Mounting a VVol Datastore with PowerCLI

```
PS C:\Windows\system32> get-help Mount-PfaVvolDatastore

NAME
    Mount-PfaVvolDatastore

SYNOPSIS
    Mounts a FlashArray VVol Datastore to a host or cluster


SYNTAX
    Mount-PfaVvolDatastore [[-flasharray] <PureArray>] [[-datastoreName] <String>] [[-datastore] <Datastore>] [-cluster] <Cluster>
    [[-protocolEndpoint] <String>] [[-vasaArray] <VasaStorageArray>] [<CommonParameters>]


DESCRIPTION
    Mounts a FlashArray VVol Datastore to a host or cluster, connects a PE to the cluster if not present.


RELATED LINKS

REMARKS
    To see the examples, type: "get-help Mount-PfaVvolDatastore -examples".
    For more information, type: "get-help Mount-PfaVvolDatastore -detailed".
    For technical information, type: "get-help Mount-PfaVvolDatastore -full".
```

I've been making a lot of updates to my PowerShell module around VVols recently and this was the last "table stakes" cmdlet I wanted to add. There are certainly more to come, but now we definitely have the basics. In 1.2.2.1 release of the PowerShell module I added a cmdlet called Mount-PfaVvolDatastore.

As of today we support a single VVol datastore–though we are working on adding support for more than one.


```
PS C:\Windows\system32> get-help Mount-PfaVvolDatastore

NAME
    Mount-PfaVvolDatastore

SYNOPSIS
    Mounts a FlashArray VVol Datastore to a host or cluster


SYNTAX
    Mount-PfaVvolDatastore [[-flasharray] <PureArray>] [[-datastoreName] <String>] [[-datastore] <Datastore>] [-cluster] <Cluster>
    [[-protocolEndpoint] <String>] [[-vasaArray] <VasaStorageArray>] [<CommonParameters>]


DESCRIPTION
    Mounts a FlashArray VVol Datastore to a host or cluster, connects a PE to the cluster if not present.


RELATED LINKS

REMARKS
    To see the examples, type: "get-help Mount-PfaVvolDatastore -examples".
    For more information, type: "get-help Mount-PfaVvolDatastore -detailed".
    For technical information, type: "get-help Mount-PfaVvolDatastore -full".
```


This cmdlet behaves as the following:

- Checks to see if a protocol endpoint for the specified array is present to the cluster. If not it presents one. You can specify a specific one if you so choose, but there is not much of a use case for that today (there will be in the future)
- Rescans the cluster and makes sure the PE is seen. If it is not, the process will fail. Likely a network issue (iSCSI not configured, zoning etc)

- Mounts the VVol datastore to the cluster
- Returns the datastore

You have a few options on how to specify which VVol datastore (as of writing this is analogous to an FlashArray since it is 1:1):

- Pass in a VVol datastore. Basically this means, hey mount this same VVol datastore to this other cluster.
- A FlashArray PowerShell connection. I will decipher the VVol datastore and mount it.
- A VASA Storage array. This is the representation of a storage array in the VMware context (see Get-VasaStorageArray).

If the datastore has never been mounted, you need to specify a name as well. If it is already mounted somewhere I will just find the name for you.

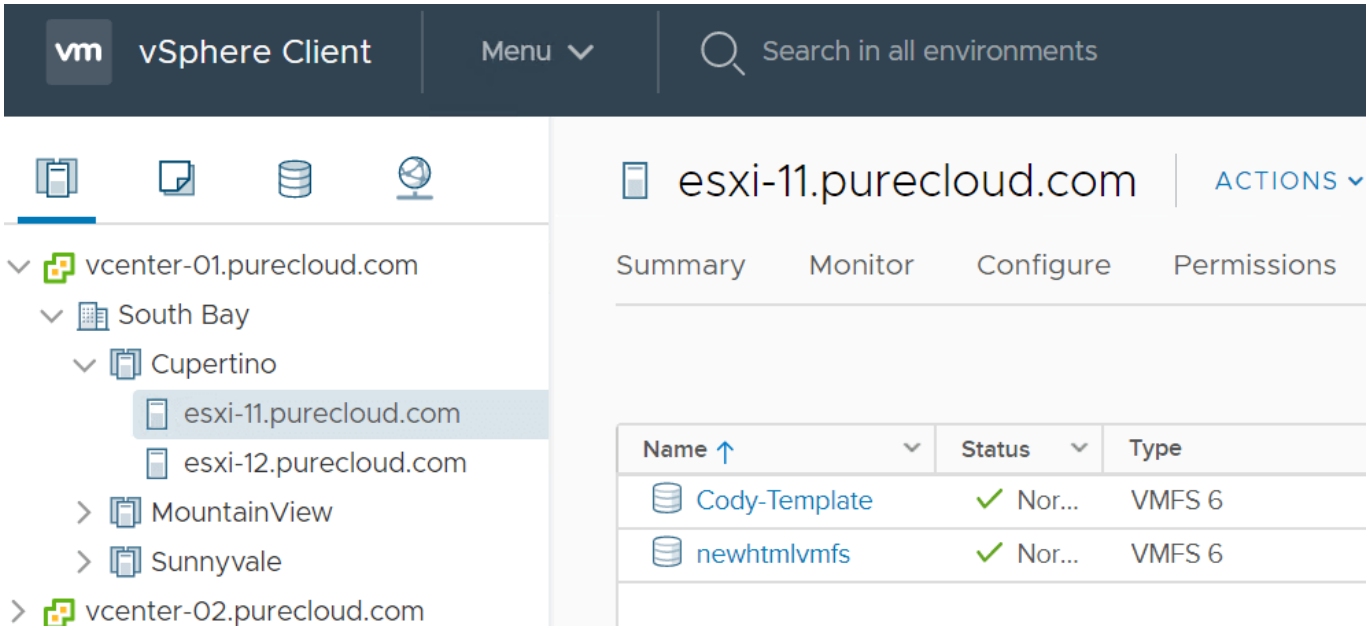
You do NOT need to pass in a FlashArray connection if the PE has already been presented–this is only required if the PE is not already presented to the cluster. So if you don't have access to the FlashArray, ask that it be presented to the cluster. Whether a PE is presented to a cluster is how the array allows or disallows VVol access to that cluster. Or pass in the connection and I will do it.

Example 1

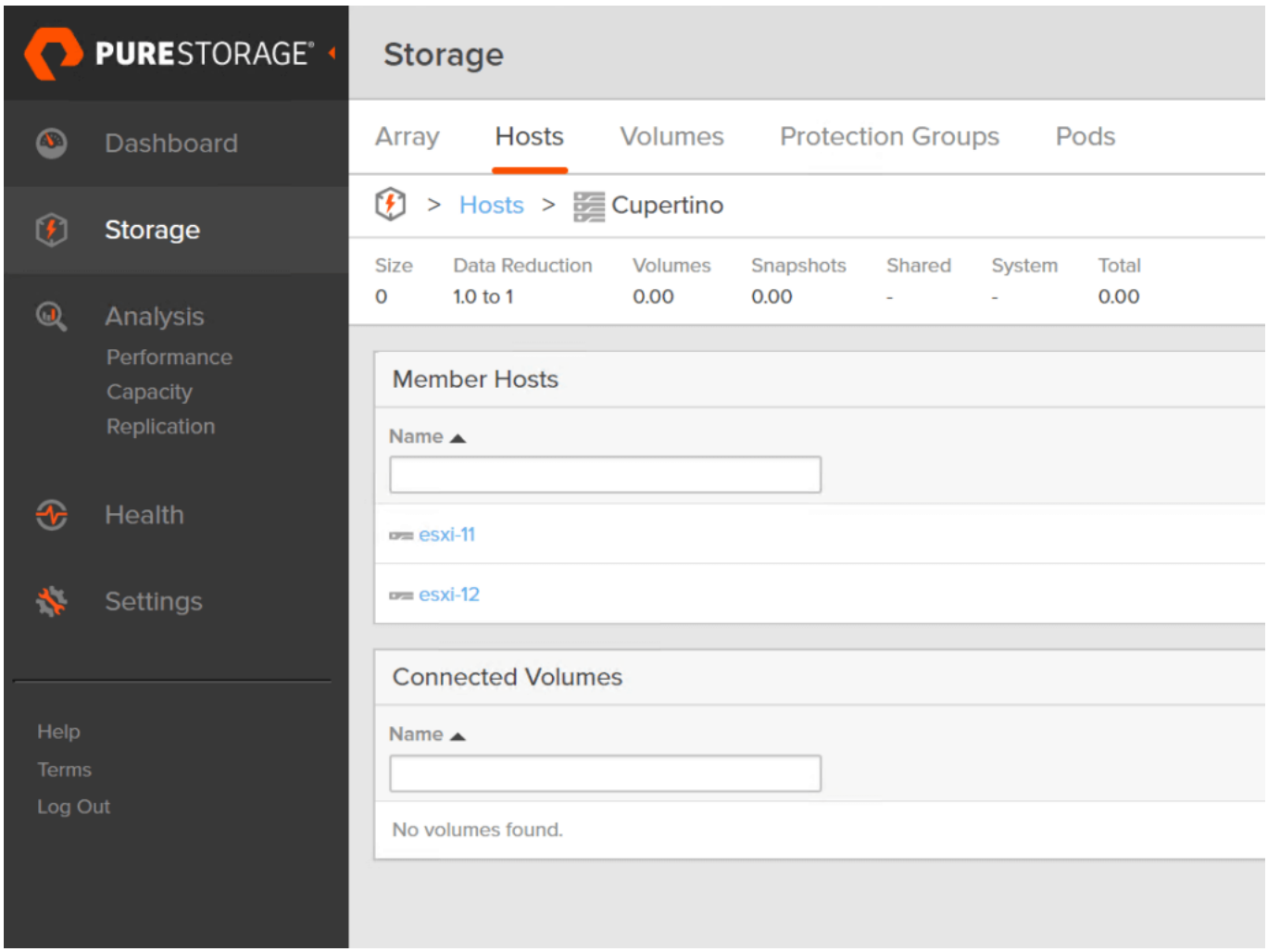
Let's walk through a common example. This is a new VVol datastore and the PE has not been presented. So this means I need to pass in three things, a cluster, a FlashArray connection, and a datastore name.

So I connect to vCenter and create my FlashArray connection:

[crayon-6424c53623c7e752207558/]
As you can see the VVol datastore is not currently mounted:



And the PE is not connected:

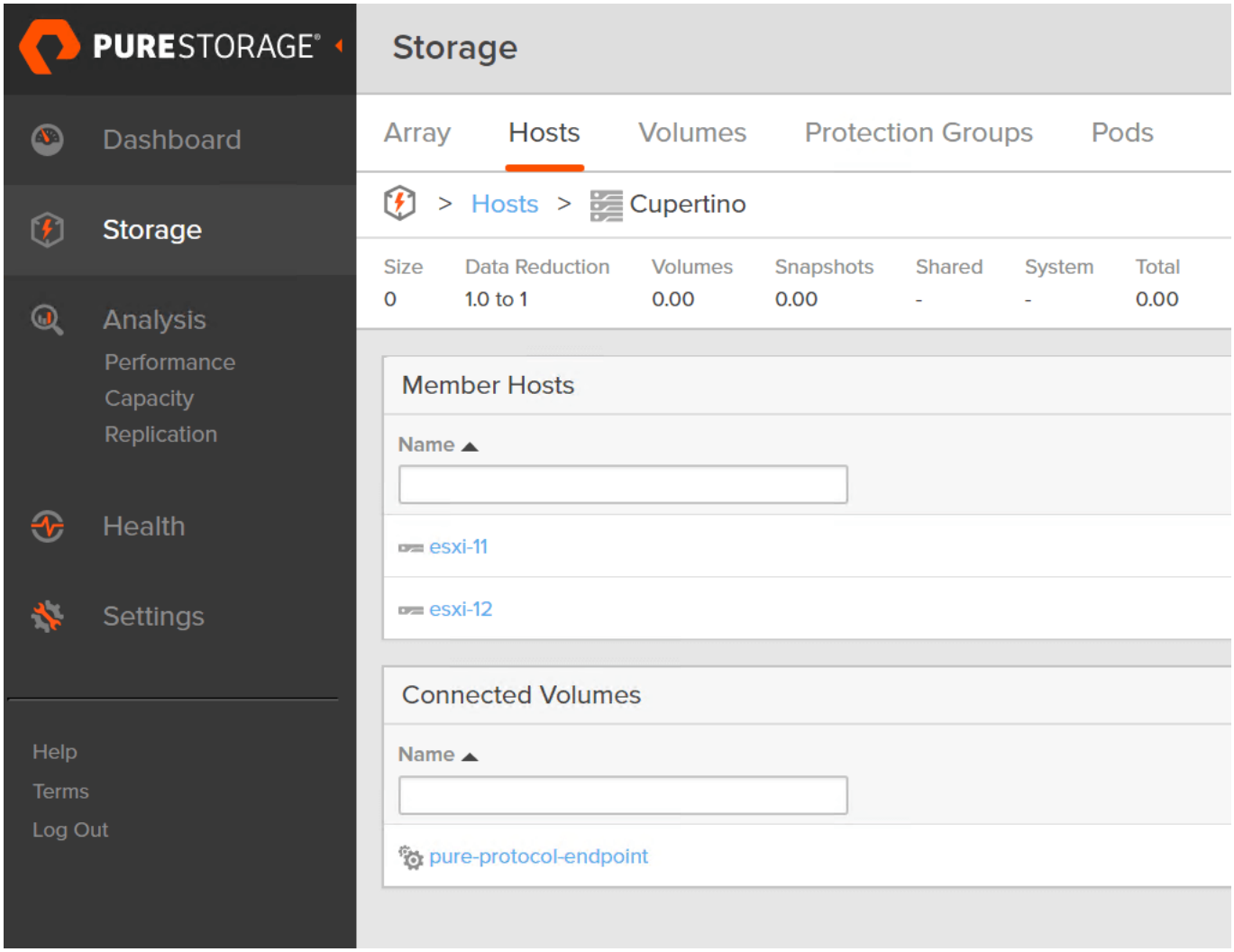


Let’s now run Mount-PfaVvolDatastore.

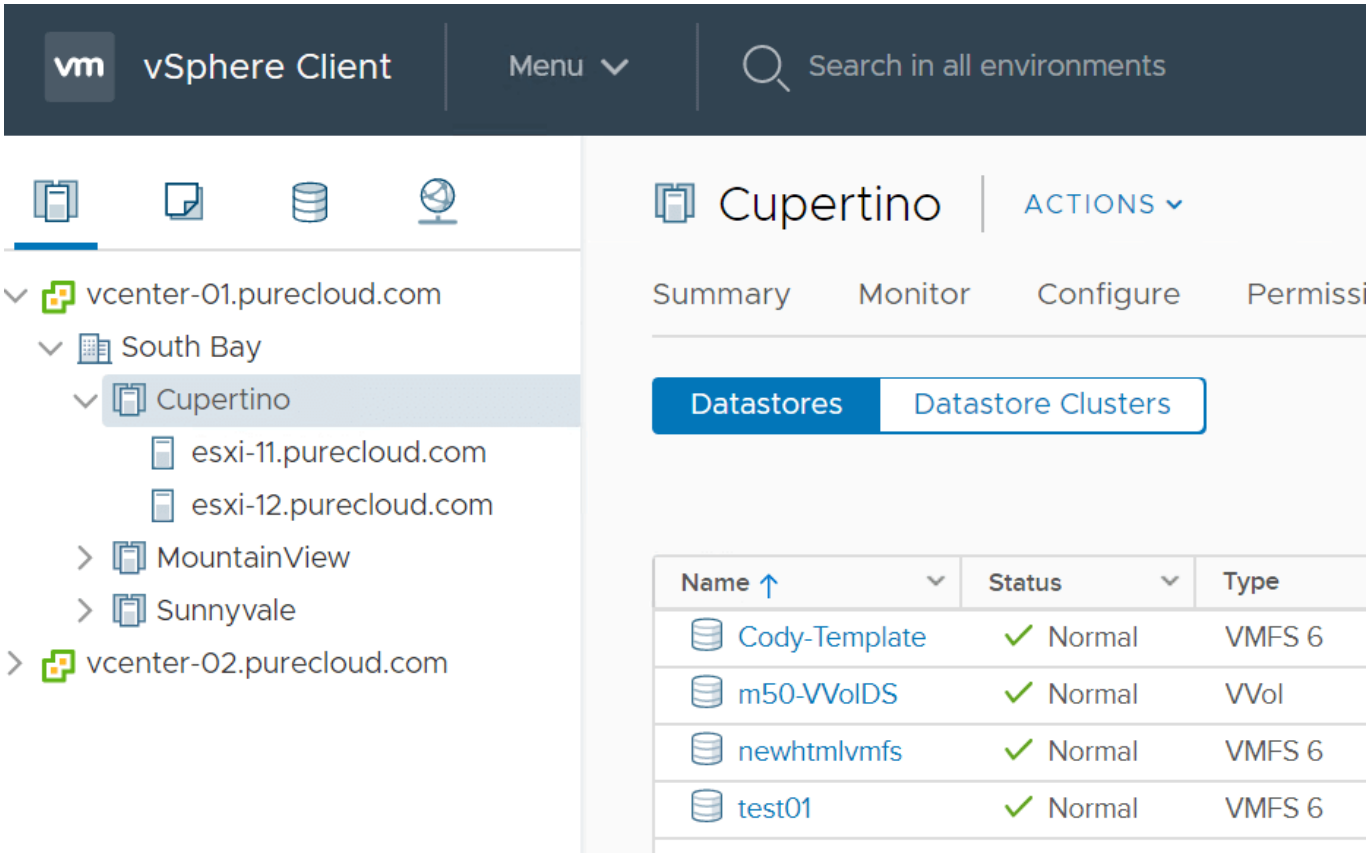
[crayon-6424c53623c88946379748/]

```
PS C:\Windows\system32>
PS C:\Windows\system32> Mount-PfaVvolDatastore -flasharray $flasharray -cluster (get-cluster Cupertino) -datastoreName m50-vvolDS

Name                                 FreeSpaceGB      CapacityGB
----                                 -----------      ----------
m50-vvolDS                      4,194,232.260    4,194,304.000
```

We can see that the PE is now connected:

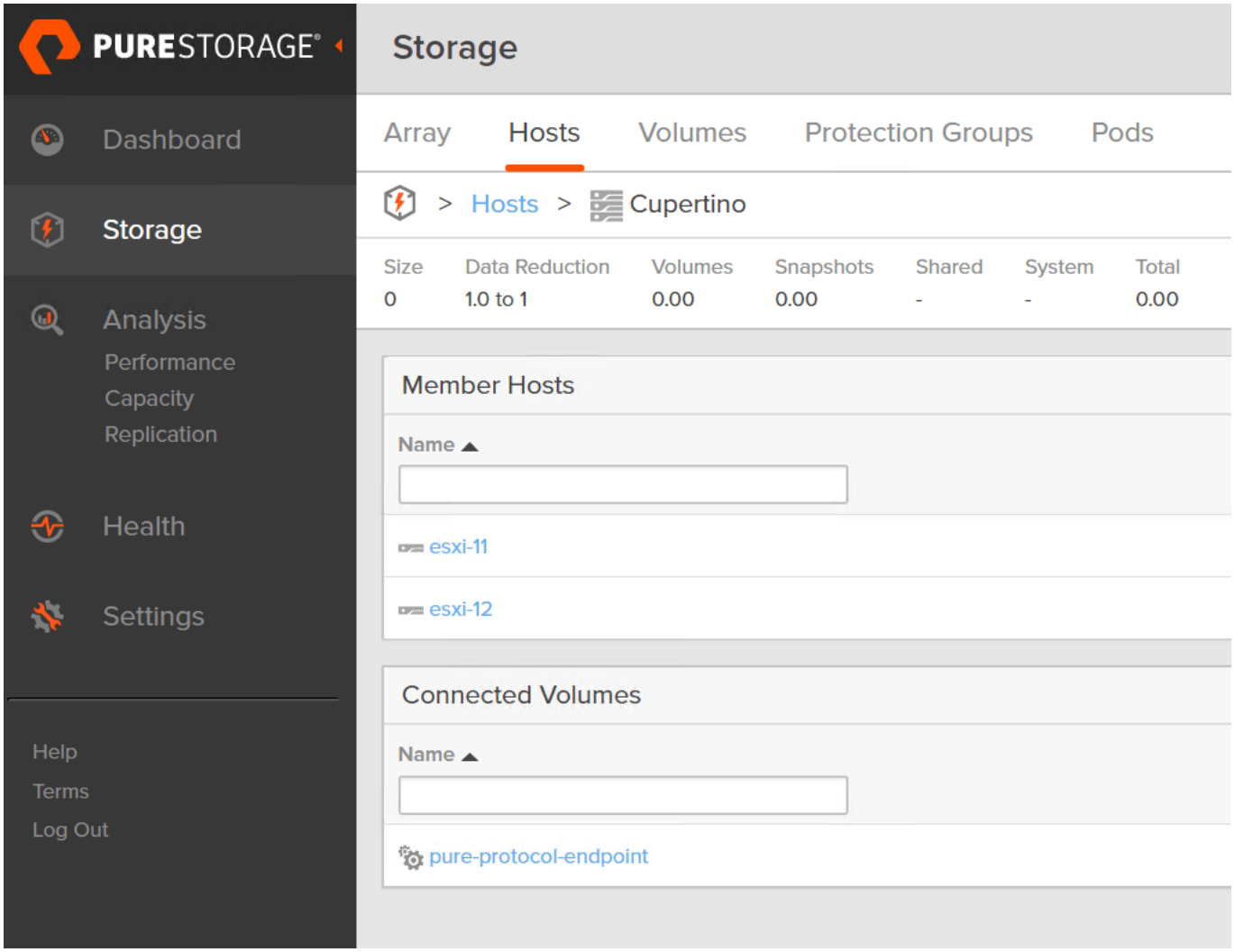


And the datastore has been mounted to the cluster:



Example 2

What if you don't have access to the array? Well in this case, let's presume the protocol endpoint is already presented to the cluster. Yay! We've been enabled for VVols!



So now I only need to mount the VVol datastore. If the VVol has never been mounted before, and I cannot create a PowerShell array connection (I am not authorized in this situation), how do I say what datastore I want to mount? Well this goes back to the cmdlet I referenced built into PowerCLI: Get-VasaStorageArray:

```
PS C:\Windows\system32> Get-VasaStorageArray

Name                           ModelId              VendorId
----                           -------              --------
5afcb17b-a3b7-f390-3d7d-002... IOFILTERS           VMware Inc
sn1-m20-c09-25                 FA-m20               PURE
5aeb0571-cfbd-d074-ef7a-002... IOFILTERS           VMware Inc
5aeb0e65-5fe6-3a40-a12e-002... IOFILTERS           VMware Inc
5aeb0cac-d540-6495-7a77-002... IOFILTERS           VMware Inc
5cb49884-57b2-4330-ea46-002... IOFILTERS           VMware Inc
5cb49759-661b-1be8-e33f-002... IOFILTERS           VMware Inc
sn1-m50-c10-30                 FA-m50               PURE
```

So by passing in the array I want and and a datastore name, I am good to go (and a cluster):
[crayon-6424c53623c8a535860537/]

```
PS C:\Windows\system32> $vasaArrays = Get-VasaStorageArray
PS C:\Windows\system32> Mount-PfaVvolDatastore -vasaArray $vasaArrays[7] -cluster (get-cluster Cupertino) -datastoreName m50-VVolDS

Name                           FreeSpaceGB      CapacityGB
----                           -----------      ----------
m50-VVolDS                    4,194,232.258   4,194,304.000


PS C:\Windows\system32> _
```

Wrapping up

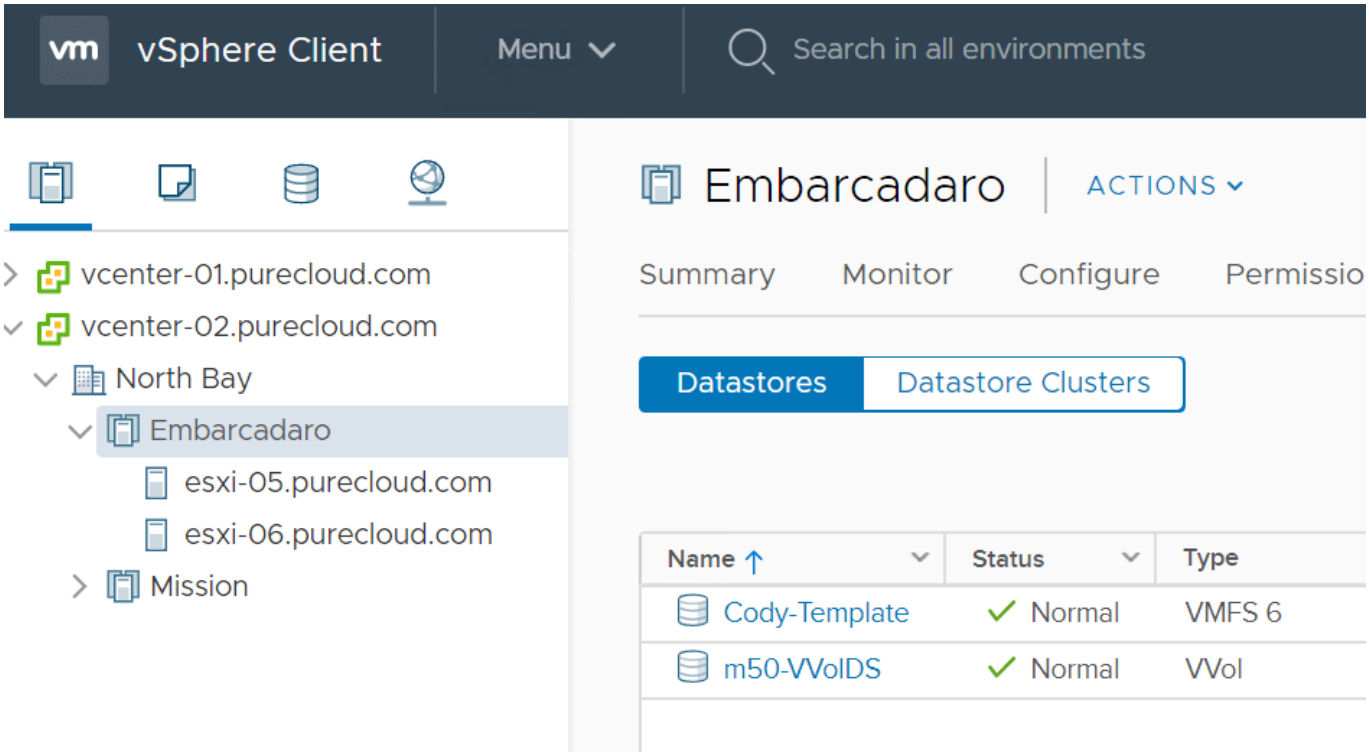
Let's review setting up VVols from scratch on the FlashArray with PowerCLI now. There are five steps (once the FlashArray has a management IP):

- Install the PowerShell module
- Connect to the FlashArray
- Connect to vCenter
- Create the host group (and in the case of iSCSI, configure iSCSI targets)
- Register VASA
- Mount the VVol datastore

The whole script:
[crayon-6424c53623c91764355675/]

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> install-module PureStorage.FlashArray.VMware
PS C:\Windows\system32> $flasharray = new-pfaConnection -endpoint flasharray-m50-1 -credentials (get-credential) -ignoreCertificateError -
nDefaultArray

cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:
Credential
PS C:\Windows\system32> connect-viserver -Server vcenter-02

Name                           Port  User
----                           ----  ----
vcenter-02                     443   PURECLOUD\cody


PS C:\Windows\system32> New-PfaHostGroupfromVcCluster -cluster (get-cluster Embarcadaro) -iscsi -flasharray $flasharray

hosts              name
-----              ----
{esxi-05, esxi-06} Embarcadaro


PS C:\Windows\system32> Set-ClusterPfaiSCSI -cluster (get-cluster Embarcadaro) -flasharray $flasharray

Address            Port  Type
-------            ----  ----
10.21.88.220       3260  Send
10.21.88.150       3260  Send
10.21.88.151       3260  Send
10.21.88.220       3260  Send
10.21.88.150       3260  Send
10.21.88.151       3260  Send
10.21.88.220       3260  Send
10.21.88.150       3260  Send
10.21.88.151       3260  Send
10.21.88.220       3260  Send
10.21.88.150       3260  Send
10.21.88.151       3260  Send


PS C:\Windows\system32> New-PfaVasaProvider -flasharray $flasharray -credentials (get-credential)

cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:
Credential

Name               Status       VasaVersion LastSyncTime           Namespace            Url
----               ------       ----------- ------------           ---------            ---
sn1-m50-c10-30-CT0 online       3.0         7/16/2019 8:38:25 PM   com.purestorage      https://10.21.88.26:8084
sn1-m50-c10-30-CT1 online       3.0                                com.purestorage      https://10.21.88.25:8084


PS C:\Windows\system32> Mount-PfaVvolDatastore -flasharray $flasharray -cluster (get-cluster Embarcadaro) -datastoreName m50-VVolDS
Name                           FreeSpaceGB      CapacityGB
----                           -----------      ----------
m50-VVolDS                    4,194,232.256   4,194,304.000
```

Done!:


vSphere Client
Menu
Search in all environments
vcenter-01.purecloud.com
vcenter-02.purecloud.com
North Bay
Embarcadaro
esxi-05.purecloud.com
esxi-06.purecloud.com
Mission
Embarcadaro
ACTIONS
Summary
Monitor
Configure
Permissio
Datastores
Datastore Clusters
Name
Status
Type
Cody-Template
Normal
VMFS 6
m50-VVolDS
Normal
VVol