Pure Storage FlashArray Management Extension For Microsoft SQL Server Management Studio

purestorage.com

Contents

Chapter 1. Introduction

Overview

The Pure Storage FlashArray Management Extension for Microsoft SQL Server Management Studio (the FA SSMS Extension) is a host-based integration that executes on the Windows Server™ to manage SQL Server backup and restore operations. The Extension supports creating application-consistent snapshots using the Pure Storage Volume Shadow Copy Service (VSS) Hardware Provider. The Extension can be used from the FA SSMS GUI or automated using the Pure Storage Backup SDK Windows PowerShell module. Database administrators can manage backup and restore operations on local and remote databases from a centralized SSMS deployment.

The included Pure Storage Backup SDK provides automation of application-consistent SQL backup and restore operations, which can greatly help administrators learn and accomplish their tasks easily while avoiding common mistakes.

Microsoft® SQL Server™ Management Studio (SSMS) is an easy to use GUI provided by Microsoft as part of the SQL Server product. SSMS is based on the Visual Studio™ Shell, which can be extended using plugins.

The Pure Storage® FlashArray™ system is an ideal storage platform for performance intensive workloads. Pure's fast and highly available storage enables enterprise database applications to achieve top notch performance. Pure Storage strives for usage simplicity and provides management UIs and tools to help database administrators accomplish daily tasks easily.

Features

The Pure Storage FA SSMS Extension supports following backup and restore operations:

- Create and edit a backup configuration
- Perform backups based on named backup configurations
- Search and filter the backup history
- Remove a backup from the backup history
- Restore from a backup
- Mount or dismount a backup

Full and copy-only backups are supported:

- **Full backup** -- Backs up the entire database, including part of the transaction log so that the full database can be recovered after a full database backup is restored. A full database backup represents the database at the time the backup finished.
- **Copy-only backup** -- Provides the ability to use a database for scenarios such as development, test, and QA. A copy-only backup is independent of the sequence of conventional SQL Server backups.

Contrast this with a full backup, in which case the database is alerted and which affects how later backups are restored. This is not the case with copy-only backups.

Note: Administrators are responsible for ensuring point-in-time recovery when copy-only backups are used.

The following database restore use cases are supported:

- **Database restore** -- Import a backup (snapshot), then restore a database to the original location.
  This operation performs an overwrite of the original database volumes in place by restoring Pure Volumes that hold database or transaction log files. Other data, including other SQL data files located on those Pure Volumes, are also impacted by this point-in-time restore.
- **Database restore with No-Recovery** -- Import a backup (snapshot), then restore a database in No-Recovery Mode. The restored database is left in the *Recovering* state. This option provides support for point-in-time recovery. Additional transaction logs can be applied manually.
- **Database mount** -- Import a backup (snapshot), then restore a database to a different drive letter or mount point on the original or a different server, using the **`Mount`** option.
  The database files can be copied to the original location, or the database can be directly attached. If attached to the original server, be sure to make the **`Attach as`** name unique.

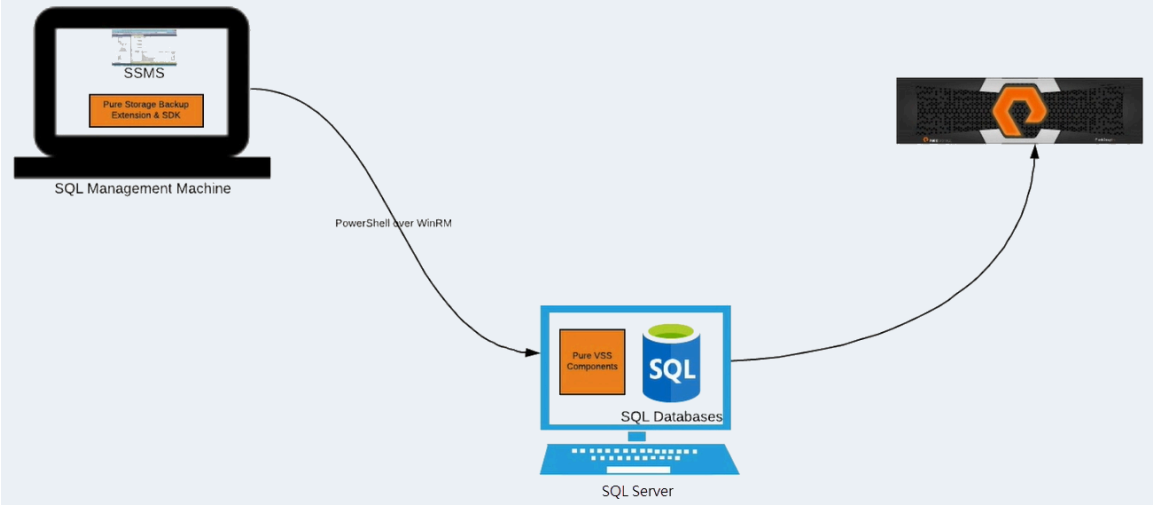
The Pure Storage FA SSMS Extension also supports following related features:

- Backups on running databases, without bringing the databases offline
- Automation through the Pure Storage PowerShell Backup SDK
- Application-consistent snapshots using the VSS Hardware Provider
- Microsoft SQL Server physical deployments
- VMware ESXi™ physical raw device maps (pRDMs)
- Scheduling administrative tasks using SQL Server Agent jobs with the **`CmdExec`** proxy option
- Managing databases running on the Pure Cloud Block Store™ virtual appliance
- A centralized installation to back up multiple databases (both local and remote)

The following diagram shows a simple scenario:

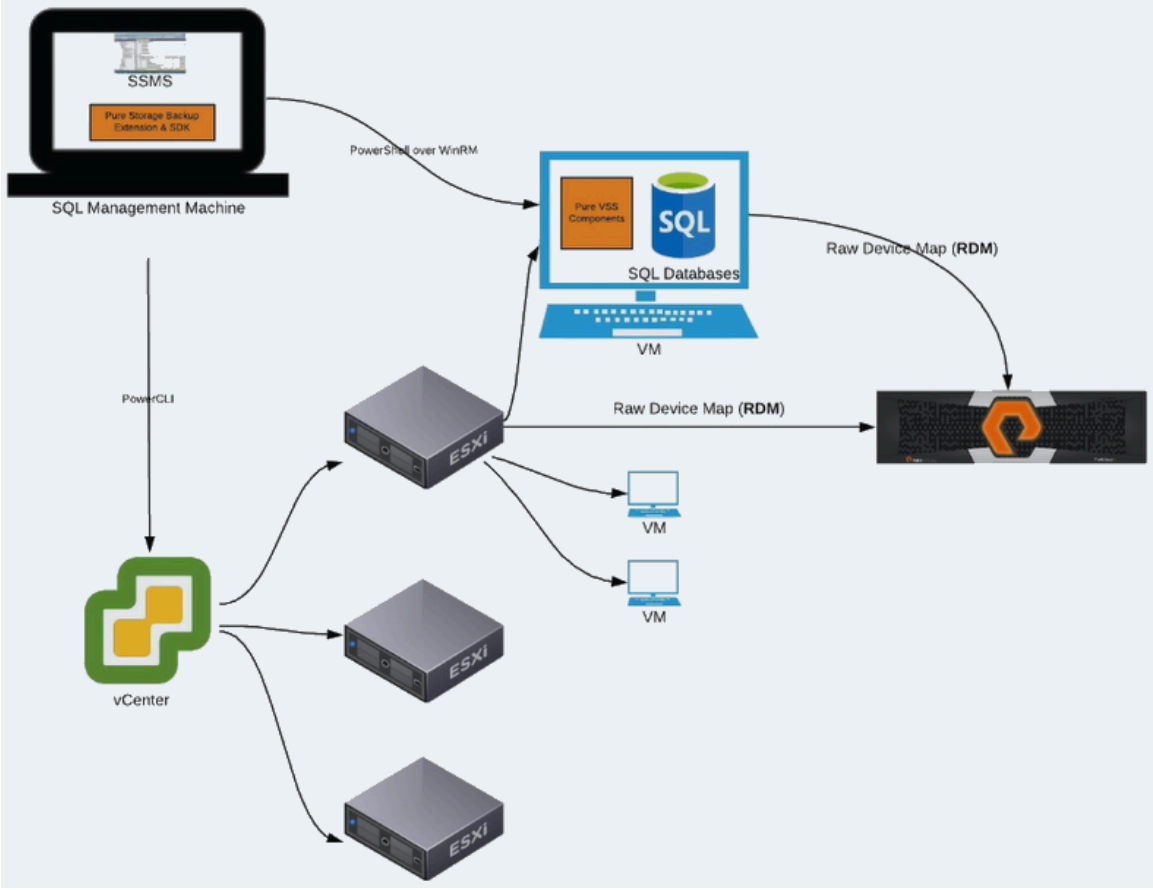
- The SQL database is hosted on the FlashArray.
- The Pure Storage FA SSMS Extension is installed on the SQL Management Machine with the Microsoft SSMS application.
- The Extension communicates with the SQL Server with PowerShell over WinRM.
- Multiple SQL Servers and multiple FlashArrays are also supported (not shown).

Figure 1.1:



The following diagram shows a scenario in which the SQL Server is hosted on an ESXi VM:

Figure 1.2:



Components

The following components are included with the FA SSMS Extension:

- The FA SSMS Extension for SQL Server Management Studio
- The Pure Storage Volume Shadow Copy Service (VSS) hardware provider
- The Pure Storage Backup SDK Windows PowerShell module for automation of all operations supported in the FA SSMS Extension

Named Backup Configurations

The FA SSMS Extension uses named backup configurations that define the databases to be backed up, along with other information such as credentials.

Each configuration has a name (that you provide). The name is like an alias for the backup job and allows the configuration to be reused conveniently.

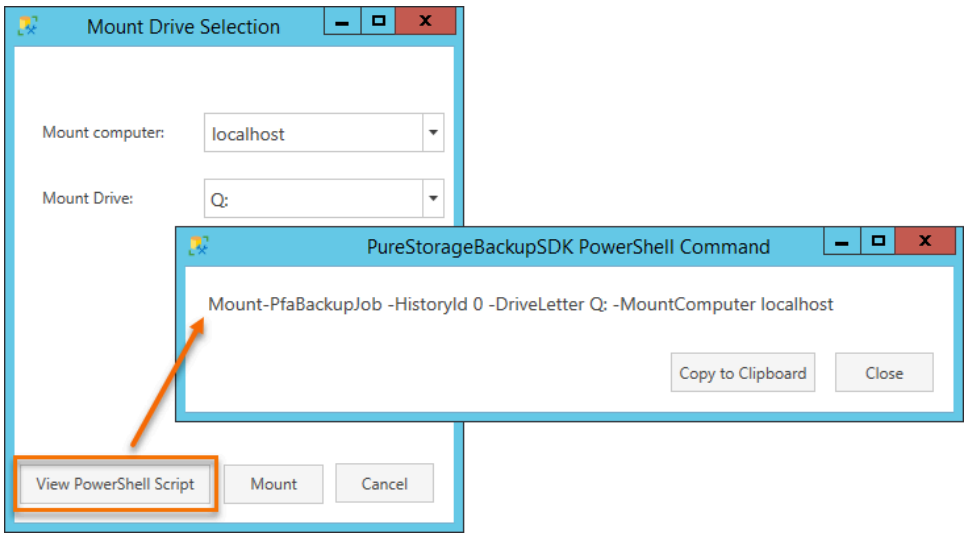
Automation

The FA SSMS Extension includes the Pure Storage Backup SDK, which offers scripting for all of the Extension's backup, restore, and history functions.

PowerShell scripts use the backup configuration name to automate your backup jobs.

As a convenience, FA SSMS Extension GUI screens provide a **`View PowerShell Script`** button that provides the equivalent PowerShell CLI script to the each GUI operation.

Figure 1.3:



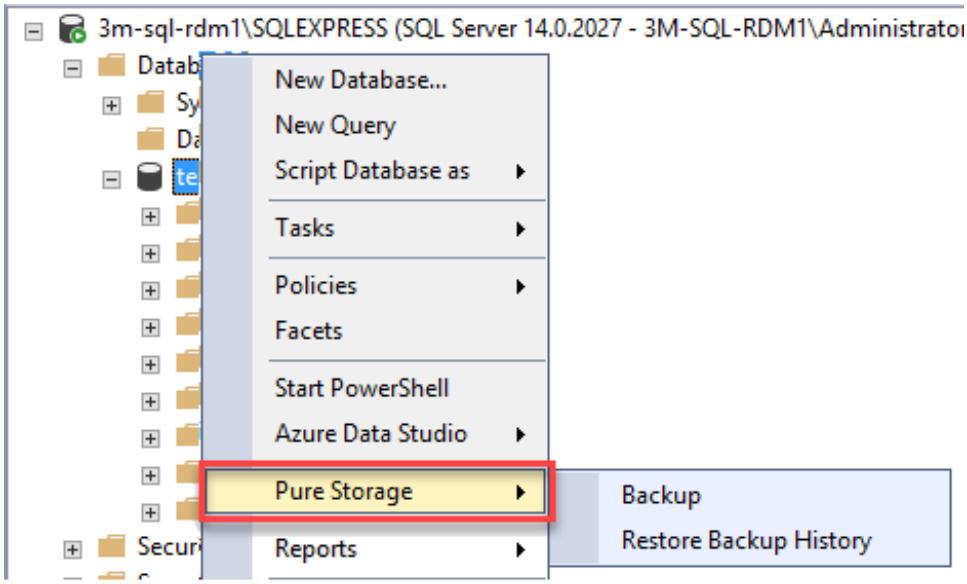
The FA SSMS Extension is also compatible with automation through SQL Server Agent jobs scheduling.

Access in the SQL Server Studio

The FA SSMS Extension appears in the Microsoft SQL Server Management Studio as an individual database context menu ("Pure Storage Snapshot Management") and also as a top level menu item ("Pure Storage").

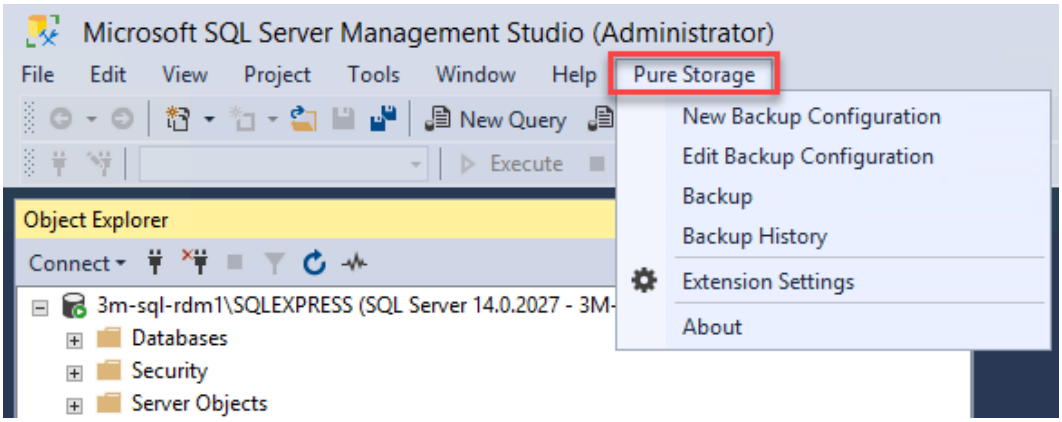
In a database context menu:

Figure 1.4:



In the SSMS top-level tool bar:

Figure 1.5:



Audience

This guide describes how to install, configure, and use the Pure Storage FlashArray SSMS Extension to provide application-consistent backup and restore of SQL databases with Microsoft SQL Server Management Studio and the Pure Storage VSS Hardware Provider.

The intended audiences for this guide include database administrators and SSMS administrators, who configure and support database backup and restore.

This guide assumes familiarity with storage, application-consistent snapshots, and Microsoft SSMS administration.

Chapter 2. Best Practices

Metadata Files

The FA SSMS Extension uses metadata to track backup information along with information about the database. Each backup run generates metadata that is required to successfully restore from the backup.

The Extension saves this metadata as XML files in a central location of your choosing so that the required information can be found during a restore. While the central location of the metadata files is configurable, after a backup has been made, it is extremely important to not move, edit, rename, or in any way modify a metadata file. To do so would make a restore of that backup impossible.

Important: Do *NOT* move, modify, or rename a backup's metadata file after a backup or snapshot has been created.

It is impossible to mount or restore a snapshot through the Extension if the metadata file is been modified or cannot be located. A manual volume copy (with or without overwrite) is required to access the data, provided that the snapshots still exist.

The default location for the Extension's metadata files is **`C:\Users\<username>\AppData\Roaming`**.

Back up Extension Metadata and Configuration Data

Important: We recommend that you regularly backup the following directories on the SQL management machine:

- The Extension's metadata files directory.
  The default location for the Extension's metadata files is **`C:\Users\<username>\AppData\Roaming`**. However, a different location could be specified during installation.
- The Extension's configuration data directory, **`%PROGRAMDATA%\PureStorage\PureStorageBackupSDK`**.

Configuration

Volume design principals

Critical information

- Consider assigning each database its own FlashArray volume. This practice makes restore operations safer, by being less likely to unintentionally overwrite other software.
- Consider assigning each database its own volume that is not also the operating system volume.
- Ensure that no database to be backed up is directly on the same volume as the operating system or the SQL Server.

Remote connectivity

With PowerShell remoting and Windows™ systems, ensure that the admin level of TrustedHosts is correctly set.

For more information, see *Windows PowerShell Remoting* and *Trusted Hosts* in the *Setup* chapter.

Databases as LUNs on the FlashArray

The databases hosted on Microsoft SQL Server can be on one or more disks. The recommendation from Pure Storage is that each of these disks be configured as a LUN (Logical Unit of storage) on the Pure Storage FlashArray.

For example, if a database named **`testDB`** contains **`testDB.mdf`** and **`testDB_log.ldf`**, both can reside on **`E:\`** and **`F:\`** drives respectively. Note that **`E:\`** and **`F:\`** drives correspond to two LUNs on on the Pure Storage FlashArray.

When the Pure Storage Backup SDK takes a backup of a database, that operation creates a Pure Storage snapshot for the set of LUNs that correspond to the input database.

For SQL Servers that are workstations

Both domain accounts and local user accounts are supported.

Users may have to set appropriate security policies to enable PowerShell remoting.

With SQL Server Availability Groups

The current release does not support seeding a database in order to join a database into an existing SQL Server Availability Group. Application consistent and crash consistent snapshots can be used for databases or volumes that participate in a SQL Server Availability Group, to be mounted as a standalone database outside of the SQL Server Availability Group.

As a workaround, add the database to the availability group after the database is restored.

With a Windows Server Failover Cluster

When deploying a database, the database should be isolated onto its own clustered disk and not be included with the SQL Server master, model, or msdb databases. Master, model, and msdb are typically installed on the root drive (C:\) but will be installed on the shared disk assigned to the SQL Server role for the SQL instance in a SQL Failover Cluster Instance. The new user database disk must be added to the cluster, added to the SQL role, and then made dependent of the SQL Server resource.

Runtime

Run SSMS only as administrator

- Run the SSMS application with administrator credentials for the machine the SSMS application is hosted on.

Mounting a backup in a failover cluster

- When performing a mount operation in a Microsoft failover cluster, mount the database node outside of the current failover cluster, in order to avoid a signature collision.

Restoring volumes that contain multiple databases

- When restoring a volume that contains multiple databases, do not perform an overwrite (an in-place restore).

  An in-place restore effectively reverts all databases on the volume back to the time of the backup.

  Following this practice makes restore operations safer and less likely to unintentionally overwrite other software.

Run management operations on the SSMS machine only

- Run the backup, restore, and mount management operations only from the machine where SSMS is deployed.

Avoid manual operations on the SQL Servers

- Manual operations on the SQL Servers are not recommended.

  Operations to be avoided on the SQL Servers include masking (removing a shadow copy or snapshot), importing a snapshot, mounting a snapshot, and dismounting a snapshot.

Limitations in this Release

Mounting on the Backup Machine

Mounting is only supported on the same physical SQL machine that was backed up. Mounting to a VM or to a different physical machine is not supported in this release.

SQL Servers Hosted on VMs

Only VMware ESXi virtual machines configured with the raw device mapping (RDM) storage access method are supported. On virtual SQL Server machines, mounting is supported only to virtual machines on the same hypervisor machine.

Manual Cleanup after Removing a Backup History Entry

In this release, when an entry is removed from the Backup History listing, the Extension does not delete either the backup snapshots and volumes on the FlashArray or associated items in the VSS history or cache.

Clean up of these associated items has do be done manually.

Chapter 3. Setup

Before You Begin

Before you install the Pure Storage FA SSMS Extension, verify that your system meets the minimum requirements.

Verify System Requirements and Compatibility

Verify that your system meets the minimum requirements to run the FA SSMS Extension:

Software	Minimum Version
Microsoft SQL Server (supported only on Windows Server or Windows 10)	2012
Microsoft SQL Server Management Studio (SSMS)	18.x
Purity	4.8
Purity REST API	1.7

The following sections describe additional requirements in certain environments.

For SQL Servers Hosted on VMs

The following are required when a SQL Server is hosted on a VM:

- VMware vCenter Server 6.0 or higher
- PowerCLI, on the SSMS host

For SQL Servers Hosted on Windows Server 2012

If a SQL Server is hosted on Windows Server 2012, PowerShell 5.0 must be installed on the SQL Server host. PowerShell 6.0 and 7.0 Preview are not supported.

Windows PowerShell Remoting

PowerShell remoting must be enabled on the SQL Server machine.

For steps to enable PowerShell remoting, please see Microsoft PowerShell documentation. At the time of this writing, the link is *Enable-PSRemoting* [https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/enable-psremoting?view=powershell-7].

The Extension installer includes the Pure Storage PowerShell Backup SDK and the Pure Storage VSS Hardware Provider. When Windows PowerShell remoting is enabled and running on the SSMS machine, the Extension automatically installs the Pure Storage PowerShell Backup SDK and the Pure Storage VSS Hardware Provider on SQL Server machines when required.

Windows PowerShell remoting is a prerequisite for this feature.

See also the Microsoft article *Installation and Configuration for Windows Remote Management* [https://docs.microsoft.com/en-us/windows/win32/winrm/installation-and-configuration-for-windows-remote-management].

Trusted Hosts

If the environment includes hosts for the SSMS application or the SQL Servers that are not in the same Active Directory domain, manually add those hosts that are not within the SSMS application host's domain to the TrustedHosts list on the SSMS application host.

Example command to add a SQL Server to the TrustedHosts list:

```
Set-Item WSMan:\localhost\Client\TrustedHosts -Value <SQL Server host address>
```

The SQL Server host's fully-qualified domain name (FQDN) can be used in place of the address.

You may need to restart the WinRM service for the new settings to take effect.

```
Restart-Service WinRM
```

See *TroubleShooting* for an example of an error message seen due to TrustedHost issues.

Download and Install the Microsoft SSMS Application

If you have not already done so, download and install Microsoft's stand-alone SSMS application before installing the Pure Storage FA SSMS Extension.

Download the Microsoft SSMS 18.x or later installer from one of the following links:

- The Microsoft SSMS download page is ***Download SQL Server Management Studio*** [https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15].
- This link begins the download of the SSMS installer executable directly (no landing page): ***go.microsoft.com*** [https://go.microsoft.com/fwlink/\?linkid=2108895&clcid\=0x409].

Links are subject to change. Follow Microsoft SSMS documentation for installation instructions.

Download and Install the Pure Storage FA SSMS Extension

Install the Extension on the Microsoft SQL Server Management Studio machine. The FA SSMS Extension needs to be installed only once (and that is on the SSMS application host machine).

The installer is hosted on both github.com [https://github.com] and Pure1 Support [https://support.purestorage.com]. Please see the release notes for the download locations.

Run the FA SSMS Extension installer, **`PureSSMSInstaller.msi`**, and follow the prompts.

Note: If you do not select the default location for the Extension's metadata files, make a note of the new location in order to back up the metadata files regularly. See *Back up Extension Metadata and Configuration Data*.

Chapter 4. Using the FA SSMS Extension GUI

Overview of Using the Pure Storage FA SSMS Extension

The first step is to create a backup configuration, which is the definition of what a backup should do. (See *Create a New Backup Configuration*.)

You can create multiple backup configurations, which can be used to run a backup on demand and can be automated through the Pure Storage PowerShell Backup SDK.

Open the Pure Storage FA SSMS Extension Menu

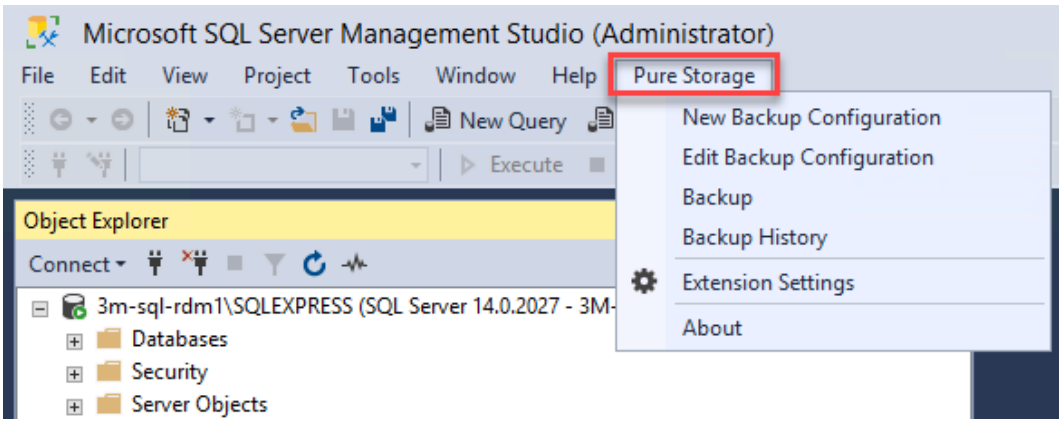
There are two ways to open the FA SSMS Extension menu, from the SSMS tool bar or from a database selected in the Object Explorer.

The Pure Storage tab in the SSMS tool bar provides the full FA SSMS Extension menu, which applies to SQL databases that are backed up by the Extension. The Extension menu opened from a specific database offers actions only on that database and its backups.

From the SSMS Tool Bar

To open the Extension menu from the SSMS tool bar, select the **Pure Storage** tab:

Figure 4.1:



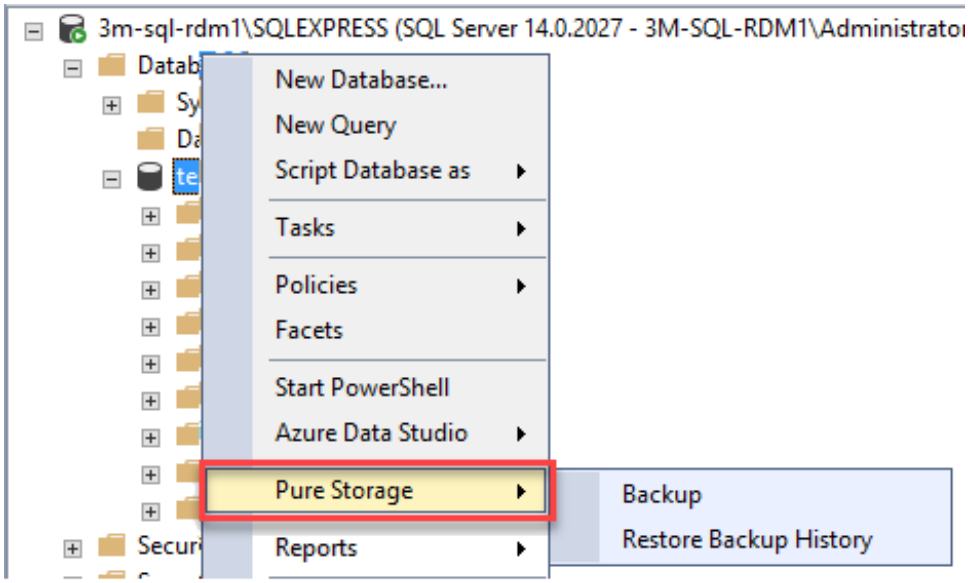
From the Context Menu for a Database

Note: the first time you open a database's context menu, if the Extension is still initializing, the Pure Storage menu does not appear. If that happens, please close the menu and reopen it.

To open the FA SSMS Extension menu from a specific database:

1. In the Object Explorer, select the database.
2. Right click the database and select an action for that specific database.

Figure 4.2:



Create a New Backup Configuration

Creating a backup configuration is the first step to protecting your database.

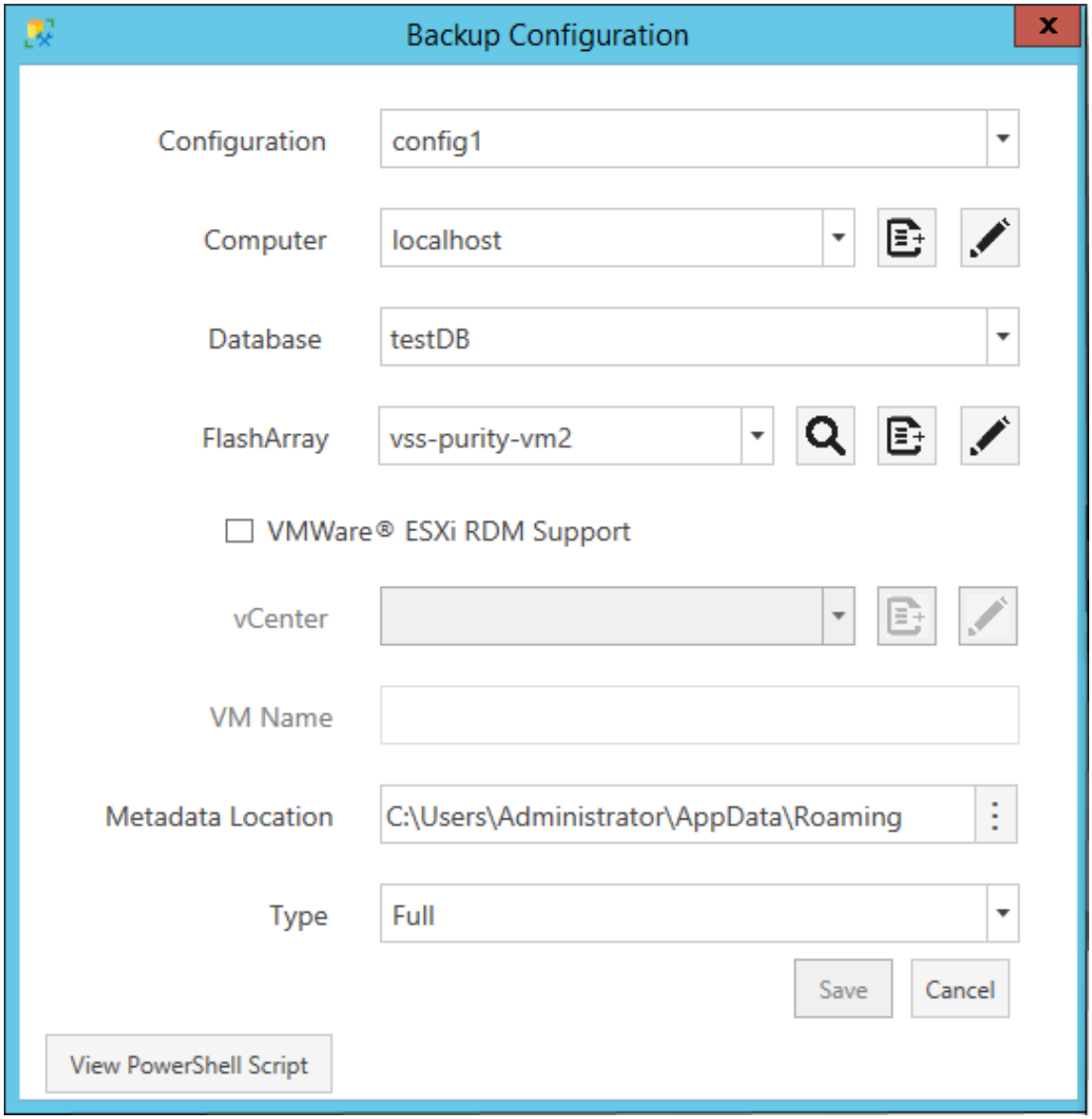
A backup configuration is required before a backup can be performed. A backup configuration includes the information needed to perform the backup, including database name, whether the database is on a physical machine or on a virtual machine hosted on ESXi and configured in VMware raw device mapping (RDM) mode, whether the backup is full or copy only, and various credentials.

Backup configurations are named, which allows automating backups and reusing configurations by referencing the configuration name.

Note: PowerCLI must be installed on the Microsoft SQL Server Management Studio host before you can create a backup configuration for a SQL Server that is hosted on a VMware ESXi VM.

The following is an example of a completed Backup Configuration dialog.

Figure 4.3:



Note: Enter the Computer, Computer Credential, Database, FlashArray, and FlashArray Credential fields in exactly the order shown.

After you specify the SQL Server ("Computer"), the Extension discovers which databases are hosted on that server and presents them in the database dropdown menu. Similarly, after the database is selected, the Extension can discover which FlashArray provides storage for that database.

These instructions describe the steps to create the first backup configuration. When subsequent configurations are created, host names and FlashArray names are available in the dropdown menus and credentials do not have to be re-entered.

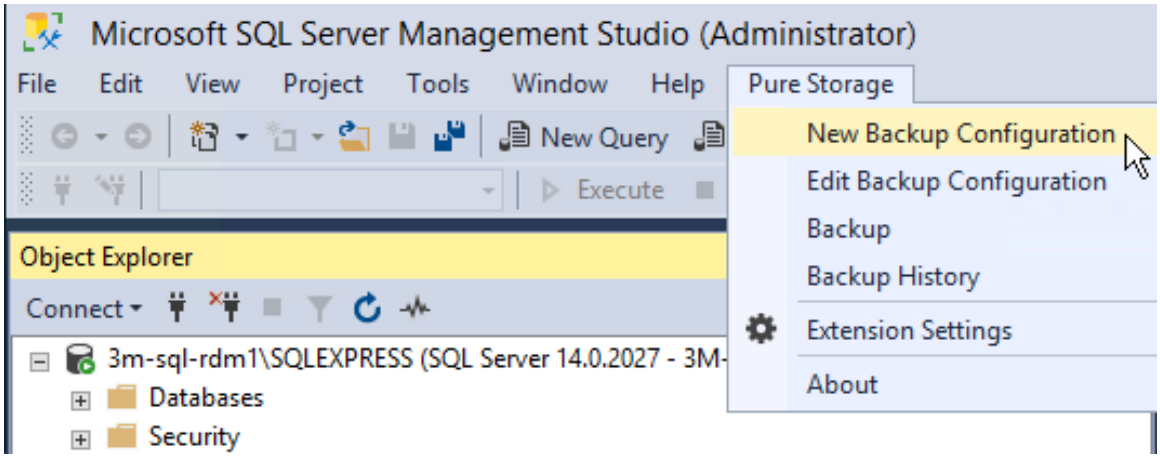
Note about the icons on the Backup Configuration dialog:

- The New Credential icon is used to specify a new SQL Server, FlashArray, or vCenter Server, as well as administrator credentials for that server or array.
- The Edit icon is used to change an administrator's password or to supply a different administrator account. The Edit icon is used when editing an existing backup configuration, not when creating a new backup configuration.
- The optional Discover icon identifies the specific FlashArray that hosts the SQL database in the backup configuration and puts that array name in the FlashArray field. Note that discovery may take a few moments.

To create a new backup configuration:

1. In the SSMS tool bar, select **Pure Storage > New Backup Configuration**.

   Figure 4.4:

   The Backup Configuration screen opens.

2. In the **Configuration** field, enter a descriptive name for the new configuration.
   On-demand backups in the GUI and backup automation scripts will reference this name.
3. In the **Computer** row, specify the SQL Server that hosts the database to be backed up:

   a. Click the **Create a new computer credential** icon.

   b. The New Computer Credential dialog opens.
   Enter the following information:

   Computer Label
   Enter the name of the machine (or VM) hosting the SQL Server.
   This name can be the actual machine name or a nickname used only within the Extension.

Address
Enter the fully-qualified domain name or IP address of the machine hosting the SQL Server.
For VMs on ESXi using SQL Failover Cluster Instance, this name must to be the SQL Server Network Name. Otherwise, the backup will fail if the role is running on another VM in the cluster.

Use credential for current machine
Enable this check box if the Extension is running on the SQL Server host.
In this case, the Username and Password fields below are grayed out.

Username
Enter an administrator for the machine hosting the SQL Server.

Password
Enter that administrator's password.

When you save the credentials dialog, the Extension confirms that it can reach the specified SQL Server host with those credentials.

The Extension also discovers the server's SQL databases that are hosted on FlashArrays.

4. Back on the Backup Configuration dialog, select the SQL Server in the Computer dropdown menu.
5. In the **Database** row, select the database to be backed up.
6. In the **FlashArray** row, specify the FlashArray hosting the database:
   a. Click the **Create a new FlashArray credential** icon.
   b. The New FlashArray Credential dialog opens.
   Enter the following information:

   FlashArray Label
   Enter the name of the FlashArray hosting the SQL database.
   This name can be the actual machine name or a nickname used only within the Extension.

   Address
   Enter the fully-qualified domain name or IP address of the FlashArray hosting the SQL database.

   Username
   Enter a FlashArray administrator, whose credentials will be used to connect to the array. Array admin privileges are required.

   Password
   Enter that administrator's password.

7. Back on the Backup Configuration dialog, select that FlashArray in the FlashArray dropdown menu.
8. Select the RDM checkbox if the computer is a virtual machine hosted on ESXi and configured in VMware raw device mapping (RDM) mode for datastore.
   If that is the case, specify the vCenter in the **vCenter** row:
   a. Click the **Create a new vCenter credential**  icon.
   b. The New vCenter Credential dialog opens.
      Enter the following information:

      vCenter Label
      Enter the name of the vCenter.
      This name can be the actual vCenter name or a nickname used only within the Extension.

      Address
      Enter the fully-qualified domain name or IP address of the vCenter.

      Username
      Enter an administrator for *the machine hosting the vCenter*.

      Password
      Enter that adminstrator's password.
9. Back on the Backup Configuration dialog, select that vCenter in the vCenter dropdown menu.
10. If the SQL Server is hosted on a VM, enter its VM Name as configured in vCenter.
11. The **Metadata Location** field shows the location of the backup metadata files.
    Do not change the **`Metadata Location`** field if any backup has already been made. To do so would make the Extension unable to restore those backups.
    Please see the warnings in *Metadata Files*.
12. In the **Type** field, select the backup type, either **`Full`** or **`Copy only`**.
    See *Microsoft SQL Server documentation* [https://docs.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-overview-sql-server] for a description of backup types.
13. Click **Save**.
    Your backup configuration is available to use.

Edit an Existing Backup Configuration

To change an administrator account or password:

1. In the SSMS tool bar, select **Pure Storage > Edit Backup Configuration**.
   The Backup Configuration screen opens.
2. To change an administrator or administrator's password, click the Edit icon in the row for that system.
   In the popup, enter the new account information.
3. When you have made your changes, click **Save**.

To edit an existing backup configuration:

1. In the SSMS tool bar, select **Pure Storage > Edit Backup Configuration**.
2. The Backup Configuration screen opens.
   If necessary, use the **Configuration** dropdown menu to select the correct backup configuration, and the **Database** dropdown menu to select the correct database.
3. Make the necessary changes.
   However, do not change the **`Metadata Location`** field if any backup has already been made. To do so would make the Extension unable to restore those backups.

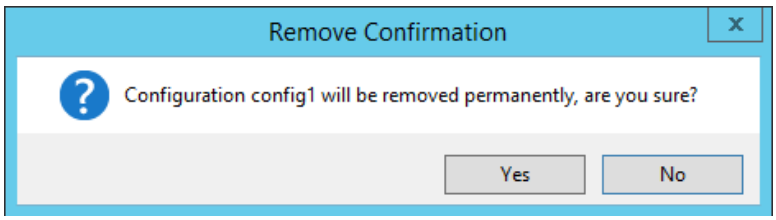
   Please see the warnings in *Metadata Files*.
4. When you have made your changes, click **Save**.

Delete a Backup Configuration

To delete an existing backup configuration:

1. In the SSMS tool bar, select **Pure Storage > Edit a Backup Configuration**.
2. In the Edit a Backup Configuration screen, use the Configuration drop-down menu to select the configuration that is to be deleted.
   If you are sure the correct configuration is selected, click **Remove**.
3. In the confirmation dialog, click **Pure Storage > Edit a Backup Configuration**.

   Figure 4.5:



   If you are sure the correct configuration is selected, click **Yes**.

Backup Using an Existing Configuration

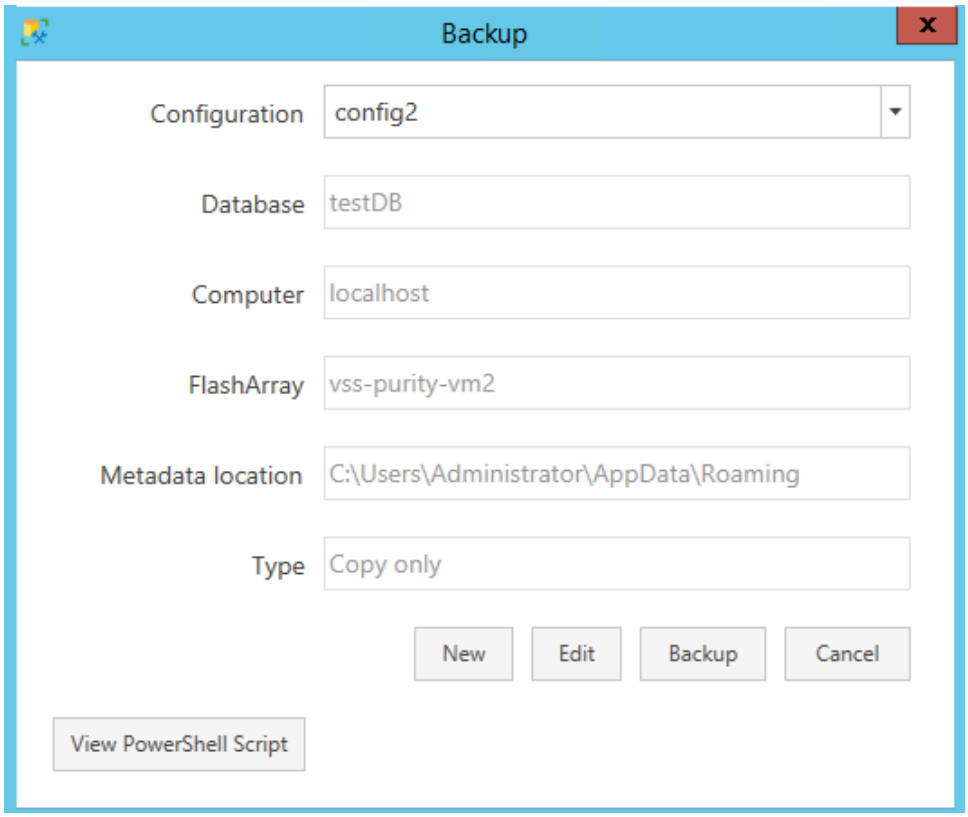
This section describes how to run an on-demand backup.

These steps require that an appropriate named backup configuration has already been created (see *Create a New Backup Configuration*).

To perform a backup using an existing configuration:

1. In the SSMS tool bar, select **Pure Storage > Backup**, or in the Object Explorer, select the database to be backed up, right click, and select **Pure Storage > Backup**.
2. The Backup dialog opens.
   If necessary, use the **Configuration** dropdown menu to select the correct backup configuration.

   Figure 4.6:



3. Confirm that the correct configuration is selected in the Configuration menu.
   Then click **Backup** to start the backup.

Use the **New** or **Edit** buttons to create a new backup configuration or to edit the selected backup configuration, if necessary.

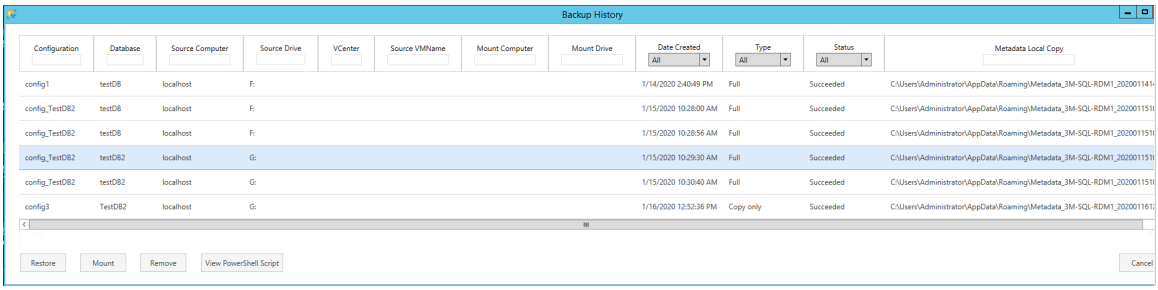
View the Backup History

The backup history lists backups that are available to restore, mount, dismount, or remove.

To view the backup history:

1. In the SSMS tool bar, select **Pure Storage > Backup History**, or in the Object Explorer, select the database to be backed up, right click, and select **Pure Storage > Backup History**.
2. The Backup History listing opens.

Figure 4.7:



Notes:

- To search or filter the history, enter a keyword in any of the filter boxes (Configuration, Database, etc.).
- To sort the listing, click a column header.
- Select a specific backup entry to enable the Restore, Mount, and Remove buttons.
  Select a mounted backup entry to enable the Dismount button.
- The View PowerShell Script button displays the command to remove the selected backup.
  To view the Restore or Mount commands, select that button and complete the dialog, then select View PowerShell Script in Restore or Mount dialog.

See also *Dismount a Backup*, *Mount a Backup*, *Remove a Backup History Entry*, and *Restore a Backup*.

Restore a Backup

Caution: Be very careful when restoring, that the correct backup is selected. Restoring from an incorrect backup will overwrite data.

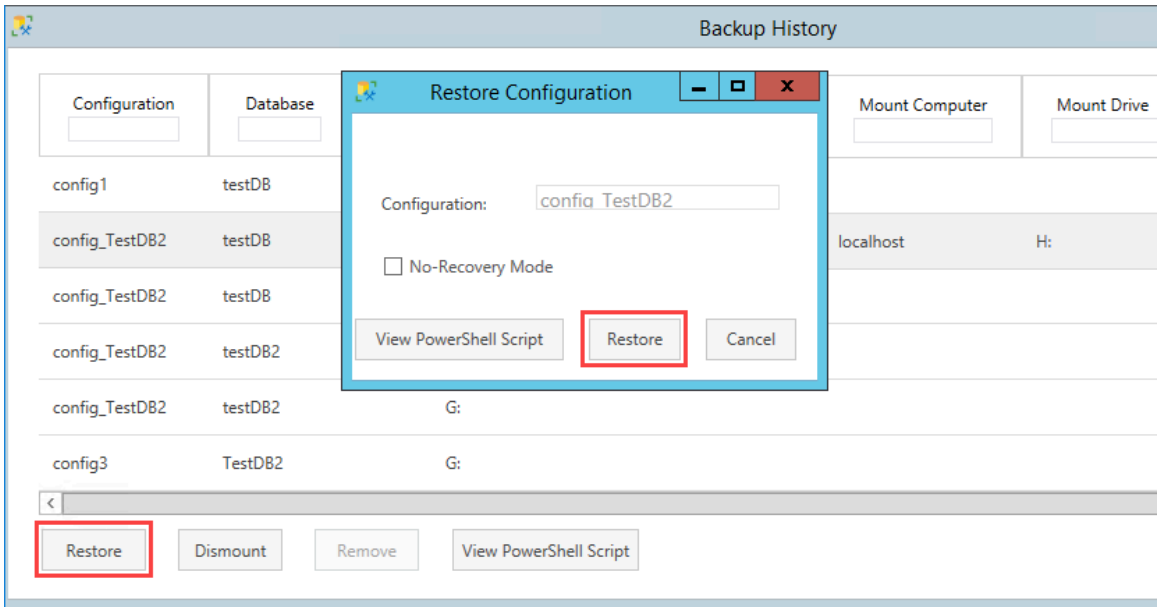
Note: When restoring a SQL Failover Cluster Instance VM, the SQL Role must be online on the VM that is the source computer in the backup history. Otherwise the restore will fail.

To restore from a backup:

1. In the SSMS tool bar, select **Pure Storage > Backup History**, or in the Object Explorer, select the database to be backed up, right click, and select **Pure Storage > Backup History**.
2. If required, search or filter the list to locate the specific backup.
   Select the backup and click **Restore**.

   The Restore Configuration dialog opens.

   Figure 4.8:



3. In the Restore Configuration dialog:
   a. Optionally enable the **`No Recovery`** check box.
      Please see Microsoft SQL Server documentation for the effect of the **`No Recovery`** option.
   b. To view the equivalent PowerShell command, first complete the Restore Configuration dialog fields, then click **View PowerShell Script** (optional).
   c. Double-check that the correct backup is selected.
      If you are sure, click **Restore**.

Mount a Backup

The mount operation imports a backup to the SQL Server machine, where a database administrator can inspect it.

For physical SQL Server machines, mounting is only supported on the same SQL machine that was backed up. For virtual SQL Server machines, mounting is supported only to virtual machines on the same hypervisor machine.

Notes on SQL Server Failover Cluster Instances:

- A Failover Cluster Instances cannot mount a snapshot with the same disk signature. Such a mount attempt fails.
- As a workaround, mount the snapshot to a server or VM outside of the cluster, then change the disk signature.
- After the disk signature is changed, the snapshot can be manually mounted to the host group, or in the case of VMs, the snapshot can be added as RDM/Existing disk.
- See *Using FlashProtect Snapshots and Cluster Shared Volumes* [https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/Failover_Clustering/Using_FlashProtect_Snapshots_and_Cluster_Shared_Volumes] for more information.

To mount a backup:

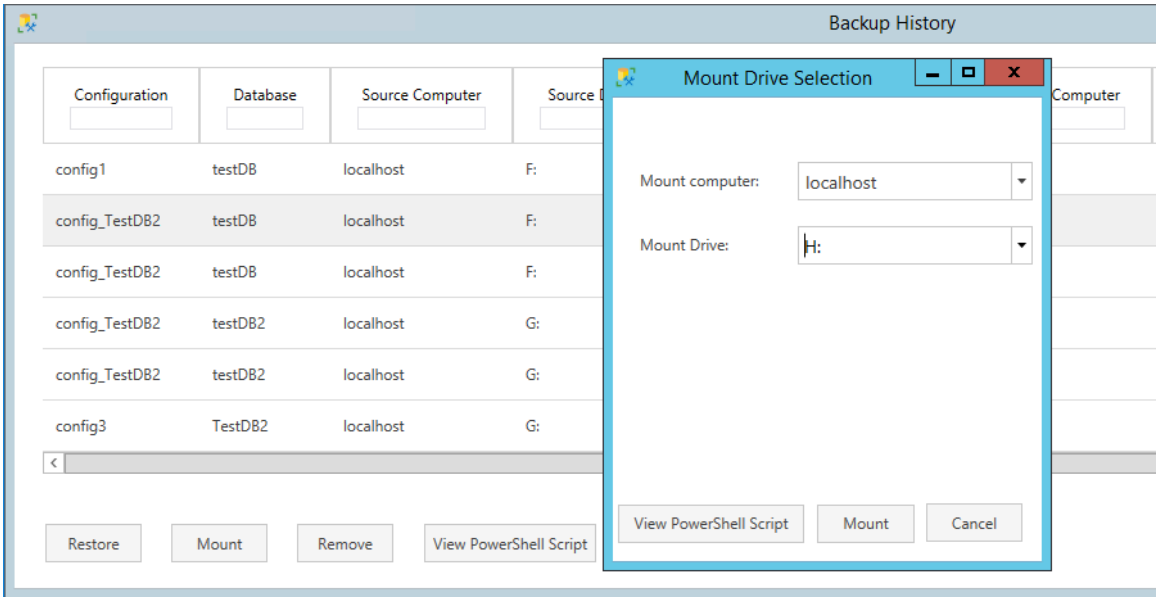
1. In the SSMS tool bar, select **Pure Storage > Backup History**, or in the Object Explorer, select the database to be backed up, right click, and select **Pure Storage > Backup History**.
   If necessary, search or filter the list to locate the specific backup.
2. Select the backup and click **Mount**.
3. Enter the following information in the Mount Drive Selection dialog:

   Mount Computer
   A friendly name of mount destination computer
   Accept the suggestion in the **Mount computer** dropdown menu.

   Mount Directory
   Path to the mounted database backup on the destination computer
   The path can be a drive name (such as D:) or a mount path (such as D:\Database).
   Multiple paths are supported, for a database that resides on multiple source drives.

   VM Name
   For a SQL Server that is running on an ESXi host managed by vCenter, enter the VM Name of that computer in vCenter.
   (The VM Name field is not shown in the example image.)

Figure 4.9:

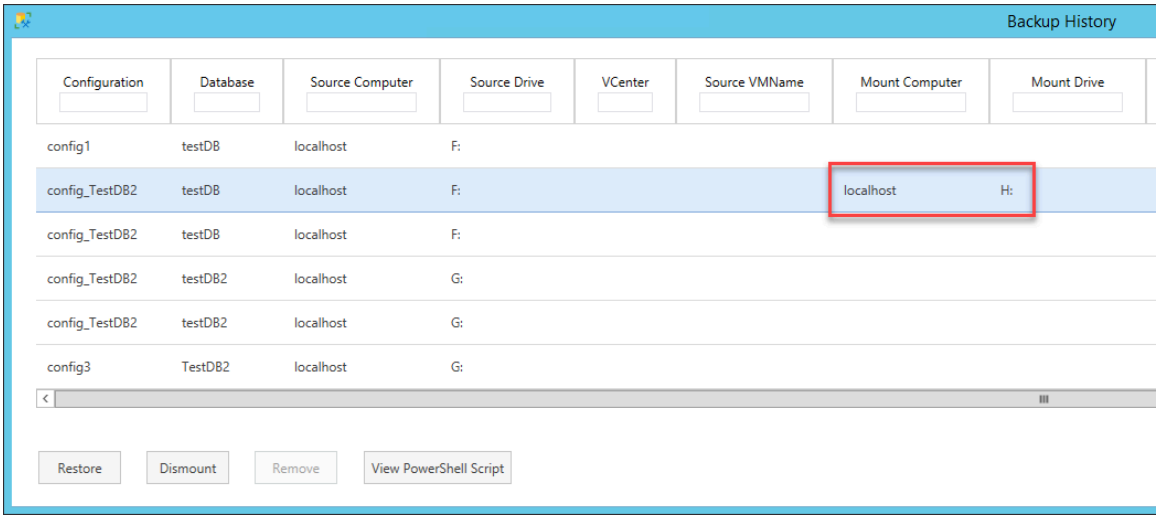


4. To view the equivalent PowerShell command, first complete the Mount Configuration dialog fields, then click **View PowerShell Script** (optional).

5. Double-check that the correct backup is selected.
   If you are sure, click **Mount**.

   On success, a completed message is shown and the mount is shown in the Backup History listing, with entries in the **`Mount Computer`** and **`Mount Drive`** columns.

Figure 4.10:



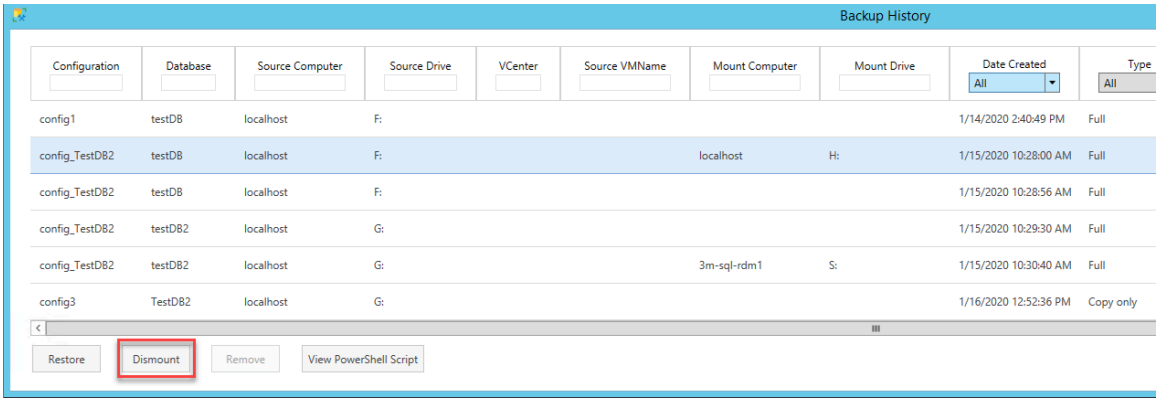
Dismount a Backup

Note: Dismounting a backup completely removes the mount drive and all its content. Any volume copies or other changes made on mounted backups are not persisted once a dismount is performed.

To dismount a backup, specify an existing backup history that is already mounted to a mount destination. The **Dismount** button is only available when a mounted backup is selected.

1. In the SSMS tool bar, select **Pure Storage > Backup History**, or in the Object Explorer, select the database to be backed up, right click, and select **Pure Storage > Backup History**.
   Search or filter the list to locate the specific backup.
2. Select the correct backup.

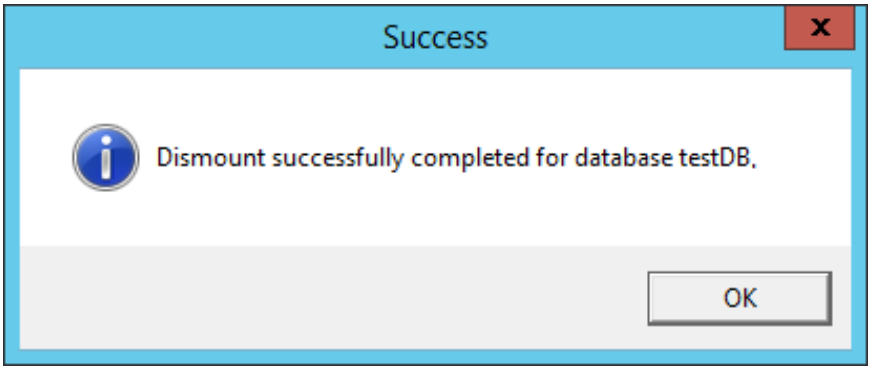
   Figure 4.11:



   If you are sure the correct backup is selected, click **Dismount**.

   The dismount begins immediately, without a confirmation question.
3. A success message is shown and the Mount Computer and Mount Drive information is removed from the Backup History listing.

   Figure 4.12:



Remove a Backup History Entry

This section describes removing an entry from the Backup History listing.

This action does not remove the backup itself, which should be removed manually. See also *Limitations in this Release*.

When a backup is mounted, its entry cannot be removed.

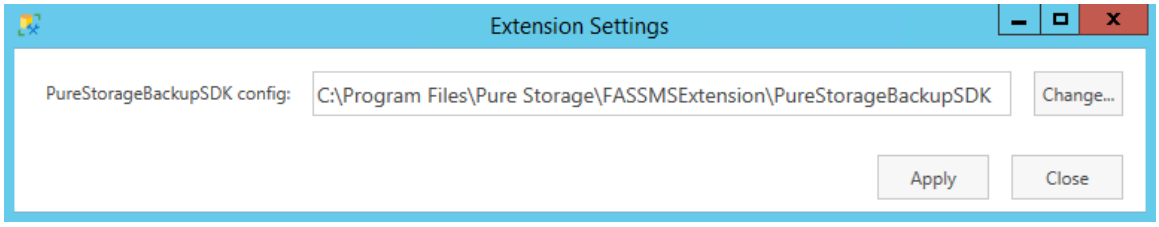
To remove a backup entry:

1. In the SSMS tool bar, select **Pure Storage > Backup History**, or in the Object Explorer, select the database to be backed up, right click, and select **Pure Storage > Backup History**.
   If necessary, search or filter the list to locate the specific backup.
2. Select the backup and click the **Remove** button.
   Click **Yes** in the Remove Confirmation dialog.

Edit Extension Settings

The Extension Settings dialog shows the location of the Pure Storage Backup SDK. We recommend not changing this field.

Figure 4.13:

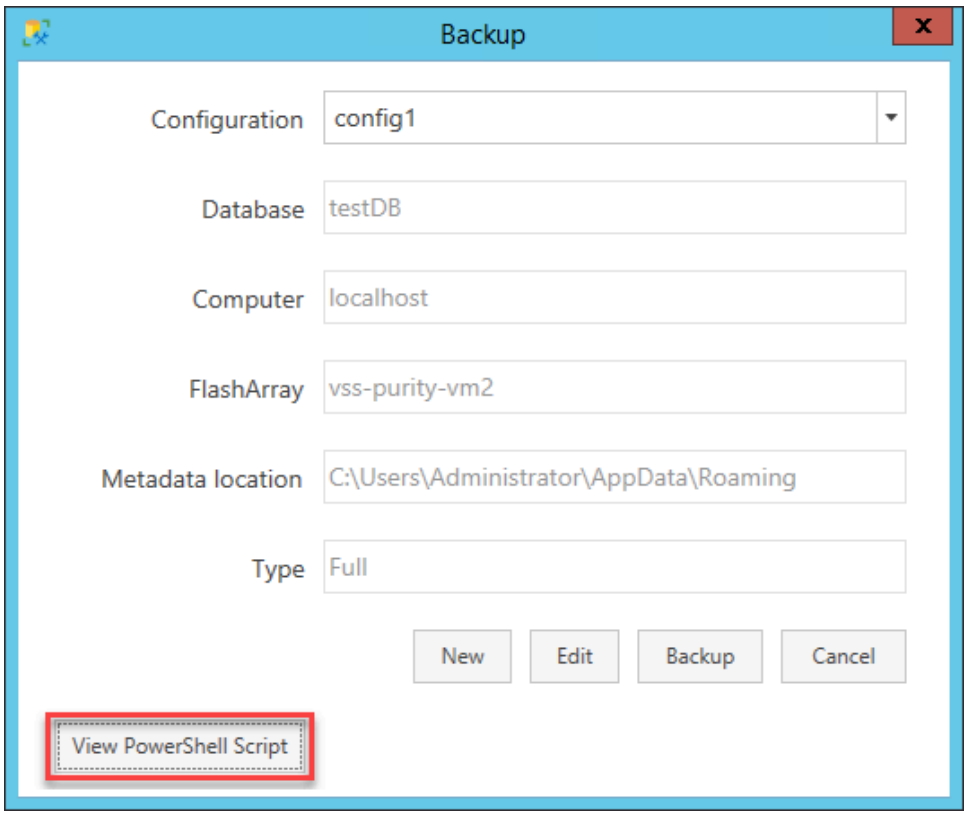


Note about the View PowerShell Script Button

Many FA SSMS Extension dialogs include a **`View PowerShell Script`** button, which displays the PowerShell Backup SDK commands equivalent to the dialog's action.

Use this button *after* you have filled out the dialog's information. For example:

Figure 4.14:



See also *Cmdlets List, Cmdlets Help, and Cmdlet Equivalents*.

Chapter 5. The Pure Storage PowerShell Backup SDK

SDK Overview

Pure Storage® provides the PowerShell Backup SDK to perform and automate operations supported by the Pure Storage FA SSMS Extension, including creating SQL backups and restoring from backups.

The SDK orchestrates the work flow required for database backup, restore, mount, and dismount for Microsoft SQL Servers.

The Backup SDK connects the SQL Server™ host and FlashArray™ (FA) connected to the SQL Server host and orchestrates the required operations needed to backup, restore, mount, and dismount.

Installation and Import

The PowerShell Backup SDK is installed during the FA SSMS Extension installation. Its default location is:

```
%PROGRAMFILES%\Pure Storage\FASSMSExtension\PureStorageBackupSDK
```

In order to use the Backup SDK, open the Windows PowerShell app and run the following PowerShell command from the install directory:

```
Import-Module -Name PureStorageBackupSDK
```

Cmdlets List, Cmdlets Help, and Cmdlet Equivalents

The following PowerShell command and lists the available cmdlets:

```
Get-Command -Module PureStorageBackupSDK
```

The output is:

```
CommandType     Name                               Version    Source
-----------     ----                               -------    ------
Function        Add-PfaBackupCred                  1.0.438    PureStorageBackupSDK
Function        Add-PfaBackupJob                   1.0.438    PureStorageBackupSDK
Function        Connect-PfaVMHardDisk              1.0.438    PureStorageBackupSDK
Function        Dismount-PfaDrive                  1.0.438    PureStorageBackupSDK
Function        Get-AvailableVM                    1.0.438    PureStorageBackupSDK
Function        Get-ESXiDetails                    1.0.438    PureStorageBackupSDK
Function        Get-PfaAvailableDBNames            1.0.438    PureStorageBackupSDK
Function        Get-PfaBackupAvailableDrives       1.0.438    PureStorageBackupSDK
Function        Get-PfaBackupCred                  1.0.438    PureStorageBackupSDK
Function        Get-PfaBackupCredList              1.0.438    PureStorageBackupSDK
Function        Get-PfaBackupHistory               1.0.438    PureStorageBackupSDK
```

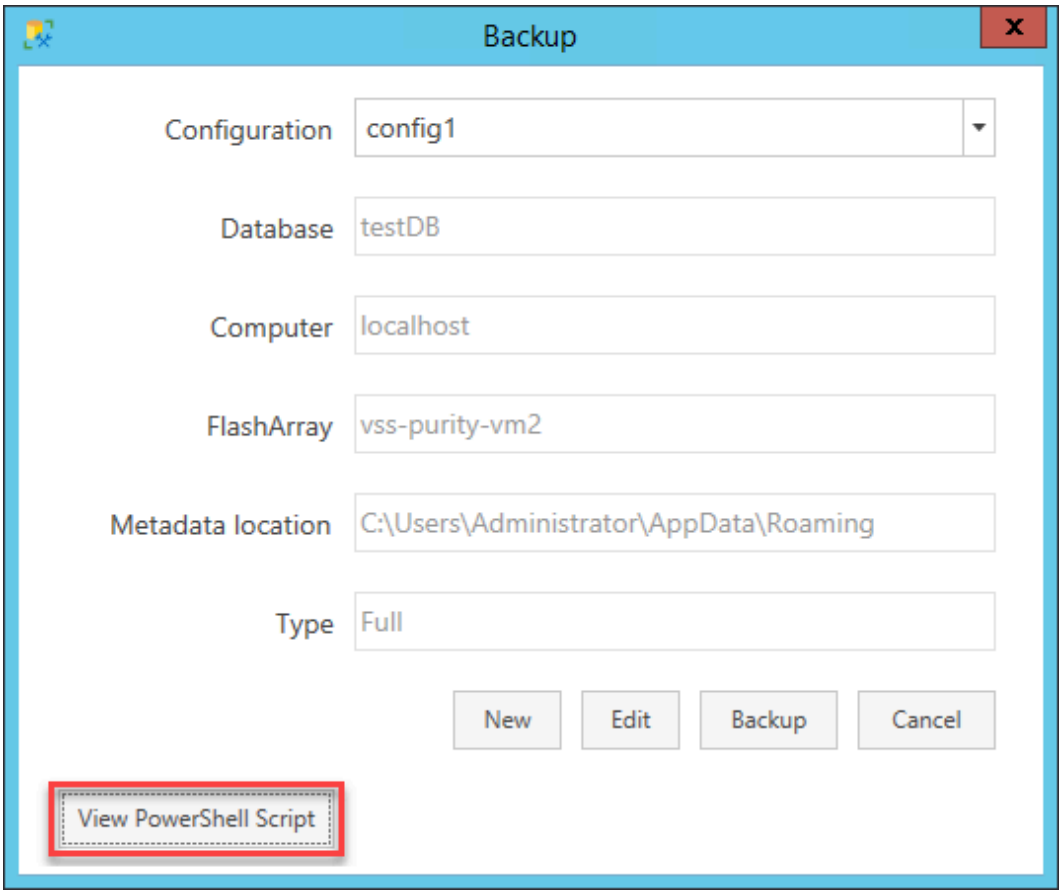
```
Function        Get-PfaBackupJobs               1.0.438    PureStorageBackupSDK
Function        Get-PfaConnectedFlashArray      1.0.438    PureStorageBackupSDK
Function        Get-PfaVSSInstallerVersion      1.0.438    PureStorageBackupSDK
Function        Install-VSS                     1.0.438    PureStorageBackupSDK
Function        Invoke-PfaBackupJob             1.0.438    PureStorageBackupSDK
Function        Mount-PfaBackupJob              1.0.438    PureStorageBackupSDK
Function        Remove-PfaBackupCred            1.0.438    PureStorageBackupSDK
Function        Remove-PfaBackupHistory         1.0.438    PureStorageBackupSDK
Function        Remove-PfaBackupJob             1.0.438    PureStorageBackupSDK
Function        Remove-PfaVMHardDisk            1.0.438    PureStorageBackupSDK
Function        Restore-PfaBackupJob            1.0.438    PureStorageBackupSDK
```

Use the **`Get-Help`** command for a detailed explanation of each cmdlet:

```
   Get-Help <cmdlet> -Detailed
```

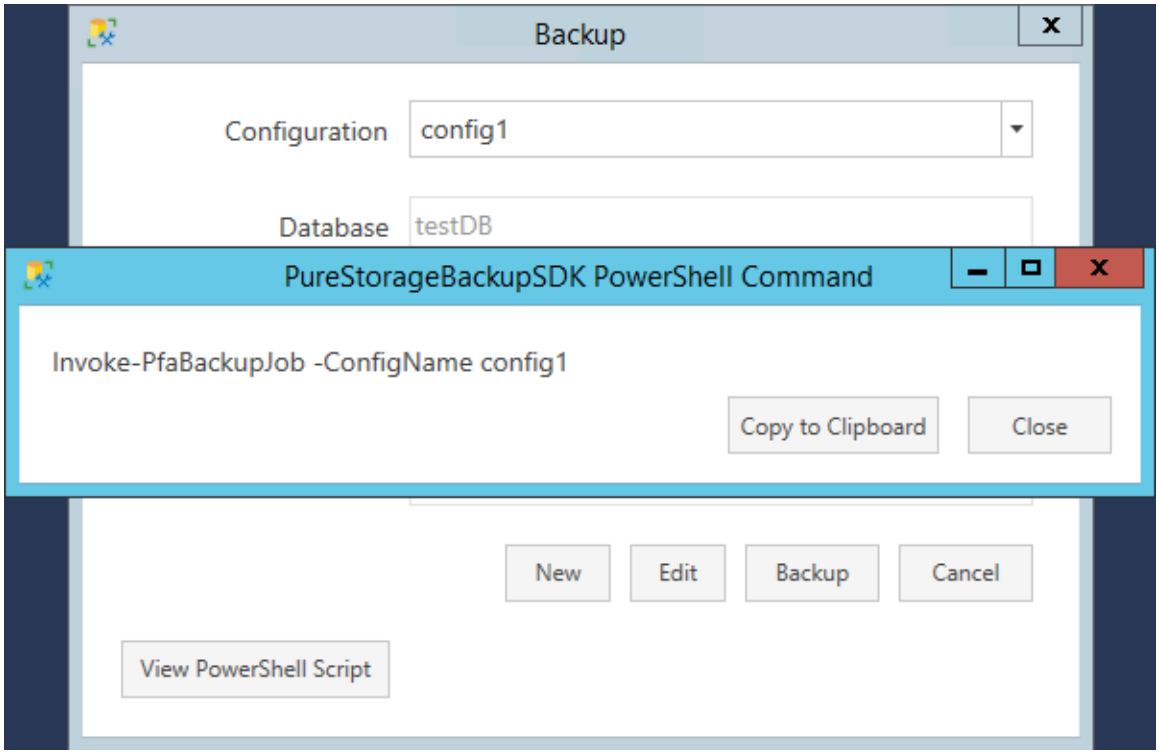
The FA SSMS Extension GUI provides a **PowerShell Script** that lists the Backup SDK cmdlets equivalent to the operations provided in the GUI. For example:

Figure 5.1:



The equivalent Backup SDK cmdlets are displayed in a pop-up. For example:

Figure 5.2:



In the Backup History dialog, the **PowerShell Script** provides the cmdlet equivalent for the Remove button. To see the cmdlet equivalents for the Restore and Mount buttons, click them and complete the subsequent popups, then click the **PowerShell Script** on the completed popup.

Using the PowerShell Backup SDK Cmdlets

The following is a typical workflow sequence with the SDK cmdlets:

1. Adding SQL Server host computer credentials
2. Adding FlashArray credentials
3. Creating a backup job configuration
4. Invoking a backup job configuration
5. Restoring or mounting a snapshot

You need to input the SQL Server host computer credentials and the Pure Storage FlashArray credentials when creating a backup configuration.

Adding SQL Server Host Computer Credential

Enter the following command (all on one line):

```
Add-PfaBackupCred -CredentialName "mySqlServer" -Address <sql-server-name>
   -CredentialType Windows -Credential (Get-Credential)
```

`CredentialName` is a friendly name to identify the Windows server.

Use **`Get-PfaBackupCred`** to check the credential that is added into the Pure Storage Backup SDK:

```
Get-PfaBackupCred -CredentialName mySqlServer -CredentialType Windows
```

As part of adding the credential to the Pure Storage Backup SDK, we do ensure that the server is alive and the credentials are accurate. This is also true when adding credentials for FlashArray and vCenter below.

If there is an issue, the cmdlet throws an exception.

Please ensure that the server is connected via network to this client machine where the credentials are being added and that the PowerShell remoting is enabled and operational.

The credentials are stored in the client computer at the following location:
`%PROGRAMDATA%\PureStorage\PureStorageBackupSDK\<username>\`

Adding FlashArray Credentials

Enter the following command (all on one line):

```
Add-PfaBackupCred -CredentialName "myFlashArray" -Address <flasharray-name>
    -CredentialType FlashArray -Credential (Get-Credential)
```

`CredentialName` is a friendly name to identify the FlashArray.

Use **`Get-PfaBackupCred`** to check the credential that is added into PureStorage Backup SDK.

```
Get-PfaBackupCred -CredentialName myFlashArray -CredentialType FlashArray
```

The credentials are stored in the client computer at the following location:
`%PROGRAMDATA%\PureStorage\PureStorageBackupSDK\<username>\`

Adding VMware vCenter Credentials

If the SQL Server host computer is a VMware virtual machine hosted on ESXi and the backing datastore is configured as RDM (raw device mapping), vCenter credentials are required to configure the ESXi host to update the backing datastore.

Enter the following command (all on one line):

```
Add-PfaBackupCred -CredentialName myVCenter -CredentialType vCenter -Address
   <vcenter-machine-name> -Credential (Get-Credential)
```

The credentials are stored in the client computer at the following location:
`%PROGRAMDATA%\PureStorage\PureStorageBackupSDK\<username>\`

Creating a Backup Job Configuration

We define a backup job that encompasses the details required to perform a backup. The configuration includes the SQL Server host computer name, the SQL Server instance name, and the database name.

If the SQL Server host is a virtual machine configured as RDM, then the vCenterName and VMName are also included.

```
Add-PfaBackupJob -ConfigName "myTestBackupConfig" -ComputerName "mySqlServer"
   -FAName "myFlashArray" -Component testDB
```

Use the following command to define a backup job configuration for a SQL Server host computer that is a virtual machine configured as RDM:

```
Add-PfaBackupJob -ConfigName "myTestBackupConfig" -ComputerName "mySqlServer"
   -FAName "myFlashArray" -VCenterName "myVCenter" -VMName "virtualMachineName"
   -Component testDB
```

Notes:

- It is mandatory to add the correct FlashArray details when creating a backup job configuration. Adding a FlashArray that is not the backing storage for the SQL Server host will interfere with backup, restore, and mount commands.
- The **`ComputerName`** and **`FAName`** here are the friendlyName configured in the **`Add-PfaBackupCred as CredentialName`**.

This helper function identifies the FlashArray corresponding to a SQL Server database. This takes the list of FlashArray friendly names previously configured using **`Add-PfaBackupCred`**. This command returns the FlashArray that corresponds to this database.

```
Get-PfaConnectedFlashArray -ComputerName mySqlServer -FAName
   myFlashArray,correctFlashArray -DBName testDB
```

Important: All database-related files for the above-mentioned component **`testDB`** must reside on the Pure Storage FlashArray disks.

Invoking a Backup Job Configuration

The command to take a backup uses the friendly configuration name given when the configuration was created.

Run the following command to perform a backup:

```
Invoke-PfaBackupJob -ConfigName "myTestBackupConfig"
```

This creates a Pure Storage snapshot on the FlashArray for the volume corresponding to the database specified in the backup job configuration.

Note that the database may belong to multiple disk drives on SQL Server host computer, in which case each disk drive corresponds to a FlashArray volume (also known as a FlashArray LUN).

When a backup is performed, a snapshot is taken of all volumes. The Pure Storage Backup SDK orchestrates between the client computer where the Backup SDK is installed and the SQL Server host computer (and vCenter and VM, if virtual) and FlashArray to create the snapshot. The metadata associated with this snapshot is created by the Pure Storage VSS Hardware Provider running on the SQL Server host computer.

This metadata is copied to the SSMS computer and stored in the default location of **C:\Users\<*username*>\AppData\Roaming**.

You can override the path at which the metadata file is copied using the **-MetadataDir** argument to the **Add-PfaBackupJob** cmdlet.

This metadata file is important as it is required for a restore or a mount of the snapshot. If a backup's metadata file is not provided for a restore, the restore operation fails.

The result of invoking a backup job is stored in the **history.json** file, located at **%PROGRAMDATA%\PureStorage\PureStorageBackupSDK\<username>\history.json**.

Listing Backup Job Configurations

The **Get-PfaBackupJobs** command lists the existing backup configurations.

This example shows two backup configurations, **config1** and **config2**.

```
Get-PfaBackupJobs

ConfigName        : config1
ComputerName      : localhost
FAName            : vss-purity-vm
vCenterName       :
VMName            :
Component         : testDB
Path              :
CopyOnly          : False
Metadata          :
MetadataDir       : C:\Users\Administrator\AppData\Roaming
MetadataOverwrite : False

ConfigName        : config2
ComputerName      : localhost
FAName            : vss-purity-vm
vCenterName       :
VMName            :
Component         : testDB
Path              :
CopyOnly          : True
Metadata          :
MetadataDir       : C:\Users\Administrator\AppData\Roaming
MetadataOverwrite : False
```

Retrieving a Historical Backup

The **Get-PfaBackupHistory** command lists backup history.

```
Get-PfaBackupHistory
```

```
HistoryId : 0
ConfigName : myTestBackupConfig
Path : E:
Component : testDB
Metadata : Metadata_SQL-RDM_20190930141943
MetadataDir : C:\Users\Administrator\AppData\Roaming\
MetadataFullPath : C:\Users\Administrator\AppData\Roaming\Metadata_SQL-RDM_20200...
Status : Succeeded
TimeCreated : 09/30/2019 14:19:45
BackupType : Full
ComputerName : mySqlServer
FAName : myFlashArray
VCenterName : myVCenter
VMName : sql-rdm
MountComputer :
MountDrive :
```

Each invocation of the backup job using Invoke-BackupJob creates an entry into the backup history file. Each entry has a unique HistoryId and corresponding Status denoting Successful/Failed.

If the snapshot is successful, then HistoryId can be used to restore the database to that instance of the data.

Restoring the Database to a Snapshot

The Restore-PfaBackupJob cmdlet restores a database. First run the following cmdlet to determine the **`HistoryId`** of the appropriate backup:

```
Get-PfaBackupHistory | Format-Table AutoSize
```

Run the following cmdlet to restore the database to the point of the snapshot:

```
Restore-PfaBackupJob -HistoryId 0
```

Using **`-NoRecovery`**, the database is restored in no recovery mode.

```
Restore-PfaBackupJob -HistoryId 0 -NoRecovery
```

The restore operation cause all changes since the point the backup was made to be lost.

Mounting a Backup Snapshot

The **`Mount-PfaBackupJob`** command mounts a backup snapshot on to one or more Windows drive letters or mount points.

Example command:

```
Mount-PfaBackupJob -HistoryId 0 -DriveLetter "E:"
```

Replace "E:" with a valid available drive letter.

Multiple drives are supported, if the source involved multiple drives. Separate multiple drive letters with a comma (no spaces) and enclose them with double quotes. For example: **"E:,F:"**.

Use **-MountComputer** <computer credential name> of the computer where you want to mount the snapshot. If this argument is not used, then we mount on the SQL Server host computer where the database belongs.

Use **-MountVMName** if the **-MountComputer** is a virtual machine hosted on ESXi. **MountVMName** is the credential name of the VM previously configured using **Add-PfaBackupCred**.

The **Mount-PfaBackupJob** argument creates a new drive on the MountComputer or SQL Server host with the copy of the data at the point of snapshot taken with HistoryId 0.

The result of the mounted drive letter and whether it was successful is stored in the history configuration.

Dismount a mounted Drive

Dismounts a database snapshot from an existing mounted drive.

```
Dismount-PfaDrive -HistoryId 0
```

The history configuration identifies the drive letter that mounted the HistoryId 0 and the mount computer. This information is used to dismount.

Utility Cmdlets

Install-VSS

This cmdlet installs the VSS Hardware Provider on the SQL Server host computer.

Use **-ForceUpgrade** to upgrade or reinstall VSS.

Get-AvailableVM

This cmdlet returns the virtual machine names managed by specified vCenter that match the specified SQL Server.

Get-ESXiDetails

This cmdlet returns the ESXi that hosts the specified VM name.

Get-PfaAvailableDBNames

This cmdlet finds databases that exist on the SQL Server host computer.

Connect-PfaVMHardDisk

This cmdlet can be used to connect a volume (given a volume serial number) to a virtual machine computer that is hosted on ESXi using raw device mapping (RDM).

Note: The ESXi should be connected to the FlashArray using RDM configuration.

Example:

```
Connect-PfaVMHardDisk -VolumeSerial AC02BDF1FF54147900012197 -VMName sql-rdm
    -VCenterName test-vcenter
```

Remove-PfaVMHardDisk

This cmdlet disconnects a volume from a virtual machine computer that is hosted on an ESXi using raw device mapping (RDM) configuration.

Get-PfaBackupAvailableDrives

This cmdlet gets a list of available drive letters on the SQL Server host computer.

Get-PfaBackupCredList

This cmdlet gets the credential list for a vCenter, FlashArray, or Windows machine.

Remove-PfaBackupCred

This cmdlet removes a credential from the Extension's list of known credentials.

Remove-PfaBackupHistory

This cmdlet removes an entry from the backup history list or deletes all entries from the backup history list.

Remove-PfaBackupJob

This cmdlet removes a backup configuration or deletes all backup configurations.

Get-PfaConnectedFlashArray

This cmdlet identifies the connected FlashArray to a SQL Server host computer. This cmdlet takes a list of FlashArray credential names (the credential names previously configured using **`Add-PfaBackupCred`**).

Get-PfaVSSInstallerVersion

This cmdlet identifies the VSS Hardware Provider version number on the SQL Server host computer.

TroubleShooting

Debug logs on the client and server machines are stored at **`%PROGRAMDATA%\PureStorage\`**.

Client Machine Logs

This log contains the Pure Storage Backup SDK debug messages that are crucial to understanding a failure:

```
<username>_PureStorageBackupSDK.log
```

SQL Server Host Computer Logs

The following log files are stored on the SQL server host computer and are crucial in debugging:

```
PureSQLSnap.log
PureVSSHardwareProvider.log
```

Error Messages Related to TrustedHosts Issues

Error messages similar to the following indicate that the remote SQL Server has not been added to your local TrustedHosts list:

```
The running command stopped because the preference variable "ErrorActionPreference"
or common parameter is set to Stop: [10.01.01.01] Connecting to remote server
10.01.01.01 failed with the following error message : The WinRM client cannot
process the request.
Default authentication may be used with an IP address under the following conditions:
the transport is HTTPS or the destination is in the TrustedHosts list, and explicit
credentials are provided.
Use winrm.cmd to configure TrustedHosts.
Note that computers in the TrustedHosts list might not be authenticated.
For more information on how to set TrustedHosts run the following command:
winrm help config.
For more information, see the about_Remote_Troubleshooting Help topic.
```

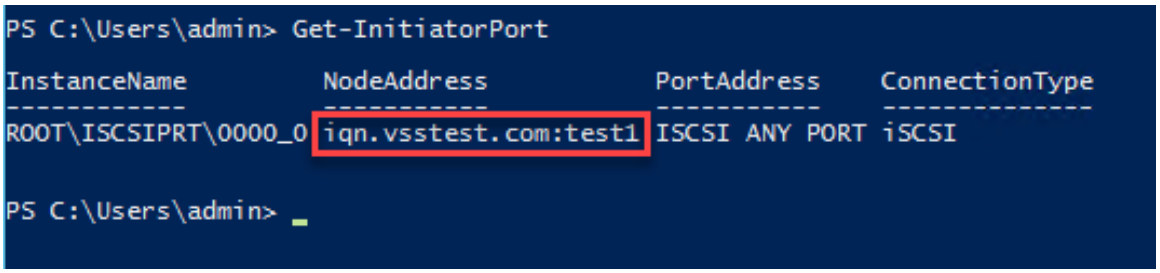
Appendix A. Example of Configuring a Database Hosted on a FlashArray Volume

This section describes how to configure a SQL Server database that is hosted on a FlashArray volume. This example shows configuration with an iSCSI initiator. However, both Fibre Channel and iSCSI are supported.

1. **On the FlashArray, create a host for the SQL database:**

   a. First, find a port on the SQL Server host.
   On the SQL Server, in PowerShell run **`Get-InitiatorPort`**.

   Figure 22:

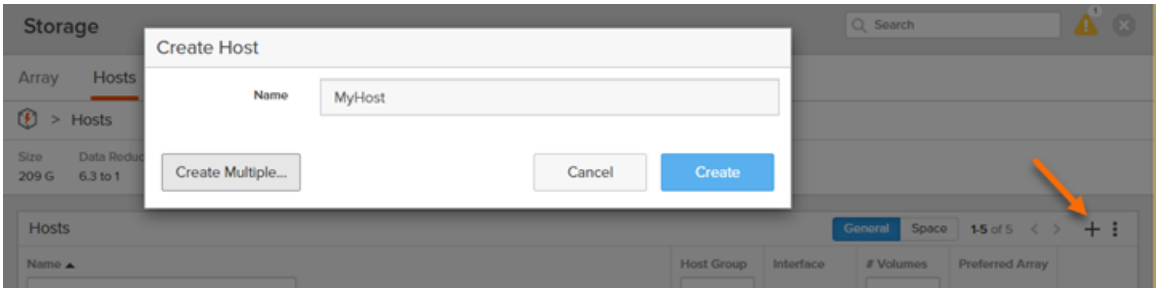


   b. Copy the text under **`NodeAddress`**.
   (The NodeAddress is **`iqn.vsstest.com:test1`** in this example.)

2. **Create a host on the FlashArray:**

   a. On the FlashArray, go to **Storage > Hosts**.
   Click the plus icon **+** to create a new host.

   Figure 23:



   b. Give the host a name and click **Create**.

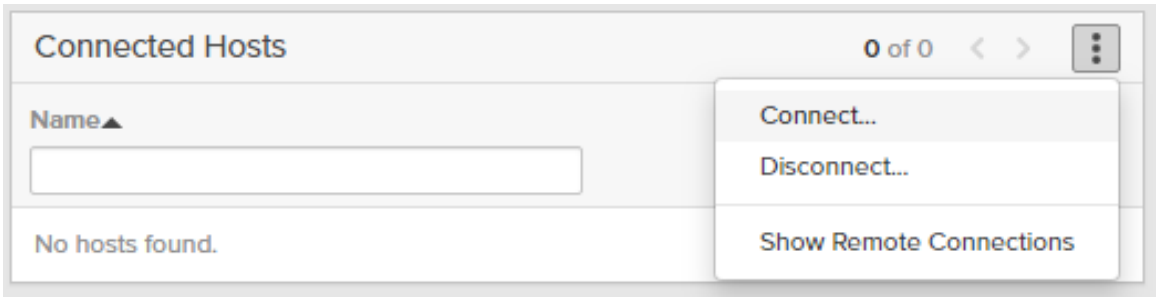
   c. Click on the newly created host.
   Under Host Ports, select **Configure IQNs**.

Figure 24:

Enter the **`NodeAddress`** from the PowerShell step above.

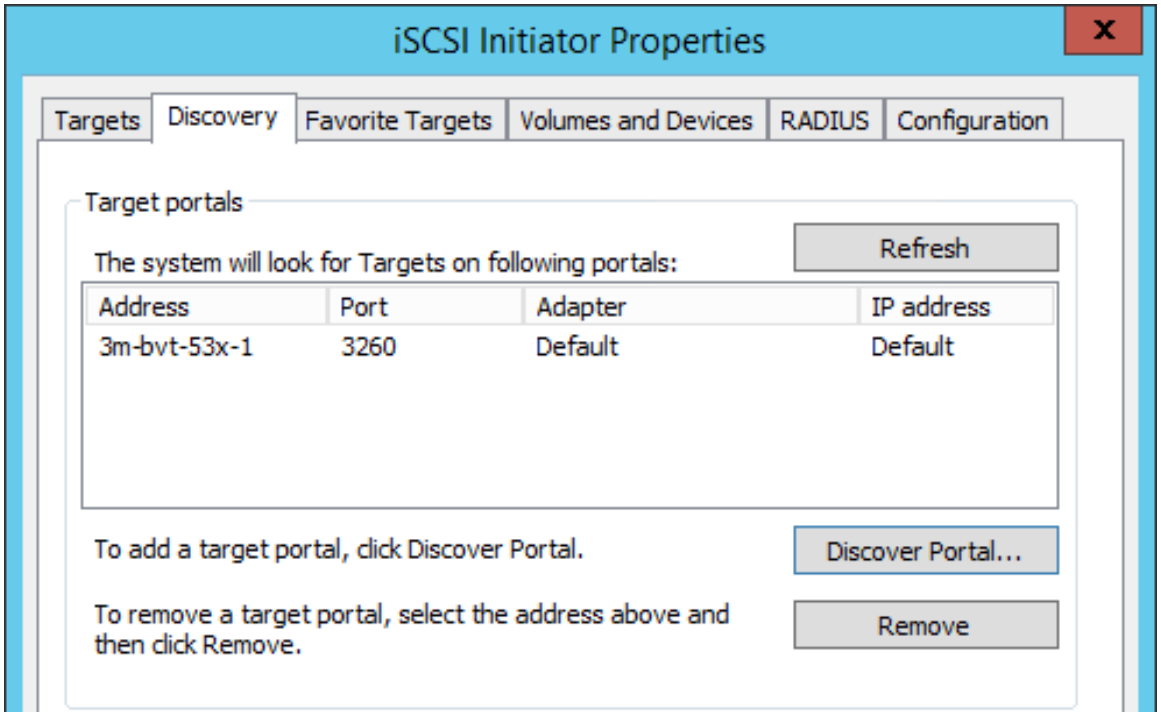
3. **Create a volume on the FlashArray:**
   a. On the FlashArray, go to **Storage > Volumes**
   Click the plus icon **+** to create a new volume.
   b. Click on the newly created volume.
   c. In the Connected Hosts pane, connect the host you created above.

Figure 25:



4. **On the SQL Server, connect to the FlashArray:**
   a. On the SQL server, run **`iSCSI Initiator`**.
   b. Under Discovery, click **Discover Portal**.

Figure 26:



c. Enter the hostname of the FlashArray

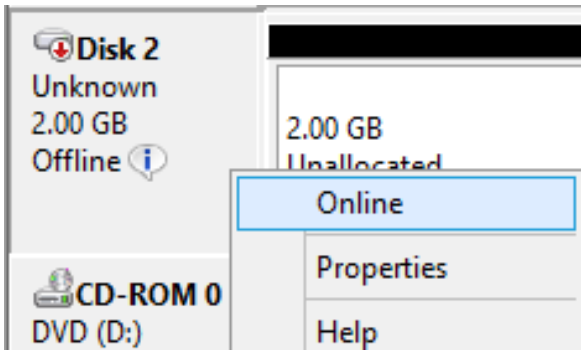
d. In iSCSI Initiator, under Targets, click **Connect**.

5. **Create a volume on the SQL Server:**

a. Go to **Disk Management**.

b. Select the new volume, right click it, and bring it online.

Figure 27:



c. Right click again and click **Initialize Disk**.

d. Right click in the partition space and create a new volume.

e. Format the new volume.

6. **Create the new database on the Pure Volume on the SQL Server:**
   a. In Microsoft SQL Server Management Studio, right-click on **Databases > New Database**.
   b. Edit the path of the new database to be on the newly-created volume on the FlashArray.

Appendix B. Resources

Links are subject to change without notice.

Pure Storage Documentation

The following Purity resources are available at ***Pure1 Support*** [http://support.purestorage.com] (registration and login are required).

Pure Storage solutions for the Microsoft platform:

- ***Pure Storage Microsoft Platform Guide*** [http://support.purestorage.com/Solutions/Microsoft_Platform_Guide]
  This section provides details on best practices for configuring Microsoft Windows Server with a Pure Storage FlashArray.
- ***Best Practices: Microsoft SQL Server*** [https://support.purestorage.com/Solutions/Microsoft_Platform_Guide/Microsoft_SQL_Server/001_Microsoft_SQL_Server_Quick_Reference]
  This section provides our recommendations for SQL Server databases hosted on Pure Storage FlashArrays.
- ***Volume Shadow Copy Service (VSS)*** [http://support.purestorage.com/Solutions/Microsoft_Platform_Guide/L_Volume_Shadow_Copy_Service_(VSS)/Volume_Shadow_Copy_Service_(VSS)]
  The Pure Storage VSS Hardware Provider is an integration that provides the ability to take application-consistent snapshots for Microsoft SQL Server.
- ***Troubleshooting the Volume Shadow Copy Service (VSS)*** [http://support.purestorage.com/Solutions/Microsoft_Platform_Guide/L_Volume_Shadow_Copy_Service_(VSS)/Troubleshooting_the_Volume_Shadow_Copy_Service_(VSS)]
  Troubleshooting information for the Pure Storage VSS Hardware Provider.

FlashArray information:

- ***Purity FlashArray User Guides*** [http://support.purestorage.com/FlashArray/PurityFA/FlashArray_User_Guide]
  This page contains PDFs of the FlashArray User Guide for current and past releases, and an HTML version of the guide for the most recent Purity//FA release.

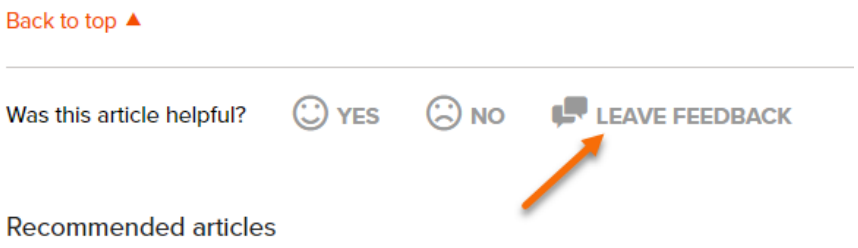
  Instructions for configuring volumes, hosts, and LUNs are included in the FlashArray User Guide.
- ***REST API PDF Reference Guides*** [http://support.purestorage.com/FlashArray/PurityFA/Purity_FA_REST_API/Reference/REST_API_PDF_Reference_Guides]
  This page contains PDFs of the REST API Reference Guide for current and past releases and an HTML version of the most recent REST API release.

Documentation Feedback

Feedback on Pure1 Support pages

The Microsoft Platform Guide and other Pure1 Support pages include a feedback link at the bottom of the page.

Figure 28: Pure1 Support Feedback Link



Your feedback is registered as sent by the email address associated with your Pure1 Support login.

You can optionally give permission to be contacted by email about your comments. (The default is not to contact you.)

Feedback Through Email

We welcome your feedback about Pure Storage documentation and encourage you to send your questions and comments to **<DocumentFeedback@purestorage.com>**. We would love to hear from you.

Appendix C. Copyright and Notices

End User Agreement

End User Agreement [http://www.purestorage.com/agreements/Pure_enduser_agreement.pdf]

Copyrights and Licenses

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

Pure Storage, Inc.
Twitter: @purestorage

650 Castro Street, Suite 260
Mountain View, CA 94041
800-379-7873

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