total: 1 Selected: 0

Storage Templates

Filter

Name	Type	Storage Pool	Default
PURE base template	Pluggable storage device		Yes

Total: 1

We can also examine the Storage Template we created with the configuration file, but there is no facility to edit this template.

IBM PowerVC Configuration Messages DRO Events root (ibm-default)

Configuration > Storage Templates > Storage Template: PURE base template

Storage Template: PURE base template

Learn about storage templates

Refresh Delete

Information

Name:	PURE base template
ID:	2b668e98-a143-4173-bddf-3e893c763366
Storage provider:	PURE
Default:	Yes
Type:	Pluggable storage device

Storage Volume Usage

Filter

No filter applied

Name	Size (GB)	State	Health	Description
test	15	Available	OK	

Total: 1

Provisioning Storage Volumes

The process of provisioning storage volumes from a provider created by a pluggable driver is the same as that for an integrated storage device.

Here we show the volume creation window in PowerVC

Create Volume

Specify the details for this storage volume.

* Storage template: PURE base template [Learn about storage templates](#)

* Volume name:

Description:

* Size (GB):

* Number of volumes:

Enable sharing

Current Storage Used
0 GB Used 11,434.6 GB Total
0%

When creating a volume for a pluggable storage provider, the projected usage cannot be determined. Therefore, it is possible that volume creation will not be successful.

Storage Provider: PURE
Available Capacity: 11,434.6 GB

And after the volume has been created we can see it has been created on the Pure Storage FlashArray

Storage 1 ×

Array Hosts **Volumes** Protection Groups Pods

+ > Volumes

Size	Data Reduction	Volumes	Snapshots	Shared	System	Total
3434499 M	1.0 to 1	0.00	0.00	-	0.00	0.00

Volumes General Space QoS 1-1 of 1 < > +

Name	Source	#Hosts	Serial
cinder			
volume-test-2ce32f0d-3cc2-cinder			
0 BDE00B13DE084CE300013573			

Destroyed (1) ▾

Volume Groups 0 of 0 < > +

Name	# Volumes	Size	Volumes	Snapshots	Reduction

No volume groups found.

Destroyed (2) ▾

Adding Multiple Pure Storage FlashArrays

Should you wish to increase the number of Pure Storage FlashArrays available to your PowerVC environment this is easily achieved by following the exact methodology above.

You must ensure that the IP address and API token in the configuration file are correct for your subsequent FlashArrays. Additionally, when running the `powervc-register` command, you must use a different value for the `-n` parameter.

```
# powervc-register -o list -r storage
Enter password for root:

Name                Display Name        Type
====                =====
generic0            PURE                generic
generic1            PURE-2              generic
```

From the PowerVC GUI this would be seen like this:

The screenshot shows the IBM PowerVC GUI interface. The top navigation bar includes 'IBM PowerVC', 'Configuration', 'Messages', and 'DRO Events'. The user is logged in as 'root (ibm-default)'. The main content area is titled 'Storage' and has tabs for 'Storage Providers', 'Fabrics', 'Service Nodes', 'Data Volumes', and 'Boot Volumes'. The 'Storage Providers' tab is active, showing a table of storage providers. The table has columns for Name, Available Capacity, State, Health, and Type. Two providers are listed: 'PURE' and 'PURE-2', both with 11,434.645 GB available capacity, in a 'Running' state with 'OK' health, and identified as 'Pluggable storage device'. The 'PURE' row is highlighted with a dashed blue border.

Name	Available Capacity	State	Health	Type
PURE	11,434.645 GB	Running	OK	Pluggable storage device
PURE-2	11,434.645 GB	Running	OK	Pluggable storage device

About the Author



As Director of New Stack, Simon is responsible for developing the technical direction of Pure Storage pertaining to Open Source technologies including OpenStack, Containers, and associated orchestration and automation toolsets. His responsibilities also include developing best practices, reference architectures and configuration guides.

With over 30 years of storage experience across all aspects of the discipline, from administration to architectural design, Simon has worked with all major storage vendors' technologies and organisations, large and small, across Europe and the USA as both customer and service provider.

Blog: <http://blog.purestorage.com/author/simon>



Pure Storage, Inc.
Twitter: [@purestorage](https://twitter.com/purestorage)
www.purestorage.com

650 Castro Street, Suite #260
Mountain View, CA 94041

T: 650-290-6088
F: 650-625-9667

Sales: sales@purestorage.com
Support: support@purestorage.com
Media: pr@purestorage.com
General: info@purestorage.com