

# PowerCLI and VVols Part VI: Running a Test Failover

```
PS C:\Users\Administrator> $vm = get-vm vm-001
PS C:\Users\Administrator> $sourceGroup = $vm |Get-SpbmReplicationGroup
PS C:\Users\Administrator> $sourceGroup

Name                               ReplicationState
----                               -
sn1-x70-b05-33:vvol-1hr           Source
```

This post I will talk about using PowerCLI to run a test failover for VVol-based virtual machines. One of the many nice things about VVols is that in the VASA 3.0 API this process is largely automated for you. The SRM-like workflow of a test failover is included—so the amount of storage-related PowerShell you have to manually write is fairly minimal.

- [PowerCLI and VVols Part I: Assigning a SPBM Policy](#)
- [PowerCLI and VVols Part II: Finding VVol UUIDs](#)
- [PowerCLI and VVols Part III: Getting VVol UUIDs from the FlashArray](#)
- [PowerCLI and VVols Part IV: Correlating a Windows NTFS to a VMDK](#)
- [PowerCLI and VVols Part V: Array Snapshots and VVols](#)
- [PowerCLI and VVols Part VI: Running a Test Failover](#)

With VVols, you don't really failover a single VM, you fail over a replication group. While it is certainly possible to just recover a single VM from that failover etc, I will, for this blog post, show failing over the entire group.

## Get your replication group

So the first step is what group? Well most likely you are working from the VM side first. So for a given VM, what replication group am I failing over?

This is fairly simple to figure out. If you have a vm object from get-vm, you can use **get-spbmreplicationgroup** to find the associated group.

So if I have a VM called vm-001 I will store it in \$vm:

```
[crayon-6515ae52ae728433162431/]
```

Then I can get the replication group from that:

```
[crayon-6515ae52ae732915359585/]
```

```
PS C:\Users\Administrator> $vm = get-vm vm-001
PS C:\Users\Administrator> $sourceGroup = $vm |Get-SpbmReplicationGroup
PS C:\Users\Administrator> $sourceGroup

Name                               ReplicationState
----                               -
sn1-x70-b05-33:vvol-1hr           Source
```

## Get target replication group

The next step is to get the target replication group, as this is what you are actually going to run the test failover against. Every source replication group has a target group on the replication side.

To find the target replication group you can use the **get-spbmreplicationpair** and pass in your source group.

[crayon-6515ae52ae734965551579/]

Note, this will not return the target group by itself, it will return the group pair, meaning the source group and the corresponding target group. You can store the target group by pulling it from the response.

[crayon-6515ae52ae735537362828/]

```
PS C:\Users\Administrator> $targetPair = get-spbmreplicationpair -source $sourceGroup
PS C:\Users\Administrator> $targetPair

Source Group          Target Group
-----
sn1-x70-b05-33:vwol-1hr  1037b35f-d0ef-40a5-9a43-ce0...

PS C:\Users\Administrator> $targetGroup = $targetPair.Target
PS C:\Users\Administrator> $targetGroup

Name                ReplicationState
-----
1037b35f-d0ef-40a5-9a4... Target
```

## Optional: Choose a Point-in-Time

The next step is optional. Do you want to failover to a specific point-in-time? If you do, you can query the available point-in-time's available. Otherwise, skip this step.

[crayon-6515ae52ae737572048720/]

```
PS C:\Users\Administrator> $targetPiTs = $targetGroup | Get-SpbmPointInTimeReplica
PS C:\Users\Administrator> $targetPiTs
```

Name	CreationTime	ReplicationGroup
	12/7/2018 3:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 2:09:30 PM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 1:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 12:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 11:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 10:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 9:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 8:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 7:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 6:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 5:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 4:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 3:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 2:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 1:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/7/2018 12:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 11:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 10:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 9:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 8:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 7:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 6:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 5:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 4:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 2:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 8:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/6/2018 2:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/5/2018 8:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/5/2018 2:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/5/2018 8:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/5/2018 2:01:00 AM	1037b35f-d0ef-40a5-9a4...
	12/4/2018 8:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/4/2018 2:01:00 PM	1037b35f-d0ef-40a5-9a4...
	12/4/2018 7:38:30 AM	1037b35f-d0ef-40a5-9a4...

You can choose the desired PiT by indexing to the right one. So If I want the second to the last, I can just choose \$targetPiTs[1]

```
PS C:\Users\Administrator> $targetPiTs[1]
```

Name	CreationTime	ReplicationGroup
	12/7/2018 2:09:30 PM	1037b35f-d0ef-40a5-9a4...

## Building the VVol Map

The last part is building what is called the VVol map. This indicates what VMs and what disks belong together. Every VVol has a UUID, and those UUIDs are stored in the VMX file and the corresponding VMDK files in the config VVol. These references need to be updated to the UUIDs for the VVols that are created from the PiT on the target site. The config VVol, which holds all of these files, is identically replicated, so the VMDK pointers will have references to the old UUIDs of the data VVols on the source side. The recovered data VVols will have new UUIDs. The test failover cmdlet will automatically update the files, but you do need to specify the source IDs.

This needs two things, the source VVol UUIDs and the source VVol datastore container ID.

The VVol datastore container ID is simple enough:

```
[crayon-6515ae52ae739638952581/]
```

Getting the source VVol IDs is fairly simple too. From your source replication group:

[crayon-6515ae52ae73f572534027/]

Then put them together:

[crayon-6515ae52ae741473474405/]

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> $containerId = $datastore.ExtensionData.Info.VVolDS.ScId
PS C:\Users\Administrator> $VVolIdList = $sourceGroup.extensionData.memberInfo.deviceId.id
PS C:\Users\Administrator> $sourceVVolMap = @{}
PS C:\Users\Administrator> foreach ($VVolId in $VVolIdList)
>> {
>>     $sourceVVolMap.Add($VVolId, $containerId)
>> }
>>
PS C:\Users\Administrator> $SourceVVolMap
```

Name	Value
rfc4122.103faa03-313f-4e33-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.597c1494-805f-4c93-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.cb73fc35-c8c3-4264-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.1a3d5d23-bb58-4f18-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.a583d0ec-b38e-4f00-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.7b213d39-8f78-4f56-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.14bcd0e8-a4a7-41b2-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.c300a0cd-8427-455a-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.0598cf53-8bdd-4bd7-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.54289d26-2ab2-48fa-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.3e2f9641-903e-4ab9-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.2ce94b75-4bff-4ba1-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.db6728e3-1db2-4ce1-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.c283e8b0-1f49-42ae-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.87596617-bd84-42b9-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.c300a9c8-7fa0-4b05-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.312633ab-0808-49aa-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.bc441d48-1aea-4d78-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.6aa303fa-be23-4a28-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.788e8931-20a4-4071-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.81c8889f-458b-4169-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.d4b3b43b-5d7b-4fa3-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.1af661f5-9887-4f85-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.b311fdfc-7bd6-422a-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.9027ba57-36f3-44ae-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.7390439e-0773-4f4d-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.578236e2-8e83-427f-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.1afa6a17-4809-4e3e-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.6af3aced-577c-45bd-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.650684b8-08a6-4c35-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.a9f0381a-55ef-4cb5-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.7dda29ae-a85b-4f5b-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.582d588a-8ec6-45df-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.1b3ce662-e9e0-4151-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.3ccb606d-0360-4a0f-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.0d53ffce-e113-4baa-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.af61c0d0-54c6-498a-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.b5a7015a-897d-4256-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.a0975342-7497-4e4c-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1
rfc4122.66c6059c-de05-4894-...	vvol:cc4341e24f9a3a06-b3ed05f8112a81a1

Now we are ready to run the test!

## Run Test Failover

[crayon-6515ae52ae742483324777/]

If you want to specify a PiT, just add your previously selected PiT to that command with:

[crayon-6515ae52ae744160744541/]

(whichever index of it you want)

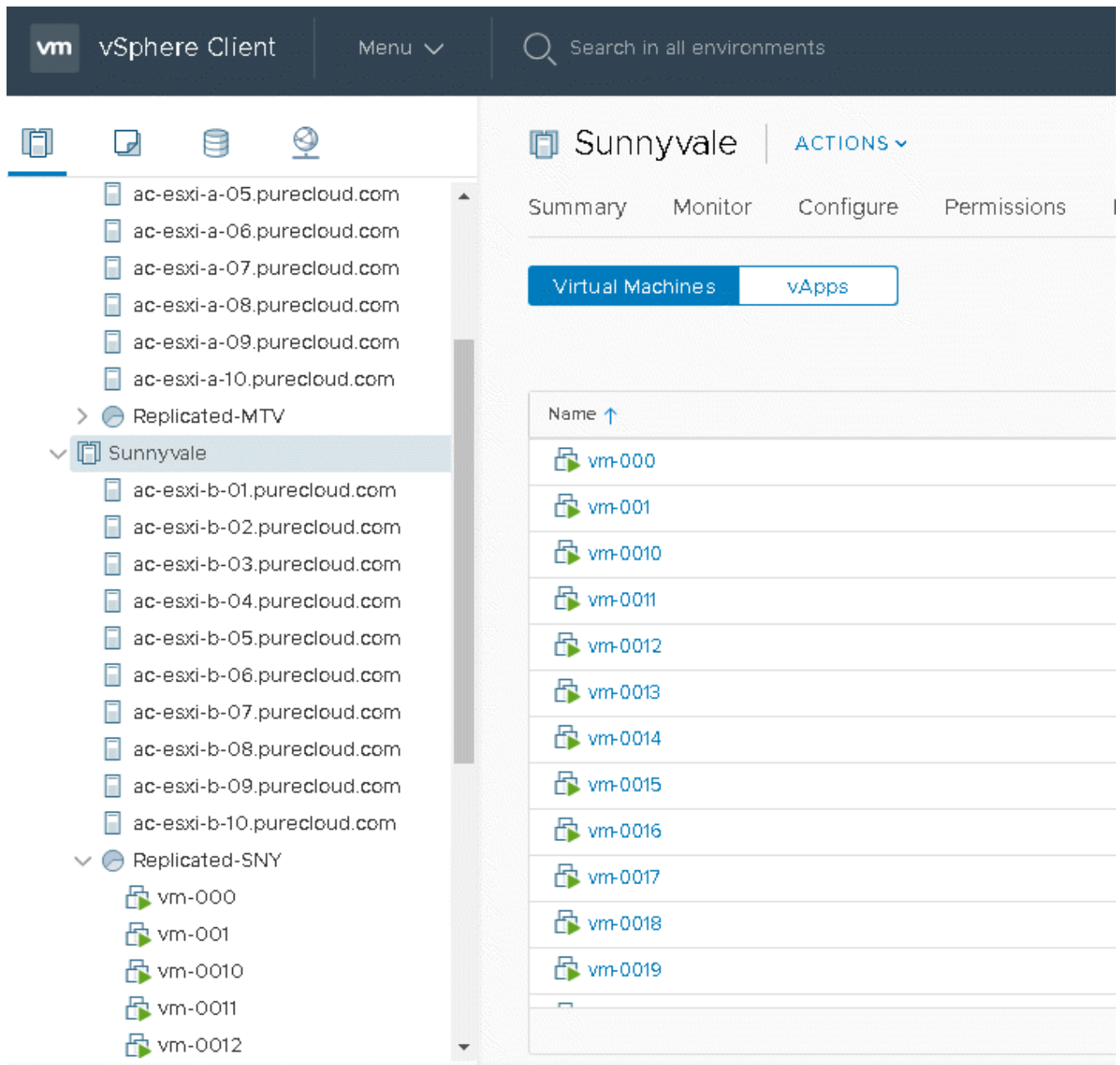
The failover on the array happens very quickly, the longest part is the VVol datastore file update process, which you will see in the task log:

Recent Tasks		Alarms		
Task Name ↑	Target	Status	Initiator	
Check new notifications	vcenter-pure.purecloud.c...	✓ Completed	VMware vS	
Datastore.updateVVolVirtual...	FlashArray-VVolDS-FA2	14%	PURECLOU	

The \$testVms object will contain the VMX path for all of the VMs you can now register and power on.

```
PS C:\Users\Administrator> $testVms = Start-SpbmReplicationTestFailover -ReplicationGroup $targetGroup -SourceVVolIdMap $sourceVVolMap
PS C:\Users\Administrator>
PS C:\Users\Administrator> $testVms
[FlashArray-VVolDS-FA2] rfc4122.bc0f5757-8c07-41ba-bd68-a22ee2cd80cd/vm-000.vmx
[FlashArray-VVolDS-FA2] rfc4122.ba22e9da-b475-48d5-9dae-174f7793ab29/vm-001.vmx
[FlashArray-VVolDS-FA2] rfc4122.99d7e42a-0908-42b5-8cd1-0cc3875a83ee/vm-002.vmx
[FlashArray-VVolDS-FA2] rfc4122.c85ac607-2ce7-49fb-b59e-4e42e0661eab/vm-003.vmx
[FlashArray-VVolDS-FA2] rfc4122.a0b1cf3e-819d-4508-a3aa-d3706bd4881c/vm-004.vmx
[FlashArray-VVolDS-FA2] rfc4122.c08a37ab-9267-4a3b-b334-ee76b6a7a016/vm-005.vmx
[FlashArray-VVolDS-FA2] rfc4122.a655afba-9f96-4abb-8f7b-16c65b85359b/vm-006.vmx
[FlashArray-VVolDS-FA2] rfc4122.61e0dabe-f6f2-413f-b1c5-1fc56aca4523/vm-007.vmx
[FlashArray-VVolDS-FA2] rfc4122.cb1b37fc-9a6f-4514-ae24-937390907eb9/vm-008.vmx
[FlashArray-VVolDS-FA2] rfc4122.d3b2dfed-52ce-431a-892d-42c2bf0a6bd3/vm-009.vmx
[FlashArray-VVolDS-FA2] rfc4122.1fd42bdc-be9f-4adb-8a59-b47b93809376/vm-0010.vmx
[FlashArray-VVolDS-FA2] rfc4122.f6612b47-327a-4f68-932d-a2d4d82a417e/vm-0012.vmx
[FlashArray-VVolDS-FA2] rfc4122.7de7b9bb-d75a-411e-87ea-fa792f6e68eb/vm-0011.vmx
[FlashArray-VVolDS-FA2] rfc4122.c9eec371-60a1-448d-a43c-92d2b5323ec3/vm-0013.vmx
[FlashArray-VVolDS-FA2] rfc4122.ace76bbc-c632-474a-a6d4-6227c4d6a6e0/vm-0014.vmx
[FlashArray-VVolDS-FA2] rfc4122.b6533388-2522-4413-95e6-6924bf45330e/vm-0015.vmx
[FlashArray-VVolDS-FA2] rfc4122.8e57f686-2fa7-4342-a1fe-6eaa419418da/vm-0016.vmx
[FlashArray-VVolDS-FA2] rfc4122.0d669db5-e38e-449b-960d-36ce703576c1/vm-0017.vmx
[FlashArray-VVolDS-FA2] rfc4122.05859b1a-32ff-40d4-ad3e-6d56ab4e099a/vm-0019.vmx
[FlashArray-VVolDS-FA2] rfc4122.182b7274-e683-4c4e-9e1e-042809b3ccaf/vm-0018.vmx
PS C:\Users\Administrator>
```

Then all you need to do is register the VMs and power them on:  
[crayon-6515ae52ae745314923284/]  
They are now test failed over:



## Clean up

The last step is to clean up. Just shut the VMs down and unregister them:


































```
[crayon-6515ae52ae747474192304/]
```

You do not need to delete the VMs, just unregister them. VASA will do that work for you.

Then tell the array to clean it up:

```
[crayon-6515ae52ae749434363473/]
```

We can see on the FlashArray that the VMs (volume groups) have been destroyed. They will automatically eradicate in 24 hrs, but you can of course destroy them earlier if you choose.

Volume Groups						1-1 of 1	<	>	+	⋮
Name ▲	# Volumes	Size	Volumes	Snapshots	Reduction					
 vvol--vSphere-HA-d0c2ed35-vg	1	4 G	7.29 K	-	42.8 to 1					 
Destroyed (20) ▲										
Destroyed Volume Groups						1-10 of 20	<	>	+	⋮
Name ▲		Volumes	Snapshots	Time Remaining						
 vvol-vm-000-bc0f5757-vg		6.40 K	-	23 h 59 m						 
 vvol-vm-001-ba22e9da-vg		16.26 K	-	23 h 59 m						 
 vvol-vm-0010-1fd42bdc-vg		5.77 K	-	23 h 59 m						 
 vvol-vm-0011-7de7b9bb-vg		6.91 K	-	23 h 59 m						 
 vvol-vm-0012-f6612b47-vg		3.99 K	-	23 h 59 m						 
 vvol-vm-0013-c9eec371-vg		0.00	-	23 h 59 m						 
 vvol-vm-0014-ace76bbc-vg		4.28 K	-	23 h 59 m						 
 vvol-vm-0015-b6533388-vg		0.00	-	23 h 59 m						 
 vvol-vm-0016-8e57f686-vg		6.01 K	-	23 h 59 m						 
 vvol-vm-0017-0d669db5-vg		20.69 K	-	23 h 59 m						 

And you're done!