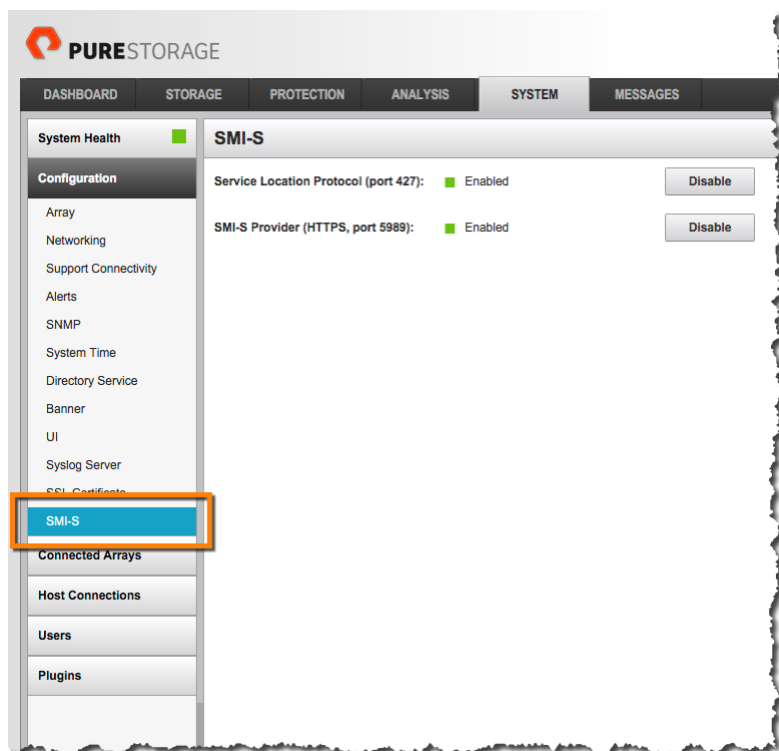


Pure Storage SMI-S Provider Overview

This specification was created by the Storage Networking Industry Association (SNIA) to provide a unified interface for multiple storage vendors within a network. With the introduction of SMI-S support, FlashArray administrators can manage the array via an SMI-S client over HTTPS. A great example of an SMI-S client is Windows Server 2012 R2, and we will be discussing this in more detail later. Functionality provided by the SMI-S provider supports in the areas of inventory, configuration, provisioning, capacity, and alerting. Keeping with our message of simplicity, the SMI-S provider is embedded within Purity vs using a Proxy which requires more complex management. An administrator has just two services to enable as shown below, these are disabled by default.

1. Service Location Protocol (port 427) — This protocol is used by SMI-S clients as a directory service to identify resources.
2. SMI-S Provider (HTTPS, port 5989) — FlashArray administrators use the SMI-S provider to manage the array through an SMI-S client.



The SMI-S provider is optional and must be enabled before its first use. The services can be enabled through the Web Management interface or via the Purity Command Line Interface (CLI) using the `puresmis` commands. Below is an example of using the `puresmis` command, this shows that the Service Location Protocol enabled and the SMI-S Provider enabled shown as WBEM-HTTPS.

```
barkz — root@csg-fa420-2-ct0: ~ — ssh pureeng@10.21.8.17 — 102x64
root@csg-fa420-2-ct0:~# puremis --h
usage: puremis [-h] {disable,enable,list} ...

positional arguments:
  {disable,enable,list}
    enable                enable SMI-S
    disable               disable SMI-S
    list                  list SMI-S status

optional arguments:
  -h, --help            show this help message and exit
root@csg-fa420-2-ct0:~# puremis list
Name      SLP      WBEM-HTTPS
csg-fa420-2  enabled  enabled
root@csg-fa420-2-ct0:~#
```

Now that the SMI-S services have been enabled let's take a look at how an SMI-S client can manage the Pure Storage FlashArray. The SMI-S client we will use is Windows Server 2012 R2. To enable Windows Server as an SMI-S client there is a specific feature called Windows Standards-Based Storage Management that needs to be installed using either the Server Manager or via Windows PowerShell with `Add-WindowsFeature WindowsStorageManagement`.

```
Administrator: Windows PowerShell
PS C:\> Get-WindowsFeature *WindowsStorage* | Format-Table -AutoSize

Display Name                                     Name                                     Install State
-----
[X] Windows Standards-Based Storage Management  WindowsStorageManagementService        Installed

PS C:\> _
```

Once this feature is added to the instance of Windows Server, using the SMI-S Provider is very straightforward. The first task is to register the SMI-S Provider. The PowerShell to register the provider is below. The `Get-Credential` will provide a prompt and in my case I am just using the standard pureuser account and password.

[crayon-6515c1367d2a5594038399/]

```
Administrator: Windows PowerShell
PS C:\> $Creds = Get-Credential
cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:
Credential
PS C:\> Register-SmisProvider -ConnectionUri https://10.21.8.17:5989 -Credential $Creds
PS C:\> _
```

Now that we have the SMI-S Provider registered there are a few cmdlets we can run to check that we have access to the FlashArray. Using `Get-StorageProvider` will show that the FlashArray registered is from the Manufacturer PureStorage and it is of Type SMI-S. The `Get-StorageSubSystem` will retrieve details about the FlashArray's HealthStatus, OperationalStatus and FriendlyName

[crayon-6515c1367d2af958677951/]

```
Administrator: Windows PowerShell
PS C:\> Get-StorageProvider

Type      Name                                     Manufacturer
----      -
SMP       Storage Spaces Management Provider      Microsoft Corporation
SMI-S     10.21.8.17                              PureStorage

PS C:\> Get-StorageSubSystem

FriendlyName          HealthStatus           OperationalStatus
-----
Storage Spaces on SCVMM1 Healthy                OK
csg-fa420-2           Healthy                OK

PS C:\>
```

Provider is registered and now we can update the cache of the PureStorage provider and associated child objects from the FlashArray. A very important detail to point out is that if you have multiple storage vendors within your environment and use the DiscoveryLevel = Full and do not set a specific vendor this will access all storage providers which can be an expensive operation.

[crayon-6515c1367d2b0141550511/]

```
Administrator: Windows PowerShell
PS C:\>
PS C:\> Update-StorageProviderCache -Name 10.21.8.17 -DiscoveryLevel Full -Manufacturer PureStorage
1/2 completed
[ooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo]
00:00:05 remaining.
Discovering provider objects
Running
[
```

Now there are other tasks that can be performed such as viewing the physical disks specific to a storage vendor. In the below example the display shows all of the drive bays and the details.

[crayon-6515c1367d2b2073178974/]

```

Administrator: Windows PowerShell
PS C:\> Get-PhysicalDisk -Manufacturer PureStorage | Format-Table -AutoSize

```

FriendlyName	CanPool	OperationalStatus	HealthStatus	Usage	Size
SH0.BAY0	False	OK	Healthy		2 GB
SH0.BAY1	False	OK	Healthy	238.23	GB
SH0.BAY10	False	OK	Healthy	238.23	GB
SH0.BAY11	False	OK	Healthy	238.23	GB
SH0.BAY12	False	OK	Healthy	238.23	GB
SH0.BAY13	False	OK	Healthy	238.23	GB
SH0.BAY14	False	OK	Healthy	238.23	GB
SH0.BAY15	False	OK	Healthy	238.23	GB
SH0.BAY16	False	OK	Healthy	238.23	GB
SH0.BAY17	False	OK	Healthy	238.23	GB
SH0.BAY18	False	OK	Healthy	238.23	GB
SH0.BAY19	False	OK	Healthy	238.23	GB
SH0.BAY2	False	OK	Healthy	238.23	GB
SH0.BAY20	False	OK	Healthy	238.23	GB
SH0.BAY21	False	OK	Healthy	238.23	GB
SH0.BAY22	False	OK	Healthy	238.23	GB
SH0.BAY23	False	OK	Healthy		2 GB
SH0.BAY3	False	OK	Healthy	238.23	GB
SH0.BAY4	False	OK	Healthy	238.23	GB
SH0.BAY5	False	OK	Healthy	238.23	GB
SH0.BAY6	False	OK	Healthy	238.23	GB
SH0.BAY7	False	OK	Healthy	238.23	GB
SH0.BAY8	False	OK	Healthy	238.23	GB
SH0.BAY9	False	OK	Healthy	238.23	GB
SH1.BAY0	False	OK	Healthy	476.45	GB
SH1.BAY1	False	OK	Healthy	476.45	GB
SH1.BAY10	False	OK	Healthy	476.45	GB
SH1.BAY11	False	OK	Healthy	476.45	GB
SH1.BAY12	False	OK	Healthy	476.45	GB
SH1.BAY13	False	OK	Healthy	476.45	GB
SH1.BAY14	False	OK	Healthy	476.45	GB
SH1.BAY15	False	OK	Healthy	476.45	GB
SH1.BAY16	False	OK	Healthy	476.45	GB
SH1.BAY17	False	OK	Healthy	476.45	GB
SH1.BAY18	False	OK	Healthy	476.45	GB
SH1.BAY19	False	OK	Healthy	476.45	GB
SH1.BAY2	False	OK	Healthy	476.45	GB
SH1.BAY20	False	OK	Healthy	476.45	GB
SH1.BAY21	False	OK	Healthy	476.45	GB
SH1.BAY22	False	OK	Healthy	476.45	GB
SH1.BAY23	False	OK	Healthy	476.45	GB
SH1.BAY3	False	OK	Healthy	476.45	GB
SH1.BAY4	False	OK	Healthy	476.45	GB
SH1.BAY5	False	OK	Healthy	476.45	GB
SH1.BAY6	False	OK	Healthy	476.45	GB
SH1.BAY7	False	OK	Healthy	476.45	GB
SH1.BAY8	False	OK	Healthy	476.45	GB
SH1.BAY9	False	OK	Healthy	476.45	GB

The last example to show is how to provision a new volume. The name of the volume is SMIS-VOL and will be 2 TB in size. Along with showing how to create a new volume the Web Management interface shows that the volume has been created and ready for use.

[crayon-6515c1367d2b3247367582/]

```

Administrator: Windows PowerShell
PS C:\> New-StorageSubsystemVirtualDisk -StorageSubSystemFriendlyName csg-fa420-2 -FriendlyName SMIS-VOL -Size 2TB

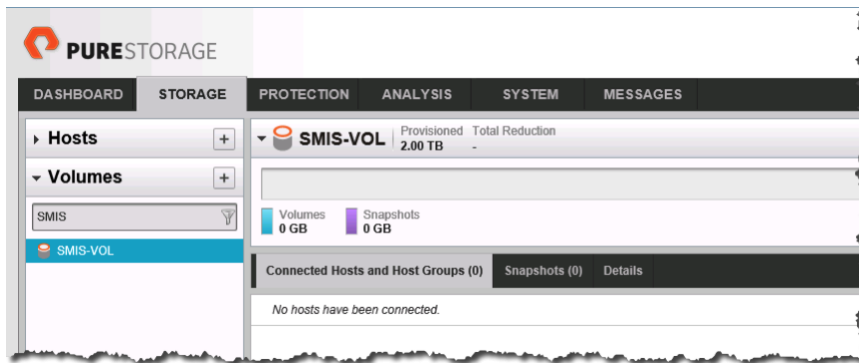
```

FriendlyName	ResiliencySettingName	OperationalStatus	HealthStatus	IsManualAttach	Size
SMIS-VOL		OK	Healthy		2 TB

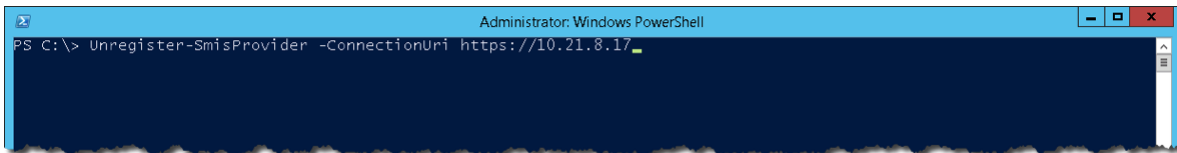
```

PS C:\>

```



The final step is to unregister the SMI-S Provider.
[crayon-6515c1367d2b4886122066/]



Below is a video demonstration showing all of the above operations.

The details discussed in this article are by no means an exhaustive list of operations that can be performed. With the multiple ways of managing our FlashArray combining the use of the SMI-S Provider and the [PowerShell](#) SDK offer a lot of different options for automation.

SMI-S References

- Storage Networking Industry Association (SNIA) — <https://www.snia.org/>
- Storage Management Initiative Specification (SMI-S) Releases — https://www.snia.org/tech_activities/standards/curr_standards/smi
- SMI-S Conformance Testing Program Official Results — <https://www.snia.org/ctp/conformingproviders/pure.html#SMIS1.6>
- Pure Storage SMI-S Provider Guide
- Storage Management Technical Specification, Part 3 Common Profiles Version 1.6.1, Revision 5 — https://www.snia.org/sites/default/files/SMI-Sv1.6.1r5_CommonProfiles.book_.pdf
- Storage Management Technical Specification, Part 4 Block Devices Version 1.6.1, Revision 5 — https://www.snia.org/sites/default/files/SMI-Sv1.6.1r5_Block.book_.pdf

Thanks,
Barkz