Microsoft System Center 2012 R2

Cmdlet Reference for System Center 2012 R2 Data Protection Manager

Microsoft Corporation

Published: November 1, 2013

Applies To

System Center 2012 R2 Data Protection Manager

Feedback

Send suggestions and comments about this document to sc2012docs@microsoft.com.

Copyright

Revision History

Release Date	Changes
November 1, 2013	Initial release of this document.

Contents

Add-DPMBackupNetworkAddress

Add-DPMBackupNetworkAddress

Specifies a backup network for a DPM server.

Syntax

```
Parameter Set: Default
Add-DPMBackupNetworkAddress [[-DPMServerName] <String> ] [-Address] <String> [-
SequenceNumber] <UInt32> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-DPMBackupNetworkAddress** cmdlet specifies an address of a backup network for a System Center 2012 – Data Protection Manager (DPM) server to use. DPM allows you to configure backup network addresses so that DPM backups do not slow down your primary network. Assign a priority for each backup network you add.

Parameters

-Address<String>

Specifies the IP address or subnet mask of a backup network.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SequenceNumber<UInt32>

Specifies the priority of this address for use as a backup network.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type `"Get-Help Add-DPMBackupNetworkAddress -detailed"`.

Examples

Example 1: Add a backup network address

This command adds the subnet address 10.10.10.0/16 as the backup network address for a DPM server named DPMServer07, with a sequence value of 1.

```
PS C:\> Add-DPMBackupNetworkAddress -DpmServername "DPMServer07" -Address "10.10.10.0/16" -
SequenceNumber 1
```

Related topics

Get-DPMBackupNetworkAddress

Remove-DPMBackupNetworkAddress

Add-DPMChildDatasource

Add-DPMChildDatasource

Adds a data source or a child data source to a protection group.

Syntax

```
Parameter Set: Default
Add-DPMChildDatasource [-ProtectionGroup] <ProtectionGroup> [-ChildDatasource]
<ProtectableObject[]> [[-Online]] [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-DPMChildDatasource** cmdlet adds a data source or a child data source to a System Center 2012 – Data Protection Manager (DPM) protection group. A child data source refers to folders on a protected volume.

You can override a data source that a previous **Remove-ChildDatasource** cmdlet excluded from a protection group by adding the data source to the protection group.

Parameters

-ChildDatasource<ProtectableObject[]>

Specifies an array of data sources, such as folders in a file system, that you can protect individually.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Online

Indicates whether online protection is enabled.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Add-DPMChildDatasource -detailed`".

Examples

Example 1: Add a child data source to a protection group

This example adds a child data source to a protection group.

The first command gets all protection groups from the DPM server named DPMServer07 and stores these groups in the $Pg variable. You cannot edit these protection groups.

The second command gets the first protection group in the $Pg array in editable mode, and then stores it in the $Mpg variable.

The third command gets an arrayof protected and unprotected data on the production server named ProductionServer22, and stores the array in the $Po variable.

The fourth command uses standard array notation to specify the ninth element of the $Po array. The command adds that data source to the protection group stored in the $Mpg variable.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer07"
PS C:\> $Mpg = Get-ModifiableProtectionGroup $Pg[0]
PS C:\> $Po = Get-Datasource -ProductionServer "ProductionServer22"
PS C:\> Add-DPMChildDatasource -ProtectionGroup $Mpg -ChildDatasource $Po[8]
```

Related topics

Get-DPMChildDatasource

Remove-DPMChildDatasource

Add-DPMDisk

Add-DPMDisk

Adds a disk to a storage pool on a DPM server.

Syntax

```
Parameter Set: Default
Add-DPMDisk [-DPMDisk] <Disk[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-DPMDisk** cmdlet adds a disk to a storage pool on a System Center 2012 – Data Protection Manager (DPM) server. A storage pool on a DPM server consists of a set of disks where the server stores replicas, shadow copies, and transfer logs for protected data sources.

To get a list of all disks on a DPM server, use the **Get-DPMDisk** cmdlet.

Parameters

-DPMDisk<Disk[]>

Specifies an array of one or more disks.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Disk**

Notes

- For more information, type "`Get-Help Add-DPMDisk -detailed`".

Examples

Example 1: Add a disk to a DPM server

This example adds a disk to a DPM server.

The first command uses the **Get-DPMDisk** cmdlet to retrieve a list of disks on a server, and stores the result in the $DPMDisk variable.

The second command uses the **Add-DPMDisk** cmdlet to add the disks in the variable $DPMDisk to the server storage pool.

```
PS C:\> $DPMDisk = Get-DPMDisk -DPMServerName "Contoso-DPMServer"
PS C:\> Add-DPMDisk -DPMDisk $DPMDisk
```

Related topics

Get-DPMDisk

Remove-DPMDisk

Add-DPMRecoveryItem

Add-DPMRecoveryItem

Identifies recoverable items.

Syntax

```
Parameter Set: Datasources
Add-DPMRecoveryItem [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [-
Datasources] <SQLDataSource[]> [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: SqlInstances
Add-DPMRecoveryItem [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [-
SqlInstances] <String[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-DPMRecoveryItem** cmdlet identifies the recoverable items that members of a System Center 2012 – Data Protection Manager (DPM) role can recover.

Parameters

-Datasources<SQLDataSource[]>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.

-- Microsoft Exchange storage group.

-- Microsoft SharePoint Server farm.

-- Microsoft Virtual Machine.

-- DPM database.

-- A system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	3

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SqlInstances<String[]>

Specifies an array of instances of SQL Server.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AmDatasourceType>

Specifies the type of data source that the cmdlet uses. Valid values are:

-- SqlDatabase

-- SqlInstance

-- Client

The acceptable values for this parameter are:

SqlDatabase	
SqlInstance	
Client	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Add-DPMRecoveryItem -detailed`".

Examples

Example 1: Add recoverable items for a DPM role

This example adds a SQL database and a SQL instance to the recovery items that members of a DPM role can recover.

The first command gets the protection groups on the DPM server named DPMServer02 and stores the result in the $ListOfPGs variable.

The second command gets the protected and unprotected data from the first protection group stored in the $ListOfPGs variable. The command stores the result in the $DatasourceInPG variable.

The third command adds a SQL database to the recovery items that members of the DPM role stored in the $Role variable can recover. The data source stored in the $DatasourceInPG variable contains the SQL database.

The fourth command adds a SQL instance to the recovery items that members of the DPM role stored in the $Role variable can recover.

```
PS C:\> $ListOfPGs = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $DatasourceInPG = Get-DPMDatasource -ProtectionGroup $ListOfPGs[0]
PS C:\> Add-DPMRecoveryItem -DPMRole $Role -Type SQLDatabase -Datasource $DatasourceInPG
PS C:\> Add-DPMRecoveryItem -DPMRole $Role -Type SQLInstance -Sqlinstances
"ProductionServer\Unit01Instance"
```

Related topics

Remove-DPMRecoveryItem

Get-DPMRecoveryItem

Get-DPMRole

Add-DPMRecoveryTarget

Add-DPMRecoveryTarget

Grants the DPM role permission to recover to a location.

Syntax

```
Parameter Set: Default
Add-DPMRecoveryTarget [-DpmRole] <DpmRole> [-RecoveryTargets] <TargetRecoveryItem[]> [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-DPMRecoveryTarget** cmdlet grants the System Center 2012 – Data Protection Manager (DPM) role permission to recover to a location.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryTargets<TargetRecoveryItem[]>

Specifies an array of target recovery items which consist of the instance of SQL Server and the folder to use for alternate instance recovery.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Add-DPMRecoveryTarget -detailed`".

Examples

Example 1: Grant a role permission to recover to a location

This example grants a DPM permission to recover to an instance of SQL Server.

The first command gets a DPM role named OpsMgrSQL and stores the result in the $Role variable.

The second command gets the recovery target for the role stored in the $Role variable, and stores the result in the $RecoveryTargetInstance variable.

The third command grants the DPM role stored in the $Role variable permission to recover to the target instance of SQL Server stored in the $RecoveryTargetInstance variable.

```
PS C:\> $Role = Get-DPMRole -Name "OpsMgrSQL"
PS C:\> $RecoveryTargetInstance = Get-DPMRecoveryTarget -DpmRole $Role -Type SQLInstance
PS C:\> Add-DPMRecoveryTarget -Role $Role -RecoveryTargets $RecoveryTargetInstance
```

Related topics

Add-DPMRecoveryTarget

New-DPMRecoveryTarget

Get-DPMRecoveryTarget

Remove-DPMRecoveryTarget

Add-DPMSecurityGroup

Add-DPMSecurityGroup

Adds security groups to a DPM role.

Syntax

```
Parameter Set: Default
Add-DPMSecurityGroup [-DpmRole] <DpmRole> [-SecurityGroups] <String[]> [-Confirm] [-WhatIf]
[ <CommonParameters>]
```

Detailed Description

The **Add-DPMSecurityGroup** cmdlet adds one or more security groups to a System Center 2012 – Data Protection Manager (DPM) role. You can use the **Get-DPMRole** to specify a DPM role. Use the **Set-DPMRole** cmdlet to save your changes.

You can see the security groups for a DPM role by using the **Get-DPMSecurityGroup** cmdlet. You can use the **Remove-DPMSecurityGroup** cmdlet to remove security groups from a DPM role.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SecurityGroups<String[]>

Specifies an array of security groups. The cmdlet adds these groups to the DPM role.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Add-DPMSecurityGroup -detailed`".

Examples

Example 1: Add a security group to a role

This example adds a security group to a DPM role.

The first command uses the **Get-DPMRole** cmdlet to get the role named OpsMgrSQL and stores it in the $DpmRole variable. The command makes the role editable.

The second command adds the specified security group to the role stored in the $DpmRole variable.

```
PS C:\> $DpmRole = Get-DPMRole -Name "OpsMgrSQL" –-Editable
PS C:\> Add-DPMSecurityGroup -DpmRole $DpmRole -SecurityGroups "Hq\OpsMgrSQL"
```

Related topics

Get-DPMSecurityGroup

Remove-DPMSecurityGroup

Get-DPMRole

Set-DPMRole

Add-DPMTape

Add-DPMTape

Adds a tape to a DPM library.

Syntax

```
Parameter Set: Default
Add-DPMTape [-DPMLibrary] <Library> [-Async] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-DPMTape** cmdlet adds a tape to a System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation to the user. Use this parameter in conjunction with the *Async* parameter

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Add-DPMTape -detailed`".

Examples

Example 1: Add a tape to a library

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library for the server, and stores the result in the $DPMLib variable.

The second command uses the **Add-DPMTape** cmdlet to add a tape to the DPM library.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> Add-DPMTape -DPMLibrary $DPMLib
```

Related topics

Get-DPMLibrary

Get-DPMTape

Remove-DPMTape

Set-DPMTape

Connect-DPMServer

Connect-DPMServer

Opens a connection to a DPM server.

Syntax

```
Parameter Set: DPMServerName
Connect-DPMServer [-DPMServerName] <String> [-AsyncOperation <AsyncOperation> ] [
<CommonParameters>]

Parameter Set: DPMServerScope
Connect-DPMServer [-DPMServerScope] <DpmServerScope> [-AsyncOperation <AsyncOperation> ] [
<CommonParameters>]
```

Detailed Description

The **Connect-DPMServer** cmdlet opens a connection to a System Center 2012 – Data Protection Manager (DPM) server.

By default, the cmdlet looks for a DPM server in the current domain. To connect to a server in another domain, specify the domain name.

Parameters

-AsyncOperation<AsyncOperation>

Specifies an **AsyncOperation** object. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerScope<DpmServerScope>

Specifies a DPM server scope object. To create a DPM scope object, use the **New-DPMServerScope** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **DpmServer**

Notes

- For more information, type "`Get-Help Connect-DPMServer -detailed`".

Examples

Example 1: Connect to a server

This command connects to a DPM server named DPMServer07 in the Corporate.Contoso.com domain.

```
PS C:\> Connect-DPMServer -DPMServerName "DPMServer07.Corporate.Contoso.com"
```

Related topics

Disconnect-DPMServer

New-DPMServerScope

Copy-DPMTapeData

Copy-DPMTapeData

Copies the recovery point data from a tape.

Syntax

```
Parameter Set: CopyToMedia
Copy-DPMTapeData [-RecoveryPoint] <RecoverySource> -SourceLibrary <Library> -TapeLabel
<String> -TapeOption <TapeOptions> -TargetLibrary <Library> [-AdhocJobsContext
<AdhocJobsContext> ] [-JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-
RecoveryPointLocation <RecoverySourceLocation> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: DumpMediaHeadlessDataset
Copy-DPMTapeData -IncompleteDataset <HeadlessDataset> -OverwriteType {Overwrite |
NoOverwrite | CopyOnExist} -Tape <Media> -TargetPath <String> -TargetServer <String> [-
DPMServerName <String> ] [-JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-
RecoveryNotification <NotificationObject> ] [-RecreateReparsePoint] [-Restore] [-
RestoreSecurity] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: DumpMediaRecoveryPoint
Copy-DPMTapeData [-RecoveryPoint] <RecoverySource> -OverwriteType {Overwrite | NoOverwrite |
CopyOnExist} -Tape <Media> -TargetPath <String> -TargetServer <String> [-AdhocJobsContext
<AdhocJobsContext> ] [-JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-
RecoveryNotification <NotificationObject> ] [-RecoveryPointLocation <RecoverySourceLocation>
] [-RecreateReparsePoint] [-Restore] [-RestoreSecurity] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Copy-DPMTapeData** cmdlet copies the recovery point data from a tape in System Center 2012 – Data Protection Manager (DPM).

Parameters

-AdhocJobsContext<AdhocJobsContext>

Specifies the context details of an ad hoc job. Do not use this parameter from Windows PowerShell.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IncompleteDataset<HeadlessDataset>

Indicates that the operation occurs only on the portion of the data present on the current tape. In some situations, data spans multiple tapes.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation to the user.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-OverwriteType<OverwriteType>

Specifies the action to take when recovering an existing file. Acceptable values for this parameter are:

--CreateCopy

--Skip

--Overwrite.

The acceptable values for this parameter are:

Overwrite	
NoOverwrite	
CopyOnExist	

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryNotification<NotificationObject>

Specifies that the recovery operation send a notification when the recovery operation completes. The **New-NotificationObject** cmdlet returns the notification object.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryPoint<RecoverySource>

Specifies a recovery point.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-RecoveryPointLocation<RecoverySourceLocation>

Specifies the location of a recovery point. A recovery item may exist in more than one location for the same point in time, such as on a disk and tape, or on two separate tapes.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-RecreateReparsePoint

Indicates that the cmdlet recreates the reparse point.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Restore

Indicates that the cmdlet performs a restore operation.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RestoreSecurity

Indicates that the cmdlet uses the security settings of the recovery point during a restore operation. When you specify the *RestoreSecurity* parameter, the cmdlet uses the security settings from the recovery point. Otherwise, the cmdlet uses the security settings of the destination server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SourceLibrary<Library>

Specifies the location of a dataset to copy.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media>

Specifies a tape object.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-TapeLabel<String>

Specifies a label that identifies the tape.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TapeOption<TapeOptions>

Specifies that the copy operation compress or encrypt the data on the tape. Acceptable values for this parameter are:

-- 0 Compress

-- 1 Encrypt

-- 2 NoCompressAndNoEncrypt

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TargetLibrary<Library>

Specifies the target library to which to copy the data set.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TargetPath<String>

Specifies a target path.

Aliases	none
Required?	true
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TargetServer<String>

Specifies a recovery server.

When you use the **New-RecoveryOption** cmdlet in a clustered environment, format the target server parameter as ResourceGroupName.ClusterName.DomainName.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Copy-DPMTapeData -detailed`".

Examples

Example 1:

```
PS C:\>
```

Related topics

Test-DPMTapeData

Disable-DPMLibrary

Disable-DPMLibrary

Disables DPM libraries.

Syntax

```
Parameter Set: Default
Disable-DPMLibrary [-DPMLibrary] <Library[]> [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Disable-DPMLibrary** cmdlet disables one or more System Center 2012 – Data Protection Manager (DPM) libraries. You can use this cmdlet to disable a tape library in order to perform maintenance or repairs, and then use the **Enable-DPMLibrary** cmdlet to enable it for use.

Parameters

-DPMLibrary<Library[]>

Specifies an array of DPM library objects. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Library**

Notes

- For more information, type "`Get-Help Disable-DPMLibrary -detailed`".

Examples

Example 1: Disable libraries for a named server

This example disables the libraries associated with the server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the tape libraries for the specified server, and stores those objects in the $DPMLib variable.

The second command disables the libraries stored in the $DPMLib variable.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Disable-DPMLibrary -DPMLibrary $DPMLib
```

Related topics

Enable-DPMLibrary

Get-DPMLibrary

Rename-DPMLibrary

Disable-DPMProductionServer

Disable-DPMProductionServer

Disables a DPM protection agent.

Syntax

```
Parameter Set: Default
Disable-DPMProductionServer [-ProductionServer] <ProductionServer> [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Disable-DPMProductionServer** cmdlet disables the System Center 2012 – Data Protection Manager (DPM) protection agent on a computer. After you disable a DPM protection agent, backup jobs for the computer do not run.

Parameters

-ProductionServer<ProductionServer>

Specifies a computer on which a DPM protection agent is installed.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **ProductionServer**

Notes

- For more information, type "`Get-Help Disable-DPMProductionServer -detailed`".

Examples

Example 1: Disable a protection agent

This example disables the DPM protection agent on a computer.

The first command gets the DPM protection agent on the computer named dist01.contoso.com that is protected by the DPM server named DpmWest01. The command stores the result in the $Dpms variable.

The second command disables the protection agent stored in $Dpms.

```
PS C:\> $Dpms = Get-DPMProductionServer -DPMServerName "DpmWest01" | Where {$_.Name -eq
"dist01.contoso.com"}
PS C:\> Disable-DPMProductionServer -ProductionServer $Dpms
```

Related topics

Get-DPMProductionServer

Enable-DPMProductionServer

Update-DPMProductionServer

Get-DPMProductionServer

Disable-DPMTapeDrive

Disable-DPMTapeDrive

Disables a tape drive in the DPM library.

Syntax

```
Parameter Set: Default
Disable-DPMTapeDrive [-TapeDrive] <Drive[]> [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Disable-DPMTapeDrive** cmdlet disables a tape drive in the System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-PassThru

Indicates that the cmdlet is part of a pipeline. Use this parameter with DPM commands to return a related object in cases where there is no default output.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TapeDrive<Drive[]>

Specifies an array of tape drives.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Disable-DPMTapeDrive -detailed`".

Examples

Example 1: Disable a tape drive in a library

This example disables a tape drive on the server Contoso-DPMServer.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library, and stores the result in the $DPMLib variable.

The second command uses the **Get-Tape** cmdlet to retrieve a list of the tapes in the library, and stores the result in the $TapeDrive variable.

The third command uses the **Enable-DPMTapeDrive** cmdlet to disable the tape drive.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $TapeDrive = Get-TapeDrive -DPMLibrary $DPMLib
PS C:\> Disable-DPMTapeDrive -TapeDrive $TapeDrive
```

Related topics

Enable-DPMTapeDrive

Get-DPMLibrary

Get-DPMTapeDrive

Disconnect-DPMServer

Disconnect-DPMServer

Closes a DPM connection session.

Syntax

```
Parameter Set: Default
Disconnect-DPMServer [[-DPMServerName] <String> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Disconnect-DPMServer** cmdlet closes a System Center 2012 – Data Protection Manager (DPM) connection session and releases all objects for that session. Specify a connection to close by using the name of a DPM server.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Disconnect-DPMServer -detailed`".

Examples

Example 1: Disconnect a server

This command closes a connection to a DPM server named DPMServer07.

```
PS C:\> Disconnect-DPMServer -DPMServerName "DPMServer07"
```

Related topics

Connect-DPMServer

Edit-DPMDiskAllocation

Edit-DPMDiskAllocation

Modifies disk allocation for a protected data source on a DPM server.

Syntax

```
Parameter Set: ConnectedDatasource
Edit-DPMDiskAllocation [-Datasource] <Datasource> [-ReplicaSize <Int64> ] [-ShadowCopySize
<Int64> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: DisconnectedClient
Edit-DPMDiskAllocation [-Datasource] <Datasource> [-ExpectedDataSizePerClientInMB <Int64> ]
[-ShadowCopySize <Int64> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Edit-DPMDiskAllocation** cmdlet modifies disk allocation on a System Center 2012 – Data Protection Manager (DPM) server for a protected data source.

Parameters

-Datasource<Datasource>

Specifies a data source. A data source is a volume on a disk, a share, a folder, or a file, that belongs to a data protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-ExpectedDataSizePerClientInMB<Int64>

Specifies a new value, in MB, for the expected size of data to be protected per client computer.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ReplicaSize<Int64>

Specifies the new size of a replica volume, in bytes.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ShadowCopySize<Int64>

Specifies the new size of a recovery point volume, in bytes.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Edit-DPMDiskAllocation -detailed`".

Examples

Example 1: Modify the size of the replica and recovery point volumes

This example modifies the replica volume size to 5 GB and recovery point volume size to 3 GB.

The first command uses the **Get-ProtectionGroup** cmdlet to retrieve the protection group object, and stores the result in the $PGroup variable.

The second command uses the **Get-Datasource** cmdlet to retrieve all data source objects for the protection group stored in $PSGroup, and then stores the objects in the $DSource variable.

The last command uses the **Edit-DPMDiskAllocation** cmdlet to modify the size of the replica volume and the shadow copy volume for the second data source object stored in the $DSource array.

```
PS C:\> $PGroup = Get-ProtectionGroup -DPMServerName "Contoso-DPMServer"
PS C:\> $DSource = Get-Datasource -ProtectionGroup $PGroup
PS C:\> Edit-DPMDiskAllocation -Datasource $DSource[1] -ReplicaSize 5368709120 -
ShadowCopySize 3221225472
```

Enable-DPMLibrary

Enable-DPMLibrary

Enables DPM libraries.

Syntax

```
Parameter Set: Default
Enable-DPMLibrary [-DPMLibrary] <Library[]> [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Enable-DPMLibrary** cmdlet enables one or more System Center 2012 – Data Protection Manager (DPM) libraries. You can use the **Disable-DPMLibrary** cmdlet to disable a tape library in order to perform maintenance or repairs, and then use this cmdlet to enable it for use.

Parameters

-DPMLibrary<Library[]>

Specifies an array of DPM library objects. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Library**

Notes

- For more information, type "`Get-Help Enable-DPMLibrary -detailed`".

Examples

Example 1: Enable libraries for a named server

This example enables the libraries associated with the server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the tape libraries for the specified server, and stores those objects in the $DPMLib variable.

The second command enables the libraries stored in the $DPMLib variable.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Enable-DPMLibrary -DPMLibrary $DPMLib
```

Related topics

Disable-DPMLibrary

Get-DPMLibrary

Rename-DPMLibrary

Enable-DPMProductionServer

Enable-DPMProductionServer

Enables a DPM protection agent.

Syntax

```
Parameter Set: Default
Enable-DPMProductionServer [-ProductionServer] <ProductionServer> [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Enable-DPMProductionServer** cmdlet enables a System Center 2012 – Data Protection Manager (DPM) protection agent on a computer. After you enable a protection agent, DPM enables backup jobs that you schedule for the computer to run.

Parameters

-ProductionServer<ProductionServer>

Specifies a computer on which a DPM protection agent is installed.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- **ProductionServer**

Notes

- For more information, type "`Get-Help Enable-DPMProductionServer -detailed`".

Examples

Example 1: Enable a protection agent

This example enables a DPM protection agent on a computer.

The first command gets the DPM protection agent on the computer named dist01.contoso.com that is protected by the DPM server named DpmWest01. The command stores the result in the $Dpms variable.

The second command disables the protection agent stored in $Dpms.

```
PS C:\> $Dpms = Get-DPMProductionServer -DPMServerName "DpmWest01" | Where {$_.Name -eq
"dist01.contoso.com"}
PS C:\> Enable-DPMProductionServer -ProductionServer $Dpms
```

Related topics

Get-DPMProductionServer

Disable-DPMProductionServer

Update-DPMProductionServer

Get-DPMProductionServer

Enable-DPMTapeDrive

Enable-DPMTapeDrive

Enables the tape drives in the DPM library.

Syntax

```
Parameter Set: Default
Enable-DPMTapeDrive [-TapeDrive] <Drive[]> [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Enable-DPMTapeDrive** cmdlet enables the tape drives in the System Center 2012 – Data Protection Manager (DPM) library.

To determine if a tape drive is enabled, use the **Get-DPMTapeDrive** cmdlet.

Parameters

-PassThru

Indicates that the cmdlet is part of a pipeline. Use this parameter with DPM commands to return a related object in cases where there is no default output.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TapeDrive<Drive[]>

Specifies an array of tape drives.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Enable-DPMTapeDrive -detailed`".

Examples

Example 1: Enable a tape drive in a library

This example enables the tape drive in a library attached to Contoso-DPMServer.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library, and stores the result in the variable named $DPMLib.

The second command uses the **Get-Tape** cmdlet to retrieve the list of the tapes in the library, and stores the result in the variable named $TapeDrive.

The third command uses the **Enable-DPMTapeDrive** cmdlet to enable the tape drive.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $TapeDrive = Get-TapeDrive -DPMLibrary $DPMLib
PS C:\> Enable-DPMTapeDrive -TapeDrive $TapeDrive
```

Related topics

Disable-DPMTapeDrive

Get-DPMLibrary

Get-DPMTapeDrive

Get-DPMAccessLicense

Get-DPMAccessLicense

Gets licensing information for a DPM server and protected computers.

Syntax

```
Parameter Set: License
Get-DPMAccessLicense [[-License]] [[-DPMServerName] <String> ] [ <CommonParameters>]

Parameter Set: LicenseName
Get-DPMAccessLicense [-LicenseName] {SML | CML} [[-Period] <Int32> ] [[-DPMServerName]
<String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMAccessLicense** cmdlet gets licensing information for a System Center 2012 – Data Protection Manager (DPM) server and protected computers.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-License

Indicates that the cmdlet displays the following license information:

-- Name of the product
-- Name of the license
-- Type of license
-- Tabulation method

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-LicenseName<LicenseNames>

Specifies a type of license. The acceptable values for this parameter are:
-- SML. All computers managed by DPM that run a server operating system.
-- CML. All computers managed by DPM that run a client operating system.
The acceptable values for this parameter are:

SML	
CML	

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Period<Int32>

Specifies a period from zero (0) to seven (7). The cmdlet displays licenses used between the current day and (Period – n - 90). If you do not provide a value for this parameter, the cmdlet uses the value 0.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Get licenses for computers running a server operating system

This command gets licenses for computers running server operating systems that a DPM server named DPMServer07 manages.

```
PS C:\> Get-DPMAccessLicense -LicenseName SML -DPMServerName "DPMServer07"
```

Example 2: Get licenses for computers running a server operating system for the last five days

This command gets all active licenses for last five days for computers running server operating systems that a DPM server named DPMServer07 manages.

```
PS C:\> Get-DPMAccessLicense -LicenseName SML -Period 5 -DPMServerName "DPMServer07"
```

Get-DPMAlert

Get-DPMAlert

Gets alerts for a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMAlert [[-DPMServerName] <String> ] [-Async] [-IncludeAlerts {AllActive | FromContext}
] [-IncludeInactiveAlerts] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMAlert** cmdlet gets alerts for a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-IncludeAlerts<IncludeAlertsType>

Specifies which alert to display. The acceptable values for this parameter are:

-- AllActive

-- FromContext

The acceptable values for this parameter are:

AllActive	
FromContext	

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IncludeInactiveAlerts

Indicates that the cmdlet displays inactive alerts.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Inputs

The input type is the type of the objects that you can pipe to the cmdlet.

- DPMServerNameSpecifies the name of a DPM server.

Notes

- For more information, type "`Get-Help Get-DPMAlert -detailed`".

Examples

Example 1: Get all active alerts for a specified server

This command gets alerts for the DPM server named DPMServer073. The command includes all active alerts.

```
PS C:\> Get-DPMAlert -DPMServerName DPMServer073 -IncudeAlerts AllActive
```

Get-DPMAutoProtectIntent

Get-DPMAutoProtectIntent

Gets the auto-protection setting for a SQL Server instance.

Syntax

```
Parameter Set: DPMServer
Get-DPMAutoProtectIntent [-SQLInstanceName] <String> [[-DPMServerName] <String> ] [
<CommonParameters>]

Parameter Set: ProtectionGroup
Get-DPMAutoProtectIntent [-SQLInstanceName] <String> [-ProtectionGroup] <ProtectionGroup> [
<CommonParameters>]
```

Detailed Description

The **Get-DPMAutoProtectIntent** cmdlet gets the System Center 2012 – Data Protection Manager (DPM) auto-protection setting for a Microsoft SQL Server data management software instance. When you use auto-protection for a SQL Server instance, DPM identifies and protects new databases added to that instance. To see the current settings for auto-protection, specify the SQL Server instance, and either the protection group that it belongs to or the name of a DPM server.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-SQLInstanceName<String>

Specifies the name of an SQL Server instance.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMAutoProtectIntent -detailed`".

Related topics

Set-DPMAutoProtectIntent

Start-DPMAutoProtection

Get-DPMBackupNetworkAddress

Get-DPMBackupNetworkAddress

Gets addresses of backup networks for a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMBackupNetworkAddress [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMBackupNetworkAddress** cmdlet gets addresses of backup networks for a System Center 2012 – Data Protection Manager (DPM) server. DPM allows you to configure backup network addresses so that DPM backups do not slow down your primary network.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMBackupNetworkAddress -detailed`".

Examples

Example 1: Get a backup network

This command gets the backup network or networks defined for a DPM server named DPMServer07.

```
PS C:\> Get-DPMBackupNetworkAddress -DpmServerName "DPMServer07"
```

Related topics

Add-DPMBackupNetworkAddress

Remove-DPMBackupNetworkAddress

Get-DPMChildDatasource

Get-DPMChildDatasource

Returns the protectable file system objects within a data source.

Syntax

```
Parameter Set: Default
Get-DPMChildDatasource [-ChildDatasource] <ProtectableObject> [[-ProtectionGroup]
<ProtectionGroup> ] [-Async] [-Inquire] [-Tag <Object> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMChildDatasource** cmdlet returns the protectable file system objects, such as folders, that are within a data source, such as a file system volume. You can protect file systems at the child data source level, but you can protect applications only at the data source level.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ChildDatasource<ProtectableObject>

Specifies a data source, such as a folder in a file system, that you can protect individually.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Inquire

Indicates that the cmdlet queries the protected computer and returns the list of data sources or child data sources on it.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Tag<Object>

Specifies an object that helps distinguish the replies to each asynchronous call that a cmdlet makes. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ChildDatasource**

Notes

- For more information, type "`Get-Help Get-DPMChildDatasource -detailed`".

Examples

Example 1: Get a child data source

This example retrieves the child data sources on a protectable computer.

The first command gets all protection groups from the DPM server named DPMServer07 and stores these groups in the $Pg variable. You cannot edit these protection groups.

The second command gets the data source for the list of protection groups in $Pg and stores this data source in the $Ds variable.

The third command gets a list of the child data sources from element 1 of the $Ds array variable. The command uses the *Inquire* parameter, therefore, the command queries the protected computer.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer07"
PS C:\>$Ds = Get-Datasource -ProtectionGroup $Pg
PS C:\>Get-DPMChildDatasource -ChildDatasource $Ds[1] -Inquire
```

Related topics

Add-DPMChildDatasource

Remove-DPMChildDatasource

Get-DPMCloudSubscription

Get-DPMCloudSubscription

Gets a Windows Azure Online Backup subscription object.

Syntax

```
Parameter Set: Default
Get-DPMCloudSubscription [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMCloudSubscription** cmdlet gets a Windows Azure Online Backup subscription object.

Parameters

-DPMServerName<String>

Specifies the name of the Data Protection Manager (DPM) server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Return status of a Windows Azure Online Backup subscription

This command returns the Windows Azure Online Backup subscription status for the DPM server named TestingServer.

```
PS C:\> Get-DPMCloudSubscription –DPMServerName "TestingServer"
```

Related topics

Set-DPMCloudSubscriptionSetting

Get-DPMCloudSubscriptionSetting

Get-DPMCloudSubscriptionSetting

Returns configuration settings for a Windows Azure Online Backup subscription.

Syntax

```
Parameter Set: Default
Get-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMCloudSubscriptionSetting** cmdlet returns configuration settings for a Windows Azure Online Backup subscription.

Parameters

-DPMServerName<String>

Specifies the name of the System Center 2012 – Data Protection Manager (DPM) server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Get configuration settings for a Windows Azure Online Backup subscription

This command returns configuration settings for the Windows Azure Online Backup subscription on the DPM server named TestingServer.

```
PS C:\> $Setting = Get-DPMCloudSubscriptionSetting -DPMServerName "TestingServer"
```

Related topics

Get-DPMCloudSubscription

Set-DPMCloudSubscriptionSetting

Get-DPMDatasetStatus

Get-DPMDatasetStatus

Returns status of datasets on an archive tape.

Syntax

```
Parameter Set: Default
Get-DPMDatasetStatus [-Tape] <Media> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMDatasetStatus** cmdlet returns the status of datasets on an archive tape.

Parameters

-Tape<Media>

Specifies a **Tape** object. To obtain a **Tape** object, use the Get-Tape cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMDatasetStatus -detailed` ".

Examples

Example 1: Get dataset status for an archive tape

This example returns the status of datasets on an archive tape.

The first command gets the protection groups from the System Center 2012 – Data Protection Manager (DPM) server named TestingServer and stores these protection groups in the variable named $Pg.

The second command gets the tapes associated with the protection groups in $Pg and stores the result in the variable named $Pt.

The third command returns the status of datasets on the tapes associated with the protection groups in $Pg.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "TestingServer"
PS C:\> $Pt = Get-Tape -ProtectionGroup $Pg
PS C:\> Get-DPMDatasetStatus -Tape $Pt
```

Get-DPMDatasource

Get-DPMDatasource

Retrieves the list of protected and unprotected data in a computer or protection group.

Syntax

```
Parameter Set: DpmServer
Get-DPMDatasource [[-DPMServerName] <String> ] [-Inactive] [ <CommonParameters>]

Parameter Set: ClientProtectionGroup
Get-DPMDatasource [[-DPMServerName] <String> ] [-ComputerName] <String[]> [-Async] [
<CommonParameters>]

Parameter Set: DataSourceSearch
Get-DPMDatasource [[-DPMServerName] <String> ] [-Query] <String> [ <CommonParameters>]

Parameter Set: ProductionServer
Get-DPMDatasource [-ProductionServer] <ProductionServer> [-Async] [-
GetVolumesWithoutMountPoints] [-IgnoreDPMInformation] [-Inquire] [-Replica] [-Tag <Object> ]
[ <CommonParameters>]

Parameter Set: ProtectionGroup
Get-DPMDatasource [-ProtectionGroup] <ProtectionGroup> [ <CommonParameters>]

Parameter Set: Search
Get-DPMDatasource [[-DPMServerName] <String> ] [-Path] <String> [[-ProductionServerName]
<String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMDatasource** cmdlet retrieves the list of protected and unprotected data on a computer. The output displays only objects at levels that allow you to apply protection. Use the Get-ChildDatasource cmdlet to see files within the data source.

This cmdlet can also return the following information about the data on the computer:

-- All available data sources on the protected computer.

-- Protected data source on a DPM server

-- Inactive data sources on a System Center 2012 – Data Protection Manager (DPM) server. An inactive data source is one that is not actively protected on a DPM server.

-- Data sources in a protection group

Parameters

-Async

Indicates that the cmdlet runs asynchronously. When you specify this parameter, the cmdlet returns control to you immediately after the operation starts and periodically reports on the progress of the operation. This parameter is useful if the cmdlet takes a long time to complete or if you build a graphical user interface (GUI) by using cmdlets. Do not use this parameter if you are working with the DPM Management Shell.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ComputerName<String[]>

Specifies the list of client computers that you add to the protection group.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of the DPM server.

Aliases	none
Required?	false
Position?	1

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-GetVolumesWithoutMountPoints

Indicates that the cmdlet retrieves volumes without mount points.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IgnoreDPMInformation

Indicates that the cmdlet does not retrieve protected computer information for data sources.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Inactive

Indicates that the cmdlet returns inactive data sources on a DPM server. An inactive data source is one that was protected on the DPM server at one time but is not protected currently. The replicas and recovery points of an inactive data source remain available.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Inquire

Indicates that the cmdlet queries the protected computer and returns the list of data sources or child data sources on it.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Path<String>

Specifies the path on which to search for the data source.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProductionServer<ProductionServer>

Specifies a server that has DPM Protection Agent installed.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-ProductionServerName<String>

Specifies the name of the server that you protect.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies the name of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-Query<String>

Filters the list of data sources and returns only the ones whose names contain the specified string.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Replica

Indicates that the cmdlet calculates the space required for a replica on the secondary DPM server from the protected computer or the primary DPM server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tag<Object>

Specifies a custom property that distinguishes the replies to each asynchronous call.

Do not use this parameter if you work with the DPM Management Shell.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Datasource**

Notes

- For more information, type "`Get-Help Get-DPMDatasource -detailed` ".

Examples

Example 1: Return all data sources on a computer

This example returns a list of all data sources on a computer.

The first command returns a list of servers that have DPM Protection Agent installed, and stores the list in the variable named $Ps.

The second command returns the list of data sources on the second server (at position 1) in the $Ps variable.

```
PS C:\> $Ps = Get-ProductionServer -DPMServerName "TestingServer"
PS C:\> Get-DPMDatasource -ProductionServer $Ps[1] -Inquire
```

Example 2: Return protected data sources in a protection group

This example retrieves the protected data sources in the specified protection group.

The first command returns the protection group from the DPM server named TestingServer and stores the group in the variable named $Pg.

The second command returns the list of data sources from the protection group in $Pg.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "TestingServer"
PS C:\> Get-DPMDatasource -ProtectionGroup $Pg
```

Example 3: Return inactive data sources on a server

This command retrieves the inactive data sources on the DPM server named TestingServer.

```
PS C:\>Get-DPMDatasource -DPMServerName "TestingServer" -Inactive
```

Example 4: Get a data source from a search location

This command retrieves a data source from the search path F:\ on the protected server Test.contoso.com.

```
PS C:\>Get-DPMDatasource -DPMServerName "TestingServer" -SearchPath "F:\" -ProductionServer
"Test.contoso.com"
```

Get-DPMDatasourceProtectionOption

Get-DPMDatasourceProtectionOption

Returns protection options in a protection group.

Syntax

```
Parameter Set: E14Options
Get-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> -E14Options [
<CommonParameters>]

Parameter Set: ExchangeOptions
Get-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> -ExchangeOptions [
<CommonParameters>]

Parameter Set: FileSystem
Get-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> -FileSystem [
<CommonParameters>]
```

Detailed Description

The **Get-DPMDatasourceProtectionOption** cmdlet returns protection options for data sources in a Data Protection Manager (DPM) protection group. You must specify the type of data source in the protection group for which the cmdlet returns the protection options.

Parameters

-E14Options

Indicates that the cmdlet operates on an Exchange Server 2010 data sources.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExchangeOptions

Indicates that the cmdlet operates on Microsoft Exchange data sources.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-FileSystem

Indicates that the cmdlet operates on file system data sources.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies the name of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Datasource**

Examples

Example 1: Get protection options from a server

This example gets protection options from a server.

The first command gets the protection groups from the server named DPMServer02, and stores the result in the $Pg variable.

The second command uses standard array notation to specify the first member of the $Pg array (stored in position 0 of the collection).

```
PS C:\> $pg = Get-DPMProtectionGroup –DPMServerName "DPMServer02"
PS C:\> Get-DPMDatasourceProtectionOption -ProtectionGroup $pg[0] –FileSystem
```

Related topics

Set-DPMDatasourceProtectionOption

Get-DPMDisk

Get-DPMDisk

Retrieves a list of disks in a storage pool on a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMDisk [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMDisk** cmdlet retrieves a list of disks in a storage pool on a System Center 2012 – Data Protection Manager (DPM) server. A storage pool in DPM consists of a set of disks where the DPM server stores replicas, shadow copies, and transfer logs for protected data sources.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Disk**

Notes

- For more information, type "`Get-Help Get-DPMDisk -detailed`".

Examples

Example 1: Return a listing of disks on a DPM server

This command returns a listing of disks on the server named Contoso-DPMServer.

```
PS C:\> Get-DPMDisk -DPMServerName "Contoso-DPMServer"
```

Related topics

Add-DPMDisk

Remove-DPMDisk

Get-DPMGlobalProperty

Get-DPMGlobalProperty

Retrieves the global properties for a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMGlobalProperty [[-DPMServerName] <String> ] [-PropertyName] {AllowLocalDataProtection
| ConsiderForAutoDeployment | ExchangeSCRProtection | HyperVPagefileExclusions |
IsNetworkChecksumRequired | KnownVMMServers | LibraryRefreshInterval |
MaxCapacityForClientAutoDeployment | RegisteredWriters | TruncateSharePointDbLogs} [
<CommonParameters>]
```

Detailed Description

The **Get-DPMGlobalProperty** cmdlet retrieves the global properties for a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PropertyName<GlobalProperties>

Specifies a property to retrieve. Acceptable values for this parameter are:

-- IsNetworkChecksumRequired

-- TruncateSharePointDbLogs

-- LibraryRefreshInterval

-- OptimizeTapeUsage

-- TapeWritePeriodRatio

-- ExchangeSCRProtection

-- AllowLocalDataProtection

The acceptable values for this parameter are:

AllowLocalDataProtection	
ConsiderForAutoDeployment	
ExchangeSCRProtection	
HyperVPagefileExclusions	
IsNetworkChecksumRequired	
KnownVMMServers	
LibraryRefreshInterval	
MaxCapacityForClientAutoDeployment	
RegisteredWriters	
TruncateSharePointDbLogs	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type: "`Get-Help Get-DPMGlobalProperty -detailed`".

Examples

Example 1: Retrieve global properties

This command uses the **Get-DPMGlobalProperty** cmdlet to retrieve the global property for local data protection on a server.

```
PS C:\> Get-DPMGlobalProperty -PropertyName AllowLocalDataProtection
```

Related topics

Set-DPMGlobalProperty

Get-DPMHeadlessDataset

Get-DPMHeadlessDataset

Returns incomplete datasets on the archive tape.

Syntax

```
Parameter Set: Default
Get-DPMHeadlessDataset [-Tape] <Media[]> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMHeadlessDataset** cmdlet returns incomplete datasets on the archive tape.

Parameters

-Tape<Media[]>

Specifies an array of tape objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMHeadlessDataset -detailed`".

Examples

Example 1: Return an incomplete dataset

This example returns an incomplete dataset on the tape.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library, and stores the result in the variable named $DPMLib.

The second command uses the **Get-DPMTape** cmdlet to get a list of tapes in the library, and stores the result in the variable named $Tape.

The third command uses the **Get-DPMHeadlessDataset** cmdlet to retrieve the dataset on the third tape.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $Tape = Get-Tape -DPMLibrary $DPMLib
PS C:\> Get-DPMHeadlessDataset -Tape $Tape[2]
```

Get-DPMJob

Get-DPMJob

Retrieves a list of current and previous jobs on a DPM server.

Syntax

```
Parameter Set: GetJobsDPM
Get-DPMJob [[-DPMServerName] <String> ] [[-Type] {Recovery | InitialReplication | Validation
| ShadowCopy | Replication | MediaErase | DriveCleaning | DetailedInventory | Verification |
DatasetCopy | ArchiveFromSC | RecoveryTape | LibraryRescan | DumpMedia | MediaRecatalog |
OnlineRecatalog | FastInventory | CopyDataFromVolume | StagingAreaRestore |
SharePointExportAndImport | SharePointCatalog | CloudBackup | CloudToStagingAreaRestore |
StagingAreaToPsRecovery | StagingAreaToPsRestore} ] [[-Status] {Completed | Failed |
InProgress | Scheduled} ] [[-From] <DateTime> ] [[-To] <DateTime> ] [[-Newest]] [[-
AdhocJobs]] [ <CommonParameters>]

Parameter Set: GetJobsDS
Get-DPMJob [-Datasource] <Datasource[]> [[-Type] {Recovery | InitialReplication | Validation
| ShadowCopy | Replication | MediaErase | DriveCleaning | DetailedInventory | Verification |
DatasetCopy | ArchiveFromSC | RecoveryTape | LibraryRescan | DumpMedia | MediaRecatalog |
OnlineRecatalog | FastInventory | CopyDataFromVolume | StagingAreaRestore |
SharePointExportAndImport | SharePointCatalog | CloudBackup | CloudToStagingAreaRestore |
StagingAreaToPsRecovery | StagingAreaToPsRestore} ] [[-Status] {Completed | Failed |
InProgress | Scheduled} ] [[-From] <DateTime> ] [[-To] <DateTime> ] [[-Newest]] [[-
AdhocJobs]] [ <CommonParameters>]

Parameter Set: GetJobsDSName
Get-DPMJob [-DatasourceName] <String[]> [[-DPMServerName] <String> ] [[-Type] {Recovery |
InitialReplication | Validation | ShadowCopy | Replication | MediaErase | DriveCleaning |
DetailedInventory | Verification | DatasetCopy | ArchiveFromSC | RecoveryTape |
LibraryRescan | DumpMedia | MediaRecatalog | OnlineRecatalog | FastInventory |
CopyDataFromVolume | StagingAreaRestore | SharePointExportAndImport | SharePointCatalog |
CloudBackup | CloudToStagingAreaRestore | StagingAreaToPsRecovery | StagingAreaToPsRestore}
] [[-Status] {Completed | Failed | InProgress | Scheduled} ] [[-From] <DateTime> ] [[-To]
<DateTime> ] [[-Newest]] [[-AdhocJobs]] [ <CommonParameters>]

Parameter Set: GetJobsPG
Get-DPMJob [-ProtectionGroup] <ProtectionGroup[]> [[-Type] {Recovery | InitialReplication |
Validation | ShadowCopy | Replication | MediaErase | DriveCleaning | DetailedInventory |
Verification | DatasetCopy | ArchiveFromSC | RecoveryTape | LibraryRescan | DumpMedia |
MediaRecatalog | OnlineRecatalog | FastInventory | CopyDataFromVolume | StagingAreaRestore |
SharePointExportAndImport | SharePointCatalog | CloudBackup | CloudToStagingAreaRestore |
StagingAreaToPsRecovery | StagingAreaToPsRestore} ] [[-Status] {Completed | Failed |
InProgress | Scheduled} ] [[-From] <DateTime> ] [[-To] <DateTime> ] [[-Newest]] [[-
AdhocJobs]] [ <CommonParameters>]
```

```
Parameter Set: GetJobsPS
Get-DPMJob [-ProductionServer] <ProductionServer[]> [[-Type] {Recovery | InitialReplication
| Validation | ShadowCopy | Replication | MediaErase | DriveCleaning | DetailedInventory |
Verification | DatasetCopy | ArchiveFromSC | RecoveryTape | LibraryRescan | DumpMedia |
MediaRecatalog | OnlineRecatalog | FastInventory | CopyDataFromVolume | StagingAreaRestore |
SharePointExportAndImport | SharePointCatalog | CloudBackup | CloudToStagingAreaRestore |
StagingAreaToPsRecovery | StagingAreaToPsRestore} ] [[-Status] {Completed | Failed |
InProgress | Scheduled} ] [[-From] <DateTime> ] [[-To] <DateTime> ] [[-Newest]] [[-
AdhocJobs]] [ <CommonParameters>]

Parameter Set: GetJobsPSName
Get-DPMJob [-ProductionServerName] <String[]> [[-DPMServerName] <String> ] [[-Type]
{Recovery | InitialReplication | Validation | ShadowCopy | Replication | MediaErase |
DriveCleaning | DetailedInventory | Verification | DatasetCopy | ArchiveFromSC |
RecoveryTape | LibraryRescan | DumpMedia | MediaRecatalog | OnlineRecatalog | FastInventory
| CopyDataFromVolume | StagingAreaRestore | SharePointExportAndImport | SharePointCatalog |
CloudBackup | CloudToStagingAreaRestore | StagingAreaToPsRecovery | StagingAreaToPsRestore}
] [[-Status] {Completed | Failed | InProgress | Scheduled} ] [[-From] <DateTime> ] [[-To]
<DateTime> ] [[-Newest]] [[-AdhocJobs]] [ <CommonParameters>]

Parameter Set: UpdateJobsDPM
Get-DPMJob [-Job] <Job[]> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMJob** cmdlet retrieves a list of current and previous jobs on a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-AdhocJobs

Indicates that the cmdlet retrieves only ad hoc jobs, not scheduled jobs.

Aliases	none
Required?	false
Position?	7
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource[]>

Specifies an array of data source objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DatasourceName<String[]>

Specifies an array of names of data sources.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-From<DateTime>

Specifies the starting time in a range. The cmdlet retrieves jobs that fall into this range. Use the *To* parameter to specify the ending time in the range.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Job<Job[]>

Specifies a list of jobs.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Newest

Indicates the last job that ran on the filter objects.

Aliases	none
Required?	false
Position?	6
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProductionServer<ProductionServer[]>

Specifies an array of protected computer objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ProductionServerName<String[]>

Specifies an array of names of protected computers.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup[]>

Specifies an array of protection group objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Status<JobStatusType[]>

Specifies an array of job status types.

The acceptable values for this parameter are:

Completed	
Failed	
InProgress	
Scheduled	

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-To<DateTime>

Specifies the ending time in a range. The cmdlet retrieves jobs that fall into this range. Use the *From* parameter to specify the starting time in the range.

Aliases	none
Required?	false
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<DPMJobType[]>

Specifies an array of DPM job types. Acceptable values for this parameter are:

-- Recovery

-- InitialReplication

-- Validation
-- ShadowCopy
-- Replication
-- MediaErase
-- DriveCleaning
-- DetailedInventory
-- Verification
-- DatasetCopy
-- ArchiveFromSC
-- RecoveryTape
-- LibraryRescan
-- DumpMedia
-- MediaRecatalog
-- OnlineRecatalog
-- FastInventory
-- CopyDataFromVolume
-- StagingAreaRestore
-- SharePointImportAndExport
-- SharePointCatalog
-- CloudBackup
-- CloudToStagingAreaRestore
-- StagingAreaToPsRecovery
-- StagingAreaToPsRestore
The acceptable values for this parameter are:

Recovery	
InitialReplication	
Validation	
ShadowCopy	
Replication	
MediaErase	
DriveCleaning	
DetailedInventory	
Verification	
DatasetCopy	
ArchiveFromSC	

RecoveryTape	
LibraryRescan	
DumpMedia	
MediaRecatalog	
OnlineRecatalog	
FastInventory	
CopyDataFromVolume	
StagingAreaRestore	
SharePointExportAndImport	
SharePointCatalog	
CloudBackup	
CloudToStagingAreaRestore	
StagingAreaToPsRecovery	
StagingAreaToPsRestore	

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Retrieve recent jobs

This command returns the latest jobs run on the DPM server named Contoso-DPMServer.

```
PS C:\> Get-DPMJob -DPMServerName "Contoso-DPMServer" -Newest
```

Example 2: Retrieve recent jobs for a protection group

This example returns a list of recent jobs run on protection groups on the DPM server named Contoso-DPMServer.

The first command uses the **Get-ProtectionGroup** cmdlet to retrieve the protection group, and then stores the result in the variable named $PGroup.

The second command uses the **Get-DPMJob** cmdlet to retrieve the list of recent jobs for the protection group stored in $PGroup.

```
PS C:\> $PGroup = Get-ProtectionGroup –DPMServerName "Contoso-DPMServer"
PS C:\> Get-DPMJob -ProtectionGroup $PGroup
```

Related topics

Restart-DPMJob

Stop-DPMJob

Get-DPMLibrary

Get-DPMLibrary

Gets libraries associated with a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMLibrary [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMLibrary** cmdlet gets the libraries associated with a System Center 2012 – Data Protection Manager (DPM) server. You can use this cmdlet to get libraries to use with other cmdlets, such as the **Enable-DPMLibrary** and **Disable-DPMLibrary** cmdlets, or you can use the **Get-DPMLibrary** cmdlet to view the libraries associated with a DPM server, along with library status.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Library**

Notes

- For more information, type "`Get-Help Get-DPMLibrary -detailed`".

Examples

Example 1: Get libraries associated with a server

This command gets the libraries associated with the DPM server named DPMServer07.

```
PS C:\> Get-DPMLibrary -DPMServerName "DPMServer07"
```

Related topics

Disable-DPMLibrary

Enable-DPMLibrary

Rename-DPMLibrary

Get-DPMMaintenanceJobStartTime

Get-DPMMaintenanceJobStartTime

Gets the start times of DPM maintenance jobs.

Syntax

```
Parameter Set: Default
Get-DPMMaintenanceJobStartTime [[-DPMServerName] <String> ] [-MaintenanceJob]
{CatalogPruning | LibraryInventory} [ <CommonParameters>]
```

Detailed Description

The **Get-DPMMaintenanceJobStartTime** cmdlet returns the start times of maintenance jobs for System Center 2012 – Data Protection Manager (DPM) such as Catalog Pruning and Detailed Inventory.

Parameters

-DPMServerName<String>

Specifies the name of the DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MaintenanceJob<HouseKeepingJobs>

Specifies a maintenance job to be performed on a replica. The acceptable values for this parameter are:

-- CatalogPruning. Removes index entries for expired tapes.

-- DetailedInventory. Identifies new tapes and recognizes tapes DPM has seen before by reading the on-media identifier (OMID) on each tape.

The acceptable values for this parameter are:

CatalogPruning	
LibraryInventory	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **DateTime**

Notes

- For more information, type "`Get-Help Set-MaintenanceJobStartTime -detailed`".

Examples

Example 1: Get start time for catalog pruning

This command returns the time when catalog pruning is scheduled to run on the DPM server named TestServer.

```
PS C:\> Get-DPMMaintenanceJobStartTime -DPMServerName "TestServer" -MaintenanceJob
CatalogPruning
```

Related topics

Set-DPMMaintenanceJobStartTime

Get-DPMModifiableProtectionGroup

Get-DPMModifiableProtectionGroup

Retrieves a DPM protection group in an editable mode.

Syntax

```
Parameter Set: Default
Get-DPMModifiableProtectionGroup [-ProtectionGroup] <ProtectionGroup> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMModifiableProtectionGroup** cmdlet retrieves a System Center 2012 – Data Protection Manager (DPM) protection group in an editable mode.

Once you have made the changes to the protection group, save the changes by using the **Set-DPMProtectionGroup** cmdlet.

Follow this sequence of steps when you work with a modifiable protection group:

-- Run the **Get-DPMProtectionGroup** cmdlet to get the unmodifiable protection group.

-- Run the **Get-DPMModifiableProtectionGroup** cmdlet to get the protection group from the previous step in modifiable form.

-- Perform actions on the protection group.

-- Run the **Set-DPMProtectionGroup** cmdlet to save the changed protection group.

Parameters

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup (Editable)**

Notes

- For more information, type "`Get-Help Get-DPMModifiableProtectionGroup -detailed`".

Examples

Example 1: Get a protection group in a modifiable state

This example gets a DPM protection group in a modifiable state.

The first command gets the protection group from the DPM server named TestingServer and stores the protection group in the variable named $Pg.

The second command gets the protection group from the $Pg variable in a modifiable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer07"
PS C:\> Get-DPMModifiableProtectionGroup -ProtectionGroup $Pg
```

Related topics

Get-DPMModifiableProtectionGroup

Set-DPMProtectionGroup

Get-DPMProtectionGroup

Get-DPMPGSet

Get-DPMPGSet

Returns a list of protection group sets on a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMPGSet [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMPGSet** cmdlet returns a list of System Center 2012 – Data Protection Manager (DPM) Protection Group (PG) sets on a DPM server.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMPGSet -detailed`".

Examples

Example 1: Get PG sets

This command gets the PG sets for the server named DPMServer07.

```
PS C:\> Get-DPMPGSet -DPMServerName "DPMServer07"
```

Related topics

New-DPMPGSet

Remove-DPMPGSet

Update-DPMPGSet

Get-DPMPolicyObjective

Get-DPMPolicyObjective

Return the protection policy for a protection group.

Syntax

```
Parameter Set: LongTerm
Get-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> -LongTerm <LongTermLocation> [
<CommonParameters>]

Parameter Set: ShortTerm
Get-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> -ShortTerm [ <CommonParameters>]
```

Detailed Description

The **Get-DPMPolicyObjective** cmdlet returns the protection policy for a protection group. The cmdlet returns the retention range and protection frequency (synchronization to disk or backup to tape) of the protection group.

Parameters

-LongTerm<LongTermLocation>

Indicates that the protection group uses long-term tape protection. Valid values for this parameter are:

-- Tape

-- Online

-- OnlineAndTape.

Aliases	none
Required?	true
Position?	named
Default Value	Tape
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-ShortTerm

Indicates that the protection group is on disk, on tape, or on neither, if nothing is specified.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **PolicyObjective**

Notes

- For more information, type "`Get-Help Get-DPMPolicyObjective -detailed`".
  For technical information, type "`Get-Help Get-DPMPolicyObjective -full`".

Examples

Example 1: Get a short-term policy objective for a protection group

This example returns the short-term policy objective of a protection group.

The first command gets the protection group from the DPM server named "DPMServer02" and stores the result in the $Pg. variable.

The second command gets the short-term policy objective from the protection group in the $Pg. variable.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Get-DPMPolicyObjective -ProtectionGroup $Pg -ShortTerm
```

Example 2: Get a long-term policy objective for a protection group

This example returns the long-term policy objective of a protection group.

The first command gets the protection group from the DPMserver named "DPMServer02" and stores the protection group in the $Pg variable.

The second command gets the long-term policy objective from the protection group in the $Pg variable.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Get-DPMPolicyObjective -ProtectionGroup $Pg -LongTerm
```

Get-DPMPolicySchedule

Get-DPMPolicySchedule

Returns the schedule for protection jobs.

Syntax

```
Parameter Set: LongTerm
Get-DPMPolicySchedule [-ProtectionGroup] <ProtectionGroup> -LongTerm <LongTermLocation> [
<CommonParameters>]

Parameter Set: OffsetSchedule
Get-DPMPolicySchedule [-ProtectionGroup] <ProtectionGroup> -OffsetSchedule [
<CommonParameters>]

Parameter Set: ShortTerm
Get-DPMPolicySchedule [-ProtectionGroup] <ProtectionGroup> -ShortTerm [ <CommonParameters>]
```

Detailed Description

The **Get-DPMPolicySchedule** cmdlet returns the schedule for protection jobs such as synchronization, recovery point creation (shadow copy), and tape backups.

Parameters

-LongTerm<LongTermLocation>

Specifies the long term protection type for a protection group. Valid values for this parameter are:

-- Tape

-- Online

-- OnlineAndTape

Aliases	none
Required?	true
Position?	named
Default Value	Tape

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-OffsetSchedule

Indicates that the cmdlet returns the schedule for protection jobs that use an offset schedule.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-ShortTerm

Indicates that the cmdlet returns the schedule for short-term disk or short-term tape protection jobs.

Aliases	none
Required?	true
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Schedule**

Notes

- For more information, type "`Get-Help Get-DPMPolicySchedule -detailed`".
  For technical information, type "`Get-Help Get-DPMPolicySchedule -full`".

Examples

Example 1: Get a short-term synchronization schedule for a protection group

This example returns the short-term synchronization schedule of a protection group.

The first command gets the protection group from the DPM server named DPMServer02, and stores the protection group in the $Pg variable.

The second command gets the short-term synchronization schedule from the protection group stored in $Pg.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Get-DPMPolicySchedule -ProtectionGroup $Pg -ShortTerm
```

Example 2: Get a long-term synchronization schedule for a protection group

This example returns the long-term synchronization schedule of a protection group.

The first command gets the protection group from the DPM server named DPMServer02, and stores the protection group in the $Pg variable.

The second command gets the long-term synchronization schedule from the protection group stored in $Pg.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Get-DPMPolicySchedule -ProtectionGroup $Pg -LongTerm
```

Related topics

Set-DPMPolicySchedule

Get-ProtectionGroup

Get-DPMProductionCluster

Get-DPMProductionCluster

Gets a list of all clusters on which the DPM agent is installed.

Syntax

```
Parameter Set: Default
Get-DPMProductionCluster [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMProductionCluster** cmdlet returns a list of all clusters on which the System Center 2012 – Data Protection Manager (DPM) agent is installed.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server. If you do not specify a name, the cmdlet uses the current computer name.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Cluster**

Notes

- For more information, type "`Get-Help Get-DPMProductionCluster -detailed`".

Examples

Example1: Get a list of clusters

This command gets a list of clusters protected by the DPM server named DPMServer02.

```
PS C:\> Get-DPMProductionCluster -DPMServerName "DPMServer02"
```

Related topics

Get-DPMProductionVirtualName

Get-DPMProductionServer

Get-DPMProductionServer

Gets a list of computers on which the DPM protection agent is installed.

Syntax

```
Parameter Set: Default
Get-DPMProductionServer [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMProductionServer** gets a list of computers on which the System Center 2012 – Data Protection Manager (DPM) protection agent is installed.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server. If you do not specify a name, the cmdlet uses the current computer name.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProductionServer**

Notes

- For more information, type "`Get-Help Get-DPMProductionServer -detailed`".

Examples

Example 1: Get a list of computers protected by a DPM server

This command gets the list of computers on which the protection agents is installed and are protected by the DPM server named DpmWest01.

```
PS C:\> Get-DPMProductionServer -DPMServerName "DpmWest01"
```

Related topics

Enable-DPMProductionServer

Disable-DPMProductionServer

Update-DPMProductionServer

Get-DPMProductionVirtualName

Get-DPMProductionVirtualName

Gets the virtual names for a cluster.

Syntax

```
Parameter Set: Sync
Get-DPMProductionVirtualName [-ProductionCluster] <Cluster> [-Async] [-Handler
<EventHandler<VNInquiryeventArgs>> ] [-Tag <Object> ] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMProductionVirtualName** cmdlet gets the virtual names for a cluster on which the System Center 2012 – Data Protection Manager (DPM) protection agent is installed.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Handler<EventHandler<VNInquiryeventArgs>>

Specifies the event handler that the DPM calls when it receives an event.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProductionCluster<Cluster>

Specifies the name of a cluster of computers on which the DPM protection agent is installed.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Tag<Object>

Specifies a custom property that distinguishes the replies to each asynchronous call.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **VirtualName**

Notes

- For more information, type "`Get-Help Get-DPMProductionVirtualName -detailed`".

Examples

Example 1: Get the virtual names for a cluster

This example gets the virtual names for a cluster that is protected by DPM.

The first command gets the list of all clusters that are protected by the DPM named DPMServer02. The command stores the result in the $Pc variable.

The second command gets the virtual name of the clusters stored in the $Pc variable.

```
PS C:\> $Pc = Get-DPMProductionCluster -DPMServerName "DPMServer02"
PS C:\> Get-DPMProductionVirtualName -ProductionCluster $Pc
```

Related topics

Get-ProductionCluster

Get-DPMProductionServer

Get-DPMProtectionGroup

Get-DPMProtectionGroup

Gets the list of protection groups on the DPM server.

Syntax

```
Parameter Set: Default
Get-DPMProtectionGroup [[-DPMServerName] <String> ] [-Async] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMProtectionGroup** cmdlet gets the list of protection groups on the System Center 2012 – Data Protection Manager (DPM) server. A protection group that this cmdlet gets is not modifiable. To modify a protection group, you must follow this cmdlet with the **Get-DPMModifiableProtectionGroup** cmdlet.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server. If you do not specify a name, the cmdlet uses the current computer name.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMProtectionGroup -detailed`".

Examples

Example 1: Get a protection group

This command gets the protection group on the DPM server named DPMServer02.

```
PS C:\> Get-DPMProtectionGroup -DPMServerName "DPMServer02"
```

Related topics

Set-DPMProtectionGroup

New-DPMProtectionGroup

Rename-DPMProtectionGroup

Update-DPMProtectionGroup

Get-DPMModifiableProtectionGroup

Get-DPMProtectionJobStartTime

Get-DPMProtectionJobStartTime

Gets the start time of a protection job.

Syntax

```
Parameter Set: Default
Get-DPMProtectionJobStartTime [-ProtectionGroup] <ProtectionGroup> [-JobType]
<ProtectionJobType> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMProtectionJobStartTime** cmdlet gets the start time of a protection job.

Parameters

-JobType<ProtectionJobType>

Specifies the type of job for which options are being set. The only valid value for this parameter is ConsistencyCheck.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **DateTime**

Notes

- For more information, type "`Get-Help Get-DPMProtectionJobStartTime -detailed`".

Examples

Example 1: Get a job start time

This example gets the job start time for the consistency check on a protection group.

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command gets the job start time for the consistency check on the protection group stored in the $Pg variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Get-DPMProtectionJobStartTime -ProtectionGroup $Pg -JobType ConsistencyCheck
```

Related topics

Get-ProtectionGroup

Set-DPMProtectionJobStartTime

Get-DPMRecoverableItem

Get-DPMRecoverableItem

Gets a list of recoverable items in a recovery point.

Syntax

```
Parameter Set: Browse
Get-DPMRecoverableItem [-RecoverableItem] <RecoverableObject> [-BrowseType] <BrowseType> [-
Async] [-Tag <Object> ] [ <CommonParameters>]

Parameter Set: Search
Get-DPMRecoverableItem [-Datasource] <Datasource> [-SearchOption] <SearchSpecifications> [-
Async] [-Tag <Object> ] [ <CommonParameters>]

Parameter Set: Shares
Get-DPMRecoverableItem -RecoveryPointForShares <RecoverySource> [-Async] [-Tag <Object> ] [
<CommonParameters>]
```

Detailed Description

The **Get-DPMRecoverableItem** cmdlet gets a list of recoverable items in a recovery point. The recoverable items based on source are:

-- File system: Files and folders

-- Microsoft Exchange: Storage groups, databases and mailboxes

-- Microsoft SQL: Databases

-- Microsoft SharePoint: Sites, databases and documents

-- Virtual Machines

-- A DPM server

-- System state of a protected computer

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-BrowseType<BrowseType>

Specifies whether to browse only the parent nodes in a recovery point, or to browse the parent nodes and the child nodes. The acceptable values for this parameter are:

-- Parent

-- Child

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.

-- Microsoft Exchange storage group.

-- Microsoft SharePoint farm.

-- Microsoft Virtual Machine.

-- DPM database.

-- A system state that is a member of a protection group.

Aliases	none
Required?	true

Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoverableItem<RecoverableObject>

Specifies a recoverable item object. This is child item within a recovery point that is recoverable. For example, a Windows file system share or volume, a Microsoft SQL database, a Microsoft Exchange storage group, Microsoft SharePoint, Microsoft Virtual Machine, a Microsoft DPM database, system state or a recovery point.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoveryPointForShares<RecoverySource>

Specifies a recovery point object. To obtain a recovery point object, use the **Get-DPMRecoveryPoint** cmdlet.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-SearchOption<SearchSpecifications>

Specifies the search options. You can use the **New-DPMSearchOption** cmdlet to create an object that specifies search options for recoverable objects.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Tag<Object>

Specifies a custom property that distinguishes the replies to each asynchronous call.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **RecoverableItem**

Notes

- For more information, type "`Get-Help Get-DPMRecoverableItem -detailed`".

Examples

Example 1: Get a recoverable item in a recovery point

This example gets a list of recoverable items in the parent and child nodes of a recovery point.

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the result in the $Ds variable.

The third command gets the recovery point for the protection group stored in the $Ds variable, and stores the result in the $Rp variable.

The fourth command gets the recoverable items in the parent and child nodes of the recovery point stored in $Rp.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg
PS C:\> $Rp = Get-DPMRecoveryPoint -Datasource $Ds
PS C:\> Get-DPMRecoverableItem -RecoverableItem $Rp -BrowseType child
```

Related topics

Get-DPMProtectionGroup

Get-DPMRecoveryPoint

Restore-DPMRecoverableItem

Get-DPMDatasource

Get-DPMRecoveryItem

Get-DPMRecoveryItem

Gets recovery items.

Syntax

```
Parameter Set: Default
Get-DPMRecoveryItem [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [
<CommonParameters>]
```

Detailed Description

The **Get-DPMRecoveryItem** cmdlet gets the recovery items that you have attached to a System Center 2012 – Data Protection Manager (DPM) role.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AmDatasourceType>

Specifies the type of data source that the cmdlet uses. The acceptable values for this parameter are:

-- SqlDatabase

-- SqlInstance

-- Client

The acceptable values for this parameter are:

SqlDatabase	
SqlInstance	
Client	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMRecoveryItem -detailed`".

Related topics

Add-DPMRecoveryItem

Remove-DPMRecoveryItem

Get-DPMRole

Get-DPMRecoveryPoint

Get-DPMRecoveryPoint

Gets recovery points for a data source.

Syntax

```
Parameter Set: Datasource
Get-DPMRecoveryPoint [-Datasource] <Datasource> [-Async] [-Online] [ <CommonParameters>]

Parameter Set: Tape
Get-DPMRecoveryPoint [-Tape] <Media> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMRecoveryPoint** cmdlet gets all available recovery points for a data source.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.
-- Microsoft Exchange storage group.
-- Microsoft SharePoint farm.
-- Microsoft Virtual Machine.
-- DPM database.
-- A system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Online

Specifies that DPM enables online protection.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media>

Specifies a tape object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)

Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **RecoveryPoint**

Notes

- For more information, type "`Get-Help Get-DPMRecoveryPoint -detailed`".

Examples

Example 1: Get a recovery point

This example gets the recovery point for a data source.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command gets the recovery point for the data source stored in the $Ds variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg
PS C:\> Get-DPMRecoveryPoint -Datasource $Ds
```

Related topics

Get-DPMProtectionGroup

Get-DPMDatasource

New-DPMRecoveryPoint

Remove-DPMRecoveryPoint

Get-DPMRecoveryPointLocation

Get-DPMRecoveryPointLocation

Get-DPMRecoveryPointLocation

Gets the location of a recovery point.

Syntax

```
Parameter Set: Default
Get-DPMRecoveryPointLocation [-RecoveryPoint] <RecoverySource> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMRecoveryPointLocation** cmdlet gets the location of a recovery point. This cmdlet returns an object that indicates whether the recovery point is located on disk or tape. If the recovery point is located on disk, the cmdlet returns information about the tape.

Parameters

-RecoveryPoint<RecoverySource>

Specifies a recovery point object. To obtain a recovery point location object, use the **Get-DPMRecoveryPoint** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **RecoveryPointLocation**

Notes

- For more information, type "`Get-Help Get-DPMRecoveryPointLocation -detailed`".

Examples

Example 1: Get a recovery point location

This example gets the recovery point location for a recovery point.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command gets the recovery point for the data source stored in the $Ds variable, and stores the result in the $Rp variable.

The fourth command gets the recovery point location for the recovery point stored in the $Rp variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"

PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg

PS C:\> $Rp = Get-DPMRecoveryPoint -Datasource $Ds

PS C:\> Get-DPMRecoveryPointLocation -RecoveryPoint $Rp
```

Related topics

Get-DPMRecoveryPoint

Remove-DPMRecoveryPoint

New-DPMRecoveryPoint

Get-DPMRecoveryTarget

Get-DPMRecoveryTarget

Gets a recovery target.

Syntax

```
Parameter Set: Default
Get-DPMRecoveryTarget [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [
<CommonParameters>]
```

Detailed Description

The **Get-DPMRecoveryTarget** cmdlet gets a recovery target for a System Center 2012 – Data Protection Manager (DPM) role.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AmDatasourceType>

Specifies the type of data source that the cmdlet uses. The acceptable values for this parameter are:

-- SqlDatabase

-- SqlInstance

-- Client

The acceptable values for this parameter are:

SqlDatabase	
SqlInstance	
Client	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMRecoveryTarget -detailed`".

Examples

Example 1: Get a recovery target for a role

This example gets a recovery target for a DPM role.

The first command gets a DPM role named OpsMgrSQL and stores the result in the $Role variable.

The second command gets the recovery target of the SQLInstance type for the role stored in the $Role variable.

```
PS C:\> $Role = Get-DPMRole -Name "OpsMgrSQL"
PS C:\> Get-DPMRecoveryTarget -DPMRole $Role -Type SQLInstance
```

Related topics

Add-DPMRecoveryTarget

New-DPMRecoveryTarget

Add-DPMRecoveryTarget

Get-DPMReplicaCreationMethod

Get-DPMReplicaCreationMethod

Gets the replica creation method for a protection group.

Syntax

```
Parameter Set: Default
Get-DPMReplicaCreationMethod [-ProtectionGroup] <ProtectionGroup> [-Reserved] [
<CommonParameters>]
```

Detailed Description

The **Get-DPMReplicaCreationMethod** cmdlet gets the replica creation method for a protection group. The replica creation method specifies when System Center 2012 – Data Protection Manager (DPM) initiates the creation of a replica.

You can change the replica creation method by using the **Set-DPMReplicaCreationMethod** cmdlet.

Parameters

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Reserved

Specifies an internal parameter. Do not use.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ReplicaCreationMethod**

Notes

- For more information, type "`Get-Help Get-DPMReplicaCreationMethod -detailed`".

Examples

Example 1: Get the replica creation method

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command gets the replica creation method for the protection group stored in the $Mpg variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"

PS C:\> Get-DPMReplicaCreationMethod -ProtectionGroup $Pg
```

Related topics

Set-DPMReplicaCreationMethod

Get-DPMProtectionGroup

Get-DPMRole

Get-DPMRole

Gets a DPM role to view or edit.

Syntax

```
Parameter Set: Default
Get-DPMRole [[-DPMServerName] <String> ] [[-Name] <String> ] [-Editable] [
<CommonParameters>]
```

Detailed Description

The **Get-DPMRole** cmdlet gets a System Center 2012 – Data Protection Manager (DPM) role for you to view or edit. You can use this cmdlet to display the properties of a DPM role, or all the DPM roles for a DPM server. DPM roles allow Microsoft SQL Server database owners to recover databases without assistance from a DPM administrator.

To open a role for editing, use the *Editable* parameter. After you make changes to a DPM role, use the **Set-DPMRole** cmdlet to save those changes.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Editable

Indicates that you can edit the DPM role.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Name<String>

Specify the name of a protection group.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMRole -detailed`".

Examples

Example 1: Get a role

This command gets a DPM role named OpsMgrSQL. If you intend to make changes to the role, specify the *Editable* parameter.

```
PS C:\> Get-DpmRole -Name "OpsMgrSQL"
```

Related topics

New-DPMRole

Remove-DPMRole

Rename-DPMRole

Set-DPMRole

Get-DPMSecurityGroup

Get-DPMSecurityGroup

Gets the security groups for a DPM role.

Syntax

```
Parameter Set: Default
Get-DPMSecurityGroup [-DpmRole] <DpmRole> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMSecurityGroup** cmdlet gets the security groups for a System Center 2012 – Data Protection Manager (DPM) role.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMSecurityGroup -detailed`".

Examples

Example 1: Get security groups for a role

This example gets the DPM security groups for a role.

The first command uses the **Get-DPMRole** cmdlet to get the role named OpsMgrSQL and stores it in the $DpmRole variable.

The second command gets the security groups for the role stored in the $DpmRole variable.

```
PS C:\> $DpmRole = Get-DpmRole -Name "OpsMgrSQL"
PS C:\> Get-DPMSecurityGroup -DPMRole $DPMRole
```

Related topics

Add-DPMSecurityGroup

Remove-DPMSecurityGroup

Get-DPMRole

Get-DPMTape

Get-DPMTape

Returns a list of tapes in the library.

Syntax

```
Parameter Set: Library
Get-DPMTape [-DPMLibrary] <Library[]> [ <CommonParameters>]

Parameter Set: ProtectionGroup
Get-DPMTape [-ProtectionGroup] <ProtectionGroup[]> [ <CommonParameters>]

Parameter Set: RecoverySourceLocation
Get-DPMTape [-RecoveryPointLocation] <RecoverySourceLocation[]> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMTape** cmdlet returns a list of tapes in the library. In System Center 2012 – Data Protection Manager (DPM), a library refers to multi-drive tape hardware and standalone tape drives.

Parameters

-DPMLibrary<Library[]>

Specifies an array of DPM library objects. The library objects refer to multi-drive tape hardware, as well as standalone tape drives.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup[]>

Specifies the name of a protection group. A protection group consists of data sources, such as volumes or shares, with a common protection configuration and schedule.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoveryPointLocation<RecoverySourceLocation[]>

Specifies an array of recovery points. A recovery point is a copy of a replica stored on a DPM server. You can specify multiple locations for recovery items that exist on a disk and a tape, or on multiple tapes.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Tape**

Notes

- For more information, type "`Get-Help Get-DPMTape -detailed`".

Examples

Example 1: Retrieve a list of tapes for a library

This example retrieves a list of tapes for a library in DPM.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library for the server, and stores the result in the $DPMLib variable.

The second command uses the **Get-DPMTape** cmdlet to return the list of tapes on Contoso-DPMServer.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> Get-DPMTape -DPMLibrary $DPMLib
```

Related topics

Add-DPMTape

Get-DPMLibrary

Remove-DPMTape

Set-DPMTape

Get-DPMTapeBackupOption

Get-DPMTapeBackupOption

Retrieves library, drive and other backup or archive options for a protection group.

Syntax

```
Parameter Set: Default
Get-DPMTapeBackupOption [-ProtectionGroup] <ProtectionGroup> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMTapeBackupOption** cmdlet retrieves library, drive and other backup or archive options for a System Center 2012 – Data Protection Manager (DPM) protection group.

Parameters

-ProtectionGroup<ProtectionGroup>

Specifies a DPM protection group. A protection group consists of data sources, such as volumes or shares, with a common protection configuration and schedule.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Get-DPMTapeBackupOption -detailed`".

Examples

Example 1: Retrieve tape backup options

This example retrieves the tape backup options for a protection group in DPM.

The first command uses the **Get-ProtectionGroup** cmdlet to retrieve the protection group, and stores the result in the $PGroup variable.

The second command uses the **Get-DPMTapeBackupOption** cmdlet to retrieve the settings.

```
PS C:\> $PGroup = Get-ProtectionGroup -DPMServerName "Contoso-DPMServer"
PS C:\> Get-DPMTapeBackupOption -ProtectionGroup $PGroup
```

Related topics

Set-DPMTapeBackupOption

Get-DPMTapeDrive

Get-DPMTapeDrive

Retrieves a list of tape drives in a library on a DPM server.

Syntax

```
Parameter Set: Default
Get-DPMTapeDrive [-DPMLibrary] <Library[]> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMTapeDrive** cmdlet retrieves a list of tape drives in a library on a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-DPMLibrary<Library[]>

Specifies an array of DPM library objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **TapeDrive**

Notes

- For more information, type "`Get-Help Get-DPMTapeDrive -detailed`".

Examples

Example 1: Retrieve a list of tape drives in a library

This example retrieves a list of the tape drives in all libraries on the server Contoso-DPMServer.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library, and stores the result in the $DPMLib variable.

The second command uses the **Get-DPMTapeDrive** cmdlet to retrieve a list of tape drives.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> Get-DPMTapeDrive -DPMLibrary $DPMLib
```

Related topics

Disable-DPMTapeDrive

Enable-DPMTapeDrive

Get-DPMLibrary

Get-DPMTapeSlot

Get-DPMTapeSlot

Retrieves a list of tape slots in the DPM library.

Syntax

```
Parameter Set: Default
Get-DPMTapeSlot [-DPMLibrary] <Library[]> [ <CommonParameters>]
```

Detailed Description

The **Get-DPMTapeSlot** cmdlet retrieves a list of tape slots in the System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-DPMLibrary<Library[]>

Specifies an array of DPM library objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **SlotNumber**

Notes

- For more information, type "`Get-Help Get-DPMTapeSlot -detailed`".

Examples

Example 1: Retrieve the tape slots in a library

This example retrieves the tape slots in the library.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library for the server named Contoso-DPMServer, and stores the result in the $DPMLib variable.

The second command uses the **Get-DPMTapeSlot** cmdlet to retrieve the list of tape slots for the library.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> Get-DPMTapeSlot -DPMLibrary $DPMLib
```

Related topics

Get-DPMLibrary

Get-DPMVolume

Get-DPMVolume

Retrieves a list of volumes on the DPM server.

Syntax

```
Parameter Set: Default
Get-DPMVolume [[-DPMServerName] <String> ] [-AlreadyInUseByDPM] [ <CommonParameters>]
```

Detailed Description

The **Get-DPMVolume** cmdlet retrieves a list of volumes on the System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-AlreadyInUseByDPM

Indicates whether to filter the list of volumes to include those already in use on the DPM server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN

Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Get-DPMVolume -detailed`".

Examples

Example 1: Retrieve volumes on a DPM server

This command retrieves the DPM volumes on the server named Contoso-DPMServer.

```
PS C:\>Get-DPMVolume -DPMServerName "Contoso-DPMServer"
```

Lock-DPMLibraryDoor

Lock-DPMLibraryDoor

Locks the door of a DPM library.

Syntax

```
Parameter Set: Default
Lock-DPMLibraryDoor [-DPMLibrary] <Library> [-Async] [-DoorAccessJobStateChangeEventHandler
<DoorAccessJobStateChangeEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Lock-DPMLibraryDoor** cmdlet locks the door of a System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByPropertyName)
Accept Wildcard Characters?	false

-DoorAccessJobStateChangeEventHandler<DoorAccessJobStateChangeEventHandler>

Specifies an event handler for certain door access events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByPropertyName)
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Lock-DPMLibraryDoor -detailed`".

Examples

Example 1: Lock a library door

This example locks a library door for a library associated with the server named DPMServer07. The first command gets the libraries associated with the server, and stores the library objects in the $DPMLib variable.

The second command locks the library door for the first library object stored in the $DPMLib variable. This DPM server has more than one library, so the command uses standard Windows PowerShell array syntax to specify the first member of the $DPMLib array.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Lock-DPMLibraryDoor -DPMLibrary $DPMLib[0]
```

Related topics

Get-DPMLibrary

Unlock-DPMLibraryDoor

Lock-DPMLibraryIEPort

Lock-DPMLibraryIEPort

Locks the IE port for a DPM library and loads the media present in the IE port.

Syntax

```
Parameter Set: Default
Lock-DPMLibraryIEPort [-DPMLibrary] <Library> [-Async] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Lock-DPMLibraryIEPort** cmdlet locks the insert/eject (IE) port for a System Center 2012 – Data Protection Manager (DPM) library and loads the media present in the IE port into the slot.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByPropertyName)
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for certain IE port events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Lock-DPMLibraryIEPort -detailed`".

Examples

Example 1: Lock an IE port

This example locks an IE port for a library associated with a server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the library for the named server, and stores the library object in the $DPMLib variable.

The second command locks the object stored in the $DPMLib variable.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Lock-DPMLibraryDoor -DPMLibrary $DPMLib
```

Related topics

Unlock-DPMLibraryIEPort

Get-DPMLibrary

New-DPMPGSet

New-DPMPGSet

Creates a DPM PG set.

Syntax

```
Parameter Set: Default
New-DPMPGSet [[-DPMServerName] <String> ] [-Name] <String> [[-PGList] <ProtectionGroup[]> ]
[[-WritePeriodUnit] <TimeUnit> ] [[-WritePeriodValue] <UInt32> ] [[-ExpiryToleranceUnit]
<TimeUnit> ] [[-ExpiryToleranceValue] <UInt32> ] [-AllowDifferentRetentionPeriods] [
<CommonParameters>]
```

Detailed Description

The **New-DPMPGSet** cmdlet creates a new System Center 2012 – Data Protection Manager (DPM) protection group (PG) set. A DPM PG is a collection of protection groups that you collocate on the same tape.

Parameters

-AllowDifferentRetentionPeriods

Indicates that protection groups with different retention periods can be part of the same PG set.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExpiryToleranceUnit<TimeUnit>

Specifies the measurement unit for expiry tolerance.

Aliases	none
Required?	false
Position?	6
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExpiryToleranceValue<UInt32>

Specifies the maximum length of time for which an expired recovery point can remain on a tape until DPM marks the tape as expired.

Aliases	none
Required?	false
Position?	7
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Name<String>

Specifies the name of the PG set.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PGList<ProtectionGroup[]>

Specifies an array of protection groups to add to the PG set.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-WritePeriodUnit<TimeUnit>

Specifies the measurement unit for the write period.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WritePeriodValue<UInt32>

Specifies the length of time for which a tape is available for writing new backups. DPM marks the tape as offsite ready after this interval.

Aliases	none
Required?	false
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help New-DPMPGSet -detailed`".

Examples

Example 1: Create a PG set

This example creates a protection group set named PGSET2 on the specified DPM server.

The first command uses the **Get-DPMProtectionGroup** cmdlet to get PGs that have names that contain either PG1 or PG2, and then stores them in the $PgList variable.

The second command creates a PS set named PGSET2 that contains the PGs stored in the $PgList variable.

```
PS C:\> $PgList = Get-DPMProtectionGroup -DPMServerName "DPMServer07" | where
{($_.friendlyname) –match "PG1" -or ($_.friendlyname) -match "PG2"}
PS C:\> New-DPMPGSet -Name "PGSET2" -PGList $PgList
```

Related topics

Get-DPMPGSet

Remove-DPMPGSet

Update-DPMPGSet

Get-DPMProtectionGroup

New-DPMProtectionGroup

New-DPMProtectionGroup

Creates a protection group on the DPM server.

Syntax

```
Parameter Set: Default
New-DPMProtectionGroup [[-DPMServerName] <String> ] [[-Name] <String> ] [
<CommonParameters>]
```

Detailed Description

The **New-DPMProtectionGroup** cmdlet creates a protection group on a System Center 2012 – Data Protection Manager (DPM) server. This is the first step in the process of creating a protection group. This cmdlet returns a new protection group object on which you can perform the remaining steps of creating a protection group. However, the protection group is not created until you run the **Set-DPMProtectionGroup** cmdlet.

Use the following cmdlets to create a new protection group. Use one or more cmdlets from each step:

1. **New-DPMProtectionGroup**
2. **Add-DPMChildDatasource**, **Remove-DPMChildDatasource**, **Set-DPMDatasourceProtectionOption**, **Set-DPMProtectionJobStartTime**
3. **Set-DPMProtectionType**
4. **Set-DPMReplicaCreationMethod**
5. **Set-DPMPolicyObjective**, **Set-DPMPolicySchedule** (Short term)
6. **Get-DPMDatasourceDiskAllocation**
7. **Set-DPMDatasourceDiskAllocation**
8. **Set-DPMPolicyObjective**, **Set-DPMPolicySchedule** (Long term)
9. **Set-DPMTapeBackupOption**
10. **Set-DPMProtectionGroup**

Depending on your choice in a previous step, some steps might not be applicable. For example, if you set disk-based protection in step 3, steps 8 and 9 are not required.

It is important to remember that the protection group is created only after you run the **Set-DPMProtectionGroup** cmdlet. Until you run **Set-DPMProtectionGroup**, the protection group exists only in memory.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server. If you do not specify a name, the cmdlet uses the current computer name.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Name<String>

Specifies a name for the protection group.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help New-DPMProtectionGroup -detailed`".

Examples

Example 1: Create a protection group

This command creates an instance of a protection group named ProtectGroup01 in the memory of the DPM server named DPMServer02. This is the first step of creating the protection group.

```
PS C:\> New-DPMProtectionGroup -DPMServerName "DPMServer02" -Name "ProtectGroup01"
```

Related topics

Update-DPMProtectionGroup

Get-DPMProtectionGroup

Rename-DPMProtectionGroup

Set-DPMProtectionGroup

New-DPMRecoveryNotification

New-DPMRecoveryNotification

Creates a notification object.

Syntax

```
Parameter Set: Default
New-DPMRecoveryNotification [-NotificationType] <NotificationType> [-NotificationIdList]
<String[]> [ <CommonParameters>]
```

Detailed Description

The **New-DPMRecoveryNotification** cmdlet creates a notification object that System Center 2012 – Data Protection Manager (DPM) uses for recovery.

Parameters

-NotificationIdList<String[]>

Specifies an array of IDs to which the recovery sends notifications. DPM currently supports only email notifications.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-NotificationType<NotificationType>

Specifies the type of notification. Currently the only valid value is email.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Notification**

Notes

- For more information, type "`Get-Help New-DPMRecoveryNotification -detailed`".

Examples

Example 1: Create a notification object

This command creates an email notification for an event that DPM sends to john@contoso.com.

```
PS C:\> New-DPMRecoveryNotification -NotificationType email -NotificationIdList
"john@contoso.com"
```

Related topics

New-DPMRecoveryOption

New-DPMRecoveryOption

New-DPMRecoveryOption

Creates recovery options.

Syntax

```
Parameter Set: ClientDatasourceRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
ClientDatasource -OverwriteType {Overwrite | NoOverwrite | CopyOnExist} -RecoveryType
{Recover | Restore} [-AlternateLocation <String> ] [-DPMLibrary <Library> ] [-
RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-SANRecovery] [ <CommonParameters>]

Parameter Set: E14RecoveryOption
New-DPMRecoveryOption [-RecoveryLocation] <RecoveryLocation> [-TargetServer] <String> -
E14Datasource -ExchangeOperationType <ExchangeOperationType> -RecoveryType {Recover |
Restore} [-AlternateDatabase <String> ] [-DPMLibrary <Library> ] [-IsRecoveryDatabase] [-
MountDatabaseAfterRestore] [-RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-
RollForwardRecovery] [-SANRecovery] [-TargetLocation <String> ] [ <CommonParameters>]

Parameter Set: ExchangeRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
Exchange -ExchangeOperationType <ExchangeOperationType> -RecoveryType {Recover | Restore} [-
AlternateDatabase <String> ] [-AlternateStorageGroup <String> ] [-DatabaseName <String> ] [-
DPMLibrary <Library> ] [-IsRecoveryStorageGroup] [-MailboxDisplayName <String> ] [-
MountDatabaseAfterRestore] [-RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-
RollForwardRecovery] [-SANRecovery] [-StorageGroupName <String> ] [-TargetLocation <String>
] [ <CommonParameters>]

Parameter Set: FsRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
FileSystem -OverwriteType {Overwrite | NoOverwrite | CopyOnExist} -RecoveryType {Recover |
Restore} [-AlternateLocation <String> ] [-DPMLibrary <Library> ] [-RecoverToReplicaFromTape
<Boolean> ] [-RestoreSecurity] [-SANRecovery] [ <CommonParameters>]

Parameter Set: GenericDatasourceRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
GenericDatasource -RecoveryType {Recover | Restore} [-DPMLibrary <Library> ] [-
RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-SANRecovery] [-TargetLocation
<String> ] [ <CommonParameters>]

Parameter Set: HyperVDatasourceRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
HyperVDatasource -RecoveryType {Recover | Restore} [-DPMLibrary <Library> ] [-
RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-SANRecovery] [-TargetLocation
<String> ] [ <CommonParameters>]
```

```
Parameter Set: RecoverToDpm
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
PrimaryDpmServer [-DPMLibrary <Library> ] [-RecoverToReplicaFromTape <Boolean> ] [
<CommonParameters>]

Parameter Set: SharePointRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
RecoveryType {Recover | Restore} -SharePoint [-DPMLibrary <Library> ] [-
RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-SANRecovery] [-TargetLocation
<String> ] [ <CommonParameters>]

Parameter Set: SharePointSiteRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
DatabaseFileTempLocation <String> -ExportFileTempLocation <String> -IntermediateServer
<String> -IntermediateSqlInstance <String> -RecoveryType {Recover | Restore} -SharePointSite
[-DPMLibrary <Library> ] [-ItemLevelRecoveryType {RecoveryFarm | UnattachedInstance} ] [-
RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-SANRecovery] [-TargetLocation
<String> ] [-TargetSiteUrl <String> ] [ <CommonParameters>]

Parameter Set: SQLRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
RecoveryType {Recover | Restore} -SQL [-AlternateDatabaseDetails
<AlternateDatabaseDetailsType> ] [-CopyLogFiles] [-DPMLibrary <Library> ] [-
LeaveDBInRestoringState] [-LogFileCopyLocation <String> ] [-RecoverToReplicaFromTape
<Boolean> ] [-RestoreSecurity] [-RollForwardRecovery] [-SANRecovery] [-TargetLocation
<String> ] [ <CommonParameters>]

Parameter Set: StagingAreaRestoreOption
New-DPMRecoveryOption [[-DPMServerName] <String> ] [-StagingAreaPath] <String> [-
DestinationServerName] <String> [-DestinationPath] <String> [-DatasourceName] <String> [-
OverwriteType] {Overwrite | NoOverwrite | CopyOnExist} [[-RestoreSecurity]] -
DPMComponentName <Guid> [-DPMLibrary <Library> ] [-FileSystemDatasource] [-
RecoverToReplicaFromTape <Boolean> ] [ <CommonParameters>]

Parameter Set: SystemProtectionRecoveryOption
New-DPMRecoveryOption [-TargetServer] <String> [-RecoveryLocation] <RecoveryLocation> -
RecoveryType {Recover | Restore} -SystemProtectionDatasource [-DPMLibrary <Library> ] [-
RecoverToReplicaFromTape <Boolean> ] [-RestoreSecurity] [-SANRecovery] [-TargetLocation
<String> ] [ <CommonParameters>]
```

Detailed Description

The **New-DPMRecoveryOption** cmdlet creates recovery options for file servers, computers running Exchange Server, SharePoint Server, and SQL Server, and other data sources.

Parameters

-AlternateDatabase<String>

Specifies the name of an alternate database for recovery.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AlternateDatabaseDetails<AlternateDatabaseDetailsType>

Specify the type of the alternate database.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AlternateLocation<String>

Specifies an alternate location where DPM creates a recovery point.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AlternateStorageGroup<String>

Specifies an alternate storage group for a new recovery point.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ClientDatasource

Indicates that the restored data is a file system or an application.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-CopyLogFiles

Indicates that DPM copies the log files.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DatabaseFileTempLocation<String>

Specifies a temporary location for the database file.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DatabaseName<String>

Specify the name of the database.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DatasourceName<String>

Specifies the name of the data source.

Aliases	none
Required?	true
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DestinationPath<String>

Specifies the recovery location.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DestinationServerName<String>

Specifies the name of the server to which DPM recovers data.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMComponentName<Guid>

Specifies the GUID of the data source you are recovering.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server. If you do not specify a name, the cmdlet uses the name of the current computer.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-E14Datasource

Indicates that the data source is Exchange Server 2010.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-Exchange

Indicates that DPM performs the current operation on a Microsoft Exchange data source.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExchangeOperationType<ExchangeOperationType>

Specifies the type of operation that DPM performs on a Microsoft Exchange data source. Valid values are:

-- NoOperation

-- MailBoxLevelRecovery

-- NeedCleanShutdown

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExportFileTempLocation<String>

Specifies the location of the export file.

Aliases	none
Required?	true

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-FileSystem

Indicates that DPM performs the current operation on a file system data source.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-FileSystemDatasource

Indicates that you are recovering a file system data source.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-GenericDatasource

Indicates that DPM performs the current operation on a data source such as Microsoft Virtual Server.

Aliases	none
Required?	true

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-HyperVDatasource

Indicates that the data source is Hyper-V.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IntermediateServer<String>

Specifies the server that DPM uses for SharePoint site recovery.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IntermediateSqlInstance<String>

Specifies a SQL Server instance that DPM uses for SharePoint site recovery. DPM uses the SQL Server instance to temporarily host a SQL Server database.

Aliases	none

Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IsRecoveryDatabase

Indicates that the database is a recovery database.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IsRecoveryStorageGroup

Indicates that the recovery is a recovery storage group.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ItemLevelRecoveryType<SharepointRecoveryType>

Specifies the recovery type. Valid values are:

-- RecoveryFarm

-- UnattachedInstance

The acceptable values for this parameter are:

RecoveryFarm	
UnattachedInstance	

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-LeaveDBInRestoringState

Indicates that DPM leaves the database non-operational but in a restorable state. Specify this parameter to leave the database operational.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-LogFileCopyLocation<String>

Specifies the location where DPM copies log files.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MailboxDisplayName<String>

Specifies a name that DPM displays on the mailbox.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MountDatabaseAfterRestore

Indicates that DPM mounts the database after it restores the database.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-OverwriteType<OverwriteType>

Specifies the action that DPM takes when the file it is recovering already exists. Valid values are:

-- CreateCopy

-- Skip

-- Overwrite

The acceptable values for this parameter are:

Overwrite	
NoOverwrite	
CopyOnExist	

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PrimaryDpmServer

Indicates that the server where DPM recovers data is a DPM server.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoverToReplicaFromTape<Boolean>

Indicates whether recovery to the replica is from a tape.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryLocation<RecoveryLocation>

Specifies the location where DPM recovers data. Valid values are:

-- OriginalServer

-- CopyToFolder

-- OriginalServerWithDBRename

-- AlternateExchangeServer

-- ExchangeServerDatabase

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryType<RecoveryType>

Specifies the recovery type. If you specify the *HyperVDatasource* parameter, the only valid value is Recover. Valid values are: Recover or Restore.

The acceptable values for this parameter are:

Recover	
Restore	

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-RestoreSecurity

Indicates that DPM uses the security settings from the recovery point. If you do not specify this parameter, DPM uses the security settings of the destination server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RollForwardRecovery

Indicates that DPM recovers the application from the latest recovery point and applies all logs after that recovery point to bring the application to the latest state.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SANRecovery

Indicates that DPM performs the current operation on a Storage Area Network (SAN) data source.

Aliases	none
Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SharePoint

Indicates that DPM performs the current operation on a SharePoint data source.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SharePointSite

Indicates that DPM performs the current operation on a SharePoint site.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SQL

Indicates that DPM performs the current operation on a SQL Server data source.

Aliases	none
Required?	true
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-StagingAreaPath<String>

Specifies the path of the staging area.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-StorageGroupName<String>

Specifies the name of the storage group.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SystemProtectionDatasource

Indicates that DPM performs the current operation on a system protection data source.

Aliases	none
Required?	true
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TargetLocation<String>

Specify the location where DPM stores the replica.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TargetServer<String>

Specifies the target server for recovery.

If you use this cmdlet in a clustered environment, specify the target server parameter in the format ResourceGroupName.ClusterName.DomainName.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TargetSiteUrl<String>

Specifies the URL of the target where DPM recovers data.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **RecoveryOption**

Notes

- For more information, type "`Get-Help New-DPMRecoveryOption -detailed`".

Examples

Example 1: Recover a file system

This command creates a recovery option that restores the file system to west.domain.com and restores the original security settings.

```
PS C:\> New-DPMRecoveryOption -TargetServer "west.contoso.com" -RecoveryLocation
"d:\copytofolder" -FileSystem -AlternateLocation "f:\restore" -OverwriteType overwrite -
RestoreSecurity -RecoveryType Restore
```

Example 2: Recover a file system by using a replica

The example outlines the sequence of commands that you can use to recover a file server by using the replica.

The first command opens a connection to a DPM server.

The second command gets the computer named DPMServer01 on which the DPM protection agent is installed. The command stores the result in the $Ps variable.

The third command gets the list of protected and unprotected data on the production server stored in the $Ps variable. The command stores the result in the $Ds variable.

The fourth command gets the libraries associated with the DPM server named DPoreMServer01. The command stores the result in the $Rp variable.

The fifth command gets the recovery point for the data source stored in the $Ds variable, and stores the result in the $Rp variable.

The sixth command creates a recovery option on a file system data source. The command specifies that data is recovered to the replica on the target server named DPMServer02 from a tape.

```
PS C:\> Connect-DPMServer -DPMServerName "DPMServer01"
PS C:\> $Ps = Get-DPMProductionServer -DPMServerName "DPMServer01"
PS C:\> $Ds = Get-DPMDatasource -ProductionServer $Ps
PS C:\> $Dl = Get-DPMLibrary -DPMServerName "DPoreMServer01"
PS C:\> $Rp = Get-DPMRecoveryPoint -Datasource $Ds
PS C:\> $Rop = New-DPMRecoveryOption -Filesystem -RecoverToReplicaFromTape $True -
RecoveryLocation DPMReplicaVolume -DPMLibrary $dl[0] -Targetserver "DPMServer02" -
Overwritetype overwrite
```

Example 3: Recover a file system from a replica on tape

This example

The example outlines the sequence of commands that you can use to recover a file server from a replica on tape to the specified recovery location.

The first command gets the protection group on the DPM server named DPMServer02. The command stores the result in the $Pg variable.

The second command gets the list of protected and unprotected data on the production server stored in the $Ps variable. The command stores the result in the $Ds variable.

The third command gets the recovery point for the data source stored in the $Ds variable, and stores the result in the $Rp variable.

The fourth command gets the libraries associated with the DPM server named DPoreMServer01. The command stores the result in the $Lib variable.

The fifth command creates a recovery option on a file system data source. The command specifies that data is recovered to the replica on the target server named DPMServer02 from a tape.

The sixth command restores a version of the data source stored in the $Rp variable by using the recovery option stored the $Rop variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProductionServer $Pg
PS C:\> $Rp = Get-DPMRecoveryPoint -Datasource $Ds
PS C:\> $Lib = Get-DPMLibrary -DPMServerName "DPoreMServer01"
PS C:\> $Rop = New-DPMRecoveryOption -RecoverToReplicaFromTape $True -RecoveryLocation
DPMReplicaVolume -FileSystem -TargetServer "DPMServer02" -OverwriteType Overwrite -
RecoveryType Recover -DpmLibrary $Lib
PS C:\> Restore-DPMRecoverableItem -RecoverableItem $Rp -RecoveryOption $Rop
```

Related topics

Get-DPMProtectionGroup

Get-DPMDatasource

Get-DPMRecoveryPoint

Get-DPMLibrary

Restore-DPMRecoverableItem

Get-DPMLibrary

New-DPMRecoveryPoint

New-DPMRecoveryPoint

Creates a recovery point.

Syntax

```
Parameter Set: ApplicationOnDisk
New-DPMRecoveryPoint [-Datasource] <Datasource[]> -Disk [-AdhocJobsContext
<AdhocJobsContext> ] [-BackupType <BackupType> ] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-WithDataIntegrityCheck] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: FileSystemOnDisk
New-DPMRecoveryPoint [-Datasource] <Datasource[]> -Disk -DiskRecoveryPointOption
<CreateDiskRecoveryPointOption> [-AdhocJobsContext <AdhocJobsContext> ] [-
JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-WithDataIntegrityCheck] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: LongTermCloud
New-DPMRecoveryPoint [-Datasource] <Datasource[]> -Online [-AdhocJobsContext
<AdhocJobsContext> ] [-JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: TapeData
New-DPMRecoveryPoint [-Datasource] <Datasource[]> -ProtectionType <ProtectionType> -Tape [-
AdhocJobsContext <AdhocJobsContext> ] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **New-DPMRecoveryPoint** cmdlet creates a recovery point for a data source. You can create a recovery point for short-term and long-term protection types.

Parameters

-AdhocJobsContext<AdhocJobsContext>

Specifies the context details of the adhoc job. Do not use this parameter from the Windows PowerShell command line.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-BackupType<BackupType>

Specifies the type of backup. The acceptable values for this parameter are:

-- ExpressFull

-- Incremental

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource[]>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.

-- Microsoft Exchange storage group.

-- Microsoft SharePoint farm.

-- Microsoft Virtual Machine.

-- DPM database.

-- A system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1

Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Disk

Indicates that DPM creates the recovery point on a disk.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DiskRecoveryPointOption<CreateDiskRecoveryPointOption>

Specifies whether DPM creates a recovery point while synchronizing, or whether DPM synchronizes only and does not create a recovery point. The acceptable values for this parameter are:

-- WithSynchronize

-- WithoutSynchronize

-- OnlySynchronize

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for **Job.StateChanged** events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Online

Specifies that DPM enables online protection.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionType<ProtectionType>

Specifies a protection type. The only valid value is Tape.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape

Specifies that the recovery point is a tape.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WithDataIntegrityCheck

Indicates that DPM performs a data integrity check during recovery.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help New-DPMRecoveryPoint -detailed`".

Examples

Example 1: Create a recovery point and run a backup

This example creates a recovery point for a data source and performs a backup.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command creates a recovery point for the data source stored in the $Ds variable, and performs an expressfull backup.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg
PS C:\> New-DPMRecoveryPoint -Datasource $Ds -Disk -BackupType expressfull
```

Example 2: Create a recovery point and synchronize with the data source

This example creates a recovery point for a data source and synchronizes the recovery point with the data source.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command creates a recovery point for the data source stored in the $Ds variable, and synchronizes the recovery point with the data source.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg
PS C:\> New-DPMRecoveryPoint -Datasource $Ds -Disk -DiskRecoveryPointOption withsynchronize
```

Example 3: Create a short-term recovery point on tape

This example creates a short-term recovery point on tape for a data source.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command creates a recovery point on tape for the data source stored in the $Ds variable, and sets the protection type to short-term.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg
PS C:\> New-DPMRecoveryPoint -Datasource $Ds -Tape -ProtectionType ShortTerm
```

Related topics

Get-DPMProtectionGroup

Get-DPMDatasource

Get-DPMRecoveryPoint

New-DPMRecoveryPoint

Remove-DPMRecoveryPoint

Get-DPMRecoveryPointLocation

New-DPMRecoveryTarget

New-DPMRecoveryTarget

Creates a recovery target.

Syntax

```
Parameter Set: Restricted
New-DPMRecoveryTarget [-Type] {SqlDatabase | SqlInstance | Client} [-RecoveryTarget]
<String> [-RecoveredFilesPath] <String> [ <CommonParameters>]

Parameter Set: Unrestricted
New-DPMRecoveryTarget [-Type] {SqlDatabase | SqlInstance | Client} [-RecoveryTarget]
<String> [-Unrestricted] [ <CommonParameters>]
```

Detailed Description

The **New-DPMRecoveryTarget** cmdlet creates a recovery target.

Parameters

-RecoveredFilesPath<String>

Specifies the folder on the SQL Server recovery instance that the user has permissions to recover for alternate instance recovery.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryTarget<String>

Specifies the SQL Server instance for alternate instance recovery.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AmDatasourceType>

Specifies the type of data source that the cmdlet uses. The acceptable values for this parameter are:

-- SqlDatabase

-- SqlInstance

-- Client

The acceptable values for this parameter are:

SqlDatabase	
SqlInstance	
Client	

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Unrestricted

Indicates that users can recover database files to any location for which the users have write permission.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help New-DPMRecoveryTarget -detailed`".

Examples

Example 1: Create a recovery target

This command creates a recovery target for the instance of SQL Server named West01DB. The *RecoveredFilesPath* parameter specifies the target location where users can recover the files for their databases.

```
PS C:\> New-DPMRecoveryTarget -Type SQLInstance -RecoveryTarget
"ProductionServer\Unit01Instance\West01DB" -RecoveredFilesPath "C:\DPMrecovered"
```

Related topics

Remove-DPMRecoveryTarget

Get-DPMRecoveryTarget

Add-DPMRecoveryTarget

New-DPMRole

New-DPMRole

Creates a DPM role.

Syntax

```
Parameter Set: Default
New-DPMRole [[-DPMServerName] <String> ] [-Name] <String> [[-Description] <String> ] [
<CommonParameters>]
```

Detailed Description

The **New-DPMRole** cmdlet creates a System Center 2012 – Data Protection Manager (DPM) role. DPM roles allow Microsoft SQL Server database owners to recover databases without assistance from a DPM administrator.

After you create a DPM role, use the **Add-DPMSecurityGroup** cmdlet to add the role to appropriate security groups. Use the **Add-DPMRecoveryItem** cmdlet to specify instances of SQL Server and SQL Server databases that DPM protects. Use the **Add-DPMRecoveryTarget** cmdlet to specify target computers running SQL Server. After you make any changes to a DPM role, use the **Set-DPMRole** cmdlet to save those changes.

Parameters

-Description<String>

Specifies a description for the DPM role.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Name<String>

Specifies a name for the DPM role.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help New-DPMRole -detailed`".

Examples

Example 1: Create a role

This command creates a DPM role named OpsMgrSQL on the server named DPMServer07. The command includes a description for the role.

```
PS C:\> New-DPMRole -DPMServerName "DPMServer07" -Name "OpsMgrSQL" -Description "Operations
Manager SQL"
```

Related topics

Get-DPMRole

Remove-DPMRole

Rename-DPMRole

Set-DPMRole

Add-DPMSecurityGroup

Add-DPMRecoveryItem

Add-DPMRecoveryTarget

New-DPMSearchOption

New-DPMSearchOption

Creates an object that specifies search options for recoverable objects.

Syntax

```
Parameter Set: Default
New-DPMSearchOption [-FromRecoveryPoint] <DateTime> [-ToRecoveryPoint] <DateTime> [-
SearchDetail] <SearchForDetail> [-SearchType] <SearchFilterType> [-SearchString] <String> [-
Location <String> ] [-Recursive] [ <CommonParameters>]
```

Detailed Description

The **New-DPMSearchOption** cmdlet creates an object that specifies search options for recoverable objects that System Center 2012 – Data Protection Manager (DPM) manages. You can specify a search string, a range of recovery points to search, the type of object to search for, as well as a location and whether to search recursively.

Use the **Get-DPMRecoverableItem** cmdlet to perform a search.

Parameters

-FromRecoveryPoint<DateTime>

Specifies a date time object. This is the first date of the search range.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Location<String>

Specifies the location of a recovery point.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Recursive

Indicates that the search is recursive.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SearchDetail<SearchForDetail>

Specifies the type of object to search for. The acceptable values for this parameter are:

-- FileFolders

-- MailboxByAlias

-- MailboxByDisplayName

-- WssSite

-- WssDocuments

Aliases	none
Required?	true
Position?	3

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SearchString<String>

Specifies a string to search for in the target recovery points.

Aliases	none
Required?	true
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SearchType<SearchFilterType>

Specifies the type of comparison for the search. The acceptable values for this parameter are:

-- startsWith

-- contains

-- endsWith

-- exactMatch

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ToRecoveryPoint<DateTime>

Specifies a date time object. This is the last date of the search range.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **SearchOption**

Notes

- For more information, type "`Get-Help New-DPMSearchOption -detailed`".

Examples

Example 1: Create a search option

This command creates a search option object for recovery points between September 15, 2009 and September 28, 2009. The object specifies searches for files and folders that contain the string tobe in the location D:\. The command specifies the search as recursive, therefore, the search also checks files and folders in directories and subdirectories of the D:\ drive.

```
PS C:\> New-DPMSearchOption -FromRecoveryPoint "15 September 2009" -ToRecoveryPoint "28
September 2009" -SearchDetail FileFolders -SearchType contains -Recursive -SearchString
"tobe" -Location "D:\"
```

Related topics

Get-DPMRecoverableItem

New-DPMServerScope

New-DPMServerScope

Creates a DPM server scope object.

Syntax

```
Parameter Set: Default
New-DPMServerScope [[-DPMServerName] <String> ] [-ObjectType] {DpmServer | Datasource |
ProtectionGroup | ProductionServer | Disk | Volume | Library | Drive | AdhocAction} [-
ObjectId] <String[]> [-AlertType {RestoreDBAlert | NullType | AgentIncompatibleAlert |
AgentUnreachableAlert | MediaVerificationFailedAlert | MediaEraseFailedAlert |
DetailedInventoryFailedAlert | MediaDecommissionedAlert | MediaDataEraseAlert |
FreeMediaThresholdAlert | DataSetCopyFailedAlert | BackupToTapeFailedAlert |
BackupToTapeCatalogFailedAlert | LibraryDriveAlert | LibraryNotAvailableAlert |
LibraryNotWorkingEfficientlyAlert | MediaRequiredAlert |
ReplicaInitializationInProgressAlert | SynchronizationFailedAlert |
StopProtectionFailedAlert | RecoveryInProgressAlert | RecoveryPartiallySuccessfulAlert |
RecoverySuccessfulAlert | RecoveryFailedAlert | ShadowCopyFailedAlert |
ReplicaInMissingStateAlert | ReplicaInInvalidStateAlert | PartialDeployedClusterAlert |
AgentTaskFailAlert | SqmOptInAlert | DiskThresholdCrossedAlert | VerificationInProgressAlert
| DiskMissingAlert | CatalogThresholdCrossedAlert | DatasetDataVerificationFailed |
SCDiskThresholdCrossedAlert | ConfigureProtectionFailedAlert | ReplicaManualLoadPendingAlert
| ReplicaInitializationPendingAlert | CertificateExpiringAlert | EvalShareInquiryAlert |
ShadowCopyConsolidationRequired | PathChangedForShareAlert |
BackupMetadataEnumerationFailedAlert | DuplicateDisksDetectedAlert |
DataCorruptionDetectedAlert | DataCorruptionDetectedDuringReadAlert |
StagingAreaRestoreInProgressAlert | StagingAreaRestorePartiallySuccessfulAlert |
StagingAreaRestoreSuccessfulAlert | StagingAreaRestoreFailedAlert |
AgentOwnershipRequiredAlert | AutoInstanceProtectionFailedAlert | AgentAttachFailedAlert |
BackupSLAFailedAlert | DpmoRPCreationFailureAlert | DOCVolumeMissing |
SharepointROAddedAlert | SharepointRORemovedAlert | PartialBackupSuccessAlert |
GlobalDbNotAvailable | LibraryDevicesDisabledAlert | LdmWarningThresholdReachedAlert |
LdmErrorThresholdReachedAlert | CertificateExpiryWarningAlert | CertificateExpiryErrorAlert
| RecoveryFailedWarningAlert | ExternalAlert | OnlineBackupServiceUnreachableAlert |
OnlineBackupPoliciesInconsistentAlert | CloudBackupFailedAlert |
PartialCloudBackupSuccessAlert} ] [-DetailedErrorId <Int32> ] [-ErrorId <Int32> ] [
<CommonParameters>]
```

Detailed Description

The **New-DPMServerScope** cmdlet creates a System Center 2012 – Data Protection Manager (DPM) server scope object. You can create a scoped DPM server connection by using the **Connect-DPMServer** cmdlet with a scope object.

Parameters

-AlertType<AlertEnum>

Specifies the type of alert for the connection scope. The acceptable values for this parameter are:

-- RestoreDBAlert

-- NullType

-- AgentIncompatibleAlert

-- AgentUnreachableAlert

-- MediaVerificationFailedAlert

-- MediaEraseFailedAlert

-- DetailedInventoryFailedAlert

-- MediaDecommissionedAlert

-- MediaDataEraseAlert

-- FreeMediaThresholdAlert

-- DataSetCopyFailedAlert

-- BackupToTapeFailedAlert

-- BackupToTapeCatalogFailedAlert

-- LibraryDriveAlert

-- LibraryNotAvailableAlert

-- LibraryNotWorkingEfficientlyAlert

-- MediaRequiredAlert

-- ReplicaInitializationInProgressAlert

-- SynchronizationFailedAlert

-- StopProtectionFailedAlert

-- RecoveryInProgressAlert

-- RecoveryPartiallySuccessfulAlert

-- RecoverySuccessfulAlert

-- RecoveryFailedAlert

-- ShadowCopyFailedAlert

-- ReplicaInMissingStateAlert

-- ReplicaInInvalidStateAlert

-- PartialDeployedClusterAlert

-- AgentTaskFailAlert

-- SqmOptInAlert
-- DiskThresholdCrossedAlert
-- VerificationInProgressAlert
-- DiskMissingAlert
-- CatalogThresholdCrossedAlert
-- DatasetDataVerificationFailed
-- SCDiskThresholdCrossedAlert
-- ConfigureProtectionFailedAlert
-- ReplicaManualLoadPendingAlert
-- ReplicaInitializationPendingAlert
-- CertificateExpiringAlert
-- EvalShareInquiryAlert
-- ShadowCopyConsolidationRequired
-- PathChangedForShareAlert
-- BackupMetadataEnumerationFailedAlert
-- DuplicateDisksDetectedAlert
-- DataCorruptionDetectedAlert
-- DataCorruptionDetectedDuringReadAlert
-- StagingAreaRestoreInProgressAlert
-- StagingAreaRestorePartiallySuccessfulAlert
-- StagingAreaRestoreSuccessfulAlert
-- StagingAreaRestoreFailedAlert
-- AgentOwnershipRequiredAlert
-- AutoInstanceProtectionFailedAlert
-- AgentAttachFailedAlert
-- BackupSLAFailedAlert
-- DpmoRPCreationFailureAlert
-- DOCVolumeMissing
-- SharepointROAddedAlert
-- SharepointRORemovedAlert
-- PartialBackupSuccessAlert
-- GlobalDbNotAvailable
-- LibraryDevicesDisabledAlert
-- LdmWarningThresholdReachedAlert
-- LdmErrorThresholdReachedAlert
-- CertificateExpiryWarningAlert
-- CertificateExpiryErrorAlert
-- RecoveryFailedWarningAlert
-- ExternalAlert

-- OnlineBackupServiceUnreachableAlert
-- OnlineBackupPoliciesInconsistentAlert
-- CloudBackupFailedAlert
-- PartialCloudBackupSuccessAlert
The acceptable values for this parameter are:

RestoreDBAlert	
NullType	
AgentIncompatibleAlert	
AgentUnreachableAlert	
MediaVerificationFailedAlert	
MediaEraseFailedAlert	
DetailedInventoryFailedAlert	
MediaDecommissionedAlert	
MediaDataEraseAlert	
FreeMediaThresholdAlert	
DataSetCopyFailedAlert	
BackupToTapeFailedAlert	
BackupToTapeCatalogFailedAlert	
LibraryDriveAlert	
LibraryNotAvailableAlert	
LibraryNotWorkingEfficientlyAlert	
MediaRequiredAlert	
ReplicaInitializationInProgressAlert	
SynchronizationFailedAlert	
StopProtectionFailedAlert	
RecoveryInProgressAlert	
RecoveryPartiallySuccessfulAlert	
RecoverySuccessfulAlert	
RecoveryFailedAlert	
ShadowCopyFailedAlert	
ReplicaInMissingStateAlert	
ReplicaInInvalidStateAlert	

PartialDeployedClusterAlert	
AgentTaskFailAlert	
SqmOptInAlert	
DiskThresholdCrossedAlert	
VerificationInProgressAlert	
DiskMissingAlert	
CatalogThresholdCrossedAlert	
DatasetDataVerificationFailed	
SCDiskThresholdCrossedAlert	
ConfigureProtectionFailedAlert	
ReplicaManualLoadPendingAlert	
ReplicaInitializationPendingAlert	
CertificateExpiringAlert	
EvalShareInquiryAlert	
ShadowCopyConsolidationRequired	
PathChangedForShareAlert	
BackupMetadataEnumerationFailedAlert	
DuplicateDisksDetectedAlert	
DataCorruptionDetectedAlert	
DataCorruptionDetectedDuringReadAlert	
StagingAreaRestoreInProgressAlert	
StagingAreaRestorePartiallySuccessfulAlert	
StagingAreaRestoreSuccessfulAlert	
StagingAreaRestoreFailedAlert	
AgentOwnershipRequiredAlert	
AutoInstanceProtectionFailedAlert	
AgentAttachFailedAlert	
BackupSLAFailedAlert	
DpmoRPCreationFailureAlert	
DOCVolumeMissing	
SharepointROAddedAlert	

SharepointRORemovedAlert	
PartialBackupSuccessAlert	
GlobalDbNotAvailable	
LibraryDevicesDisabledAlert	
LdmWarningThresholdReachedAlert	
LdmErrorThresholdReachedAlert	
CertificateExpiryWarningAlert	
CertificateExpiryErrorAlert	
RecoveryFailedWarningAlert	
ExternalAlert	
OnlineBackupServiceUnreachableAlert	
OnlineBackupPoliciesInconsistentAlert	
CloudBackupFailedAlert	
PartialCloudBackupSuccessAlert	

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DetailedErrorId<Int32>

Specifies an integer detailed error ID.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ErrorId<Int32>

Specifies an integer error ID.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ObjectId<String[]>

Specifies an array of identifiers for objects that scope a connection.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-ObjectType<ScopedObjectType>

Specifies the type of objects defined in the *ObjectId* parameter. The object types and identifiers are the following:

-- Datasource. DatasourceID.

-- ProtectionGroup. ProtectionGroupID/ProtectionGroupName.

-- ProductionServer. ProductionServerID/ProductionServerName.

-- Disk. DiskID.

-- Volume. VolumeID.

-- Library. LibraryID.

-- Drive. DriveID.

-- AdhocAction. ActionID.

-- DPMServer. DPMServerName.

The acceptable values for this parameter are:

DpmServer	
Datasource	
ProtectionGroup	
ProductionServer	
Disk	
Volume	
Library	
Drive	
AdhocAction	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help New-DPMServerScope -detailed`".

Examples

Example 1: Create server scope for a connection

This example creates a connection to a DPM server.

The first command creates a server scope for the DPM server named DPMServer07, and stores that scope in the $ServerScope variable. The command specifies ProtectionGroup as the object type, with an object ID of Protection Group 01.

The second command uses the **Connect-DPMServer** cmdlet to create a connection that uses the server scope stored in the $ServerScope variable.

```
PS C:\> $ServerScope=New-DPMServerScope -DPMServerName "DPMServer07" -ObjectType
ProtectionGroup -ObjectID "Protection Group 01"
 PS C:\> $DPMServer=Connect-DPMServer -DPMServerScope $ServerScope
```

Example 2: Create server scope that has alerts specified

This example creates a connection to a DPM server.

The first command creates a server scope for the DPM server named DPMServer07, and stores that scope in the $ServerScope variable. The command specifies Datasource as the object type, with the specified object ID. The command also specifies an alert type and both detailed error ID and error ID.

The second command uses the **Connect-DPMServer** cmdlet to create a connection that has the server scope stored in the $ServerScope variable.

```
PS C:\> $ServerScope=New-DPMServerScope -DPMServerName "DPMServer07" -ObjectType Datasource
-ObjectID 70fd1133-1bab-4178-8e8c-24d928b6b16c -AlertType BackupToTapeFailedAlert -
DetailedErrorID 910 -ErrorID 3113
PS C:\> $DPMServer=Connect-DPMServer -DPMServerScope $ServerScope
```

Related topics

Connect-DPMServer

Remove-DPMBackupNetworkAddress

Remove-DPMBackupNetworkAddress

Removes a backup network from a DPM server.

Syntax

```
Parameter Set: Default
Remove-DPMBackupNetworkAddress [[-DPMServerName] <String> ] [-Address] <String> [-Confirm]
[-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMBackupNetworkAddress** cmdlet removes a backup network from a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-Address<String>

Specifies the IP address or subnet mask of a backup network.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none

Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMBackupNetworkAddess -detailed`".

Examples

Example 1: Remove a backup network

This command removes the network address 10.10.10.1/16 as a backup network address for the DPM server named DPMServer07.

```
PS C:\> Remove-BackupNetworkAddress -DpmServername "DPMServer07" -Address "10.10.10.1/16"
```

Related topics

Add-DPMBackupNetworkAddress

Get-DPMBackupNetworkAddress

Remove-DPMChildDatasource

Remove-DPMChildDatasource

Removes a data source or child data source from a protection group.

Syntax

```
Parameter Set: StopProtection
Remove-DPMChildDatasource [-ProtectionGroup] <ProtectionGroup> [-ChildDatasource]
<ProtectableObject[]> [-KeepDiskData] [-KeepOnlineData] [-KeepTapeData] [-PassThru] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: RejectCloud
Remove-DPMChildDatasource [-ProtectionGroup] <ProtectionGroup> [-ChildDatasource]
<ProtectableObject[]> [-Online] [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMChildDatasource** cmdlet removes a data source or child data source from a System Center 2012 – Data Protection Manager (DPM) protection group. A child data source can refer to files or folders on a protected volume.

If you remove a data source using the **Remove-Datasource** cmdlet, it is equivalent to setting an exclusion.

Parameters

-ChildDatasource<ProtectableObject[]>

Specifies an array of data sources, such as folders in a file system, that you can protect individually.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-KeepDiskData

Indicates that the cmdlet keeps existing data on a disk.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-KeepOnlineData

Indicates that the cmdlet keeps replicas even after protection ends. You must also specify the *KeepDiskData* parameter to enable this parameter.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-KeepTapeData

Indicates that the cmdlet keeps existing data on a tape.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-Online

Indicates whether online protection is enabled.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none

Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Remove-DPMChildDatasource -detailed`".

Examples

Example 1: Remove a child data source

This example removes the ninth child data source stored in a parent data source.

The first command gets all protection groups from the DPM server named DPMServer07 and stores these groups in the $Pg variable. You cannot edit these protection groups.

The second command gets the first member of the $Pg array in editable mode, and then stores this protection group in the $Mpg variable.

The third command gets the data source for the list of protection groups in the $Pg variable and stores this data source in the $Po variable.

The fourth command uses standard array notation to specify the ninth element of the $Po array. The command removes that data source from the protection group stored in the $Mpg variable.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer07"
PS C:\> $Mpg = Get-ModifiableProtectionGroup $Pg[0]
PS C:\> $Po = Get-Datasource -ProtectionGroup $Pg
PS C:\> Remove-DPMChildDatasource -ProtectionGroup $Mpg -ChildDatasource $Po[8]
```

Related topics

Add-DPMChildDatasource

Get-DPMChildDatasource

Remove-DPMDatasourceReplica

Remove-DPMDatasourceReplica

Removes an inactive replica.

Syntax

```
Parameter Set: Disk
Remove-DPMDatasourceReplica [-Datasource] <Datasource> -Disk [-PassThru] [-Confirm] [-
WhatIf] [ <CommonParameters>]

Parameter Set: Online
Remove-DPMDatasourceReplica [-Datasource] <Datasource> -Online [-PassThru] [-Confirm] [-
WhatIf] [ <CommonParameters>]

Parameter Set: Tape
Remove-DPMDatasourceReplica [-Datasource] <Datasource> -Tape [-PassThru] [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMDatasourceReplica** cmdlet removes an inactive replica from disk or tape.

Parameters

-Datasource<Datasource>

Specifies a Windows file system share or volume, Microsoft SQL Server database, Microsoft Exchange storage group, Microsoft SharePoint farm, virtual machine, System Center 2012 –
Data Protection Manager (DPM) database, or system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Disk

Indicates that the cmdlet removes the replica from disk.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Online

Indicates that the data source uses online protection.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-Tape

Indicates that the cmdlet removes the replica from tape.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMDatasourceReplica -detailed`".
  For technical information, type "`Get-Help Remove-DPMDatasourceReplica -full`".

Examples

Example 1: Remove a data source replica

This example removes a data source replica from disk.

The first command gets the protection group from the DPM server named DPMServer02 and stores the protection group in the $Pg variable.

The second command gets the data source from the protection group in the $Pg variable stores the data source in the $Ds variable.

The last command removes the replica of the data source in the $Ds variable from disk.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds = Get-Datasource -ProtectionGroup $Pg
PS C:\> Remove-DPMDatasourceReplica -Datasource $Ds -Disk
```

Related topics

Get-DPMDatasource

Remove-DPMDisk

Remove-DPMDisk

Removes a disk from a storage pool.

Syntax

```
Parameter Set: Default
Remove-DPMDisk [-DPMDisk] <Disk[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMDisk** cmdlet removes a disk from a storage pool. A storage pool in System Center 2012 – Data Protection Manager (DPM) consists of a set of disks where the DPM server stores replicas, shadow copies, and transfer logs for protected data sources.

To get a list of all disks on a DPM server, use the **Get-DPMDisk** cmdlet.

Parameters

-DPMDisk<Disk[]>

Specifies an array of disks that make up a storage pool.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Disk**

Notes

- For more information, type "`Get-Help Remove-DPMDisk -detailed`".

Examples

Example 1: Remove a disk from a storage pool

This example removes disks from a storage pool.

The first command uses the **Get-DPMDisk** cmdlet to retrieve a list of disks on a server, and stores the result in the $DPMDisk variable.

The second command uses the **Remove-DPMDisk** cmdlet to remove the disks in the variable $DPMDisk from the server storage pool.

```
PS C:\> $DPMDisk = Get-DPMDisk -DPMServerName "Contoso-DPMServer"
PS C:\> Remove-DPMDisk -DPMDisk $DPMDisk
```

Related topics

Add-DPMDisk

Get-DPMDisk

Remove-DPMObject

Remove-DPMObject

Removes a DPM object.

Syntax

```
Parameter Set: Default
Remove-DPMObject [-DPMObject] <IDisposable> [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMObject** cmdlet removes or clears a System Center 2012 – Data Protection Manager (DPM) object.

Parameters

-DPMObject<IDisposable>

Specifies a DPM object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Remove a DPM object

This command removes the DPM object in the variable named $PGroup.

```
PS C:\> Remove-DPMObject -DPMObject $PGroup
```

Remove-DPMPGSet

Remove-DPMPGSet

Deletes a PG set.

Syntax

```
Parameter Set: Default
Remove-DPMPGSet [-PGSet] <PGSet> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMPGSet** cmdlet deletes a System Center 2012 – Data Protection Manager (DPM) protection group (PG) set. A DPM PG is a collection of protection groups that you collocate on the same tape.

Parameters

-PGSet<PGSet>

Specifies the PG set object that you delete. To obtain a PG set object, use the **Get-DPMPGSet** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMPGSet -detailed`".

Examples

Example 1: Remove a PG set

This example removes the first PG set from the PG sets on the server named DPMServer07.

The first command uses the **Get-DPMPGSet** cmdlet to get the PG sets for the specified server and stores them in the $PgSet variable.

The second command specifies the first member of the $PgSet variable by using standard array notation. The command removes that PG set.

```
PS C:\> $PgSet = Get-DPMPGSet -DPMServerName "DPMServer07"
PS C:\> Remove-DPMPGSet -PGSet $PgSet[0]
```

Related topics

Get-DPMPGSet

New-DPMPGSet

Update-DPMPGSet

Remove-DPMRecoveryItem

Remove-DPMRecoveryItem

Removes a recovery item.

Syntax

```
Parameter Set: All
Remove-DPMRecoveryItem [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [-
All] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Datasources
Remove-DPMRecoveryItem [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [-
Datasources] <SQLDataSource[]> [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: SqlInstances
Remove-DPMRecoveryItem [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client} [-
SqlInstances] <String[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMRecoveryItem** cmdlet removes a recovery item from the list of recoverable items that members of a System Center 2012 – Data Protection Manager (DPM) role can recover. You must specify at least one instance of SQL Server for this cmdlet, regardless of whether you specify the *All* parameter.

Parameters

-All

Indicates that the remove action affects all objects that the cmdlet references.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-Datasources<SQLDataSource[]>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.

-- Microsoft Exchange storage group.

-- Microsoft SharePoint Server farm.

-- Microsoft Virtual Machine.

-- DPM database.

-- A system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SqlInstances<String[]>

Specifies an array of SQL Server instances.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AmDatasourceType>

Specifies the type of data source that the cmdlet uses. The acceptable values for this parameter are:

-- SqlDatabase

-- SqlInstance

-- Client

The acceptable values for this parameter are:

SqlDatabase	
SqlInstance	
Client	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMRecoveryItem -detailed`".

Related topics

Get-DPMRecoveryItem

Add-DPMRecoveryItem

Get-DPMRole

Remove-DPMRecoveryPoint

Remove-DPMRecoveryPoint

Removes a recovery point from tape or disk.

Syntax

```
Parameter Set: Default
Remove-DPMRecoveryPoint [-RecoveryPoint] <RecoverySource> [-ForceDeletion] [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMRecoveryPoint** cmdlet removes a recovery point from tape or disk. If a recovery point exists in multiple locations, for example, on disk and tape, or two tapes, use the **Get-DPMRecoveryPointLocation** cmdlet to get the location of the recovery point.

Parameters

-ForceDeletion

Indicates that DPM prunes the data source regardless if it is currently running a backup job.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryPoint<RecoverySource>

Specifies a recovery point object. To obtain a recovery point location object, use the **Get-DPMRecoveryPoint** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMRecoveryPoint -detailed`".

Examples

Example 1: Remove a recovery point

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command gets the recovery point for the for the data source stored in the $Ds variable, and stores the results in the $Rp variable.

The fourth command removes the recovery point stored in the $Rp variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"

PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg

PS C:\> $Rp = Get-DPMRecoveryPoint -Datasource $Ds

PS C:\> Remove-DPMRecoveryPoint -RecoveryPoint $Rp
```

Related topics

Get-DPMProtectionGroup

Get-DPMDatasource

Get-DPMRecoveryPoint

New-DPMRecoveryPoint

Remove-DPMRecoveryTarget

Remove-DPMRecoveryTarget

Revokes permissions to a recovery location.

Syntax

```
Parameter Set: Default
Remove-DPMRecoveryTarget [-DpmRole] <DpmRole> [-Type] {SqlDatabase | SqlInstance | Client}
[[-RecoveryTargets] <String[]> ] [-All] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMRecoveryTarget** cmdlet revokes permissions to a recovery location for a System Center 2012 – Data Protection Manager (DPM) role.

Parameters

-All

Indicates that the remove action affects all objects that the cmdlet references.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none

Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryTargets<String[]>

Specifies an array of target recovery items which consist of the instance of SQL Server and the folder to use for alternate instance recovery.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AmDatasourceType>

Specifies the type of data source that the cmdlet uses. The acceptable values for this parameter are:

-- SqlDatabase

-- SqlInstance

-- Client

The acceptable values for this parameter are:

SqlDatabase	
SqlInstance	
Client	

Aliases	none
Required?	true

Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMRecoveryTarget -detailed`".

Examples

Example 1: Revoke permission to a recovery location

This example revokes permission to a recovery location for a DPM role.

The first command gets a DPM role named OpsMgrSQL and stores the result in the $Role variable.

The second command revokes permission to the target recovery location C:\Div02Recovery for the DPM role stored in the $Role variable.

```
PS C:\> $Role = Get-DPMRole -Name "OpsMgrSQL"
PS C:\> Remove-DPMRecoveryTarget -DPMRole $Role -Type SQLInstance -RecoveryTargets
"C:\Div02Recovery"
```

Related topics

Get-DPMRecoveryTarget

Add-DPMRecoveryTarget

Add-DPMRecoveryTarget

Get-DPMRole

Remove-DPMRole

Remove-DPMRole

Deletes a DPM role.

Syntax

```
Parameter Set: Default
Remove-DPMRole [-DpmRole] <DpmRole> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMRole** cmdlet deletes a System Center 2012 – Data Protection Manager (DPM) role. DPM roles allow Microsoft SQL Server database owners to recover databases without assistance from a DPM administrator.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMRole -detailed`".

Examples

Example 1: Remove a role

This example removes a DPM role named OpsMgrSQL.

The first command gets the role by using the **Get-DpmRole** cmdlet and stores it in the $DpmRole variable. The command makes the role editable.

The second command removes the role stored in the $DpmRole variable.

```
PS C:\> $DpmRole = Get-DpmRole –Name "OpsMgrSQL" -Editable
PS C:\> Remove-DPMRole -DpmRole $DpmRole
```

Related topics

Get-DPMRole

New-DPMRole

Rename-DPMRole

Set-DPMRole

Remove-DPMSecurityGroup

Remove-DPMSecurityGroup

Removes security groups from a DPM role.

Syntax

```
Parameter Set: Default
Remove-DPMSecurityGroup [-DpmRole] <DpmRole> [[-SecurityGroups] <String[]> ] [-All] [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-DPMSecurityGroup** cmdlet removes one or more security groups from a System Center 2012 – Data Protection Manager (DPM) role. You can use the **Get-DPMRole** to specify a DPM role. Use the **Set-DPMRole** cmdlet to save your changes.

You can see the security groups for a DPM role by using the **Get-DPMSecurityGroup** cmdlet. You can use the **Add-DPMSecurityGroup** cmdlet to add security groups to a DPM role.

Parameters

-All

Indicates that the action affects all objects that the cmdlet refers to.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SecurityGroups<String[]>

Specifies an array of security groups. The cmdlet removes these groups from the DPM role.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMSecurityGroup -detailed`".

Examples

Example 1: Remove a security group from a role

This example removes a security group from a role.

The first command uses the **Get-DPMRole** cmdlet to get the role named OpsMgrSQL and stores it in the $DpmRole variable. The command makes the role editable.

The second command removes the security group DpmDom02\Administrator from the role stored in the $DpmRole variable.

```
PS C:\> $DpmRole = Get-DPMRole –Name "OpsMgrSQL" –Editable
PS C:\> Remove-DPMSecurityGroup -DpmRole $DpmRole -SecurityGroups "DpmDom02\Administrator"
```

Related topics

Add-DPMSecurityGroup

Get-DPMSecurityGroup

Get-DPMRole

Set-DPMRole

Remove-DPMTape

Remove-DPMTape

Removes a tape from a DPM library.

Syntax

```
Parameter Set: Default
Remove-DPMTape [-DPMLibrary] <Library> [-Tape] <Media[]> [-Async] [-
JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Remove-DPMTape** cmdlet removes a tape from a System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation to the user. Use this parameter in conjunction with the *Async* parameter.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media[]>

Specifies an array of tape objects.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Remove-DPMTape -detailed`".

Examples

Example 1: Remove a tape from a library

This example removes a tape from the library.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library for the server, and stores the result in the $DPMLib variable.

The second command uses the **Get-Tape** cmdlet to retrieve the tape object, and stores the result in the $Tape variable.

The third command uses the **Unlock-DPMLibraryDoor** cmdlet to unlock the tape door.

The fourth command uses the **Remove-DPMTape** cmdlet to remove the third tape from the library.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $Tape = Get-Tape -DPMLibrary $DPMLib
PS C:\> Unlock-DPMLibraryDoor -DPMLibrary $DPMLib
PS C:\> Remove-DPMTape -DPMLibrary $DPMLib -Tape $Tape[2]
```

Related topics

Add-DPMTape

Get-DPMLibrary

Get-DPMTape

Set-DPMTape

Rename-DPMLibrary

Rename-DPMLibrary

Renames a DPM library.

Syntax

```
Parameter Set: Default
Rename-DPMLibrary [-DPMLibrary] <Library> [-NewName] <String> [-PassThru] [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Rename-DPMLibrary** cmdlet renames a System Center 2012 – Data Protection Manager (DPM) library in the DPM Administrator Console.

Parameters

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-NewName<String>

Specifies a new name for the library.

Aliases	none

Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByPropertyName)
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByPropertyName)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Library**

Notes

- For more information, type "`Get-Help Rename-DPMLibrary -detailed`".

Examples

Example 1: Rename a library

This example renames a DPM library associated with the server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the tape library for the specified server, and stores that object in the $DPMLib variable.

The second command changes the name the library stored in the $DPMLib variable to Library Western 03.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Rename-DPMLibrary -DPMLibrary $DPMLib -NewName "Library Western 03"
```

Related topics

Disable-DPMLibrary

Enable-DPMLibrary

Get-DPMLibrary

Rename-DPMProtectionGroup

Rename-DPMProtectionGroup

Renames a protection group.

Syntax

```
Parameter Set: Default
Rename-DPMProtectionGroup [-ProtectionGroup] <ProtectionGroup> [-NewName] <String> [-
PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Rename-DPMProtectionGroup** cmdlet renames a protection group on the System Center 2012 – Data Protection Manager (DPM) server. The process of renaming a protection group requires the following steps:

-- 1. Retrieve the protection group by using the **Get-DPMProtectionGroup** cmdlet.

-- 2. Make the protection group modifiable by using the **Get-DPMModifiableProtectionGroup** cmdlet.

-- 3. Rename the protection group by using the **Rename-DPMProtectionGroup** cmdlet.

-- 4. Save the changes by using the **Set-DPMProtectionGroup** cmdlet.

Parameters

-NewName<String>

Specifies a new name for the protection group.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Rename-DPMProtectionGroup -detailed`".

Examples

Example 1: Rename a protection group

This example renames a protection group on the DPM server.

The first command gets the protection group on the DPM server named DPMServer02, and stores the results in the $Pg variable.

The second command gets the protection group in a modifiable mode and stores the results in the $Mpg variable.

The third command renames the protection group stored in the $Mpg variable to ProtectGroup02.

The fourth command saves all the actions on the DPM server that you performed on the protection group stored in the $Mpg variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup - DPMServerName "DPMServer02"
PS C:\> $Mpg = Get-DPMModifiableProtectionGroup -ProtectionGroup $Pg
PS C:\> Rename-DPMProtectionGroup -ProtectionGroup $Mpg -NewName "ProtectGroup02"
PS C:\> Set-DPMProtectionGroup $Mpg
```

Related topics

Update-DPMProtectionGroup

Get-DPMProtectionGroup

Rename-DPMProtectionGroup

Set-DPMProtectionGroup

New-DPMProtectionGroup

Rename-DPMRole

Rename-DPMRole

Changes the name or description of a DPM role.

Syntax

```
Parameter Set: Default
Rename-DPMRole [-DpmRole] <DpmRole> [[-Name] <String> ] [[-Description] <String> ] [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Rename-DPMRole** cmdlet changes the name or description of a System Center 2012 – Data Protection Manager (DPM) role. DPM roles allow Microsoft SQL Server database owners to recover databases without assistance from a DPM administrator. Save your changes by using the **Set-DPMRole** cmdlet.

Parameters

-Description<String>

Specifies a description for the DPM role.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Name<String>

Specifies a name for the DPM role.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Rename-DPMRole -detailed`".

Examples

Example 1: Change the name of a role

This example changes the name of a DPM role.

The first command gets the role by using the **Get-DpmRole** cmdlet and stores it in the $DpmRole variable. The command makes the role editable.

The second command renames the role stored in the $DpmRole variable.

The third command saves the change made in the second command for the role stored in the $DpmRole variable by using the **Set-DPMRole** cmdlet.

```
PS C:\> $DpmRole = Get-DpmRole –Name "OpsMgrSQL" -Editable
PS C:\> Rename-DPMRole -DpmRole $DpmRole -Name "OpsMgrSQL23"
PS C:\> Set-DPMRole -DPMRole $DpmRole
```

Related topics

Get-DPMRole

New-DPMRole

Remove-DPMRole

Set-DPMRole

Restart-DPMJob

Restart-DPMJob

Restarts failed DPM jobs.

Syntax

```
Parameter Set: Default
Restart-DPMJob [-Job] <Job[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Restart-DPMJob** cmdlet restarts failed System Center 2012 – Data Protection Manager (DPM) jobs.

Parameters

-Job<Job[]>

Specifies an array of job objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Restart failed jobs

This example restarts the failed jobs for a protection group on the DPM server Contoso-DPMServer.

The first command uses the **Get-ProtectionGroup** cmdlet to retrieve the protection group, and then stores the result in the variable named $PGroup.

The second command uses the **Get-DPMJob** cmdlet to retrieve the list of failed jobs for the protection group, and then stores the result in the variable named $Job.

The third command uses the **Restart-DPMJob** cmdlet to restart the jobs stored in $Job.

```
PS C:\> $PGroup = Get-ProtectionGroup –DPMServerName "Contoso-DPMServer"
PS C:\> $Job = Get-DPMJob -ProtectionGroup $PGroup -Status $Failed
PS C:\> Restart-DPMJob $Job
```

Related topics

Get-DPMJob

Stop-DPMJob

Restore-DPMRecoverableItem

Restore-DPMRecoverableItem

Restores a version of the data source to a target location.

Syntax

```
Parameter Set: Default
Restore-DPMRecoverableItem [-RecoveryOption] <RecoveryOptions> [[-RecoverableItem]
<RecoverableObject[]> ] [-AdhocJobsContext <AdhocJobsContext> ] [-
JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-RecoveryNotification
<NotificationObject> ] [-RecoveryPointLocation <RecoverySourceLocation[]> ] [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Restore-DPMRecoverableItem** cmdlet recovers a point in time version of a data source (recovery point), or a child recoverable item within it, to the target location.

Parameters

-AdhocJobsContext<AdhocJobsContext>

Specifies the context details of the ad hoc job. Do not use this parameter from the Windows PowerShell command line.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for **Job.StateChanged** events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoverableItem<RecoverableObject[]>

Specifies a recoverable item object. This is a child item within a recovery point that is recoverable. For example, a Windows file system share or volume, a Microsoft SQL Server database, a Microsoft Exchange storage group, Microsoft SharePoint, Microsoft Virtual Machine, a Microsoft DPM database, system state or a recovery point.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoveryNotification<NotificationObject>

Specifies a notification to send when recovery is finished. You can use the **New-DPMRecoveryNotification** cmdlet to create a notification object.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryOption<RecoveryOptions>

Specifies the recovery options for the data source. You can use the **New-DPMRecoveryOption** cmdlet to create recovery options.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoveryPointLocation<RecoverySourceLocation[]>

Specifies an array of recovery point locations. To obtain a recovery point location object, use the **Get-DPMRecoveryPointLocation** cmdlet. You need to specify the location of a recovery point if a recovery item exists in more than one recovery point.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Restore-DPMRecoverableItem -detailed`".

Examples

Example 1: Restore a version of a data source to a target location

This example restores a version of a Hyper-V data source to a recovery location.

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command gets the list of data sources for the first protection group in the collection stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command gets the recovery point for the first data source object stored in the $Ds array, and passes it to the **Sort-Object** cmdlet by using the pipe operator. The **Sort-Object** cmdlet sorts the collection of recovery points in descending order of the date and time value of the RepresentedPointInTime property. The **Select-Object** cmdlet selects the first recovery point from the collection and stores it in the $RecoveryObject variable.

The fourth command creates a recovery option for a Hyper-V data source on the server named HVDCenter02. The command specifies AlternateHyperVServer as the recovery location and specifies

that DPM stores the replica of the data source in c:\VMRecovery. The command stores the recovery option in the $RecoveryOption variable.

The fifth command restores the data source stored in the $RecoveryObject variable by using the recovery option stored in the $RecoveryOption variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup –DPMServerName "DPMServer02"
PS C:\> $Ds = Get-DPMDatasource -ProtectionGroup $Pg[0]
PS C:\> $RecoveryObject = Get-DPMRecoverypoint -Datasource $Ds[0] | Sort -Property
RepresentedPointInTime -Descending | Select-Object -First 1
PS C:\> $RecoveryOption = New-DPMRecoveryOption -HyperVDatasource -TargetServer
"HVDCenter02" -RecoveryLocation AlternateHyperVServer -RecoveryType Recover -TargetLocation
“c:\VMRecovery”
PS C:\> Restore-DPMRecoverableItem -RecoverableItem $RecoveryObject -RecoveryOption
$RecoveryOption
```

Related topics

Get-DPMRecoverableItem

Get-DPMRecoveryPointLocation

New-DPMRecoveryOption

New-DPMRecoveryNotification

Resume-DPMBackup

Resume-DPMBackup

Attempts to resume stalled DPM backup jobs.

Syntax

```
Parameter Set: ResumeAlertBackups
Resume-DPMBackup [-BackupTargetType] {Disk | Tape | Online | Invalid} [-Alert] <Alert[]> [[-
AdhocJobsContext] <AdhocJobsContext> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ResumeDsBackups
Resume-DPMBackup [-BackupTargetType] {Disk | Tape | Online | Invalid} [-Datasource]
<Datasource[]> [[-AdhocJobsContext] <AdhocJobsContext> ] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: ResumePgBackups
Resume-DPMBackup [-BackupTargetType] {Disk | Tape | Online | Invalid} [-ProtectionGroup]
<ProtectionGroup[]> [[-AdhocJobsContext] <AdhocJobsContext> ] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: ResumePsBackups
Resume-DPMBackup [-BackupTargetType] {Disk | Tape | Online | Invalid} [-ProductionServer]
<ProductionServer[]> [[-AdhocJobsContext] <AdhocJobsContext> ] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Resume-DPMBackup** cmdlet attempts to resume stalled System Center 2012 –
Data Protection Manager (DPM) backup jobs. Specify the backup target type and one of the following backup types:

-- Alert

-- Datasource

-- ProtectionGroup

-- ProductionServer

Parameters

-AdhocJobsContext<AdhocJobsContext>

Specifies the context details of the ad hoc job. Do not use this parameter from the Windows PowerShell command line.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Alert<Alert[]>

Specifies an array of alerts for which backups resume.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-BackupTargetType<BackupTargetType>

Specifies the type of backup target. The acceptable values for this parameter are:

-- Disk

-- Tape

-- Online

-- Invalid

The acceptable values for this parameter are:

Disk	

Tape	
Online	
Invalid	

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource[]>

Specifies an array of data sources for which backups resume.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ProductionServer<ProductionServer[]>

Specifies an array of protected computers for which backups resume.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)

Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup[]>

Specifies an array of protection groups for which backups resume.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Resume-DPMBackup -detailed`".

Examples

Example 1: Resume backing up production servers

This example resumes backing up all computers protected by a DPM server. The first command uses the **Get-DPMProductionServer** cmdlet to get the production servers for the named DPM server named DPMServer07, and stores it in the $DPMPS variable.

The second command resumes backing up the production servers stored in the $DPMPS variable for the DPM server named DPMServer07.

```
PS C:\> $DPMPS=Get-DPMProductionServer -DPMServerName "DPMServer07"
PS C:\> Resume-DPMBackup -DPMServerName "DPMServer07" -ProductionServer $DPMPS
```

Related topics

Get-DPMProductionServer

Set-DPMAutoProtectIntent

Set-DPMAutoProtectIntent

Turns DPM auto-protection on or off for a SQL Server instance.

Syntax

```
Parameter Set: Default
Set-DPMAutoProtectIntent [-ProtectionGroup] <ProtectionGroup> [-SQLInstanceName] <String[]>
[-AutoProtectIntent] <AutoProtectionIntent> [-ProductionServer] <ProductionServer> [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMAutoProtectIntent** cmdlet turns System Center 2012 – Data Protection Manager (DPM) auto-protection on or off for a Microsoft SQL Server data management software instance. When you use auto-protection for a SQL Server instance, DPM identifies and protects new databases added to that instance. To change the auto-protection setting, specify the SQL Server instance, the protection group that it belongs to, and a production server.

When you first add a SQL Server instance to a protection group, the DPM server enables auto-protection for that instance.

Parameters

-AutoProtectIntent<AutoProtectionIntent>

Specifies whether the cmdlet turns auto-protection on or off.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProductionServer<ProductionServer>

Specifies a server that runs a DPM agent.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-SQLInstanceName<String[]>

Specifies an array of names of SQL Server instances.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type `"Get-Help Set-DPMAutoProtectIntent -detailed"`.

Related topics

Get-DPMAutoProtectIntent

Start-DPMAutoProtection

Set-DPMCloudSubscriptionSetting

Set-DPMCloudSubscriptionSetting

Updates subscription settings in Windows Azure Online Backup for a DPM server.

Syntax

```
Parameter Set: Commit
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-Commit] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Encryption
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-EncryptionPassphrase] <SecureString> [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: NoProxy
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-NoProxy] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: NoThrottle
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-NoThrottle] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ProxyServer
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-ProxyServer] <String> [-ProxyPort] <Int32> [[-ProxyUsername]
<String> ] [[-ProxyPassword] <SecureString> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: StagingArea
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-StagingAreaPath] <String> [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: Throttle
Set-DPMCloudSubscriptionSetting [[-DPMServerName] <String> ] [-SubscriptionSetting]
<CloudSubscriptionSetting> [-WorkDay] {Sunday | Monday | Tuesday | Wednesday | Thursday |
Friday | Saturday} [-StartWorkHour] <TimeSpan> [-EndWorkHour] <TimeSpan> [-
WorkHourBandwidth] <UInt32> [-NonWorkHourBandwidth] <UInt32> [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Set-DPMCloudSubscriptionSetting** cmdlet updates subscription settings in Windows Azure Online Backup for a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-Commit

Indicates that DPM saves the Windows Azure Online Backup settings.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of the DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-EncryptionPassphrase<SecureString>

Specifies a secure string that contains a passphrase used to encrypt backups from the DPM server.

Aliases	none
Required?	true

Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-EndWorkHour<TimeSpan>

Specifies the end of the time range when DPM uses throttling as specified in the *WorkHourBandwidth* parameter. Use this parameter in conjunction with the *StartWorkHour* parameter.

Aliases	none
Required?	true
Position?	5
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-NonWorkHourBandwidth<UInt32>

Specifies throttling settings for hours outside the range that the *StartWorkHour* and *EndWorkHour* parameters define.

Aliases	none
Required?	true
Position?	7
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-NoProxy

Indicates that the cmdlet does not use a proxy server.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-NoThrottle

Indicates that the cmdlet does not use bandwidth throttling.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProxyPassword<SecureString>

Specifies a secure string that contains the password for the proxy server.

Aliases	none
Required?	false
Position?	6
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-ProxyPort<Int32>

Specifies a port number for the proxy server.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-ProxyServer<String>

Specifies the name of the proxy server.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ProxyUsername<String>

Specifies the user name that you use to log on to the proxy server.

Aliases	none
Required?	false
Position?	5
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-StagingAreaPath<String>

Specifies the path to which you download backups before you recover them to their final location. Ensure that the location you specify has sufficient space to hold the backups.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-StartWorkHour<TimeSpan>

Specifies the start of the time range when the throttling settings in the *WorkHourBandwidth* parameter are in effect. Use this parameter in conjunction with the *EndWorkHour* parameter.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-SubscriptionSetting<CloudSubscriptionSetting>

Specifies a **Subscription** object that contains the subscription settings. To obtain a **Subscription** object, use the Get-DPMCloudSubscription cmdlet.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-WorkDay<DayOfWeek[]>

Specifies the days of the week when Windows Azure Online Backup runs. Valid values for this parameter are:

-- Sunday

-- Monday

-- Tuesday

-- Wednesday

-- Thursday

-- Friday

-- Saturday

The acceptable values for this parameter are:

Sunday	
Monday	
Tuesday	
Wednesday	
Thursday	
Friday	
Saturday	

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-WorkHourBandwidth<UInt32>

Specifies the bandwidth that Windows Azure Online Backup uses during working hours.

Aliases	none
Required?	true

Position?	6
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Specify Windows Azure Online Backup options for a DPM server

This example sets subscription options for Windows Azure Online Backup for the DPM server named TestingServer.

The first command gets the subscription settings for the server named TestingServer and stores the settings in the $Setting variable.

The second command specifies C:\StagingArea as the staging area for backup files from TestingServer.

The third command indicates that the backup does not use a proxy server.

The fourth command converts the string passphrase123456789 to a secure string and assigns the secure string to the variable named $Passphrase.

The fifth command sets the secure string in $Passphrase as the password for encrypting backups from TestingServer.

The sixth command saves the backup options that you set in previous commands.

```
PS C:\> $Setting = Get-DPMCloudSubscriptionSetting -DPMServerName "TestingServer"
PS C:\> Set-DPMCloudSubscriptionSetting -DPMServerName "TestingServer" -SubscriptionSetting
$Setting -StagingAreaPath "C:\StagingArea"
PS C:\> Set-DPMCloudSubscriptionSetting -DPMServerName "TestingServer" -SubscriptionSetting
$Setting -NoProxy
PS C:\> $Passphrase = ConvertTo-SecureString -string "passphrase123456789" -AsPlainText -
Force
PS C:\> Set-DPMCloudSubscriptionSetting -DPMServerName "TestingServer" -SubscriptionSetting
$Setting -EncryptionPassphrase $Passphrase
PS C:\> Set-DPMCloudSubscriptionSetting -DPMServerName "TestingServer" -SubscriptionSetting
$Setting -Commit
```

Related topics

Get-DPMCloudSubscription

Get-DPMCloudSubscriptionSetting

Set-DPMCredentials

Set-DPMCredentials

Configures certificate-based authentication for computers on untrusted domains.

Syntax

```
Parameter Set: Default
Set-DPMCredentials [[-DPMServerName] <String> ] [-Type] {Certificate} [-Action] {Configure |
Regenerate} [-OutputFilePath] <String> [[-Thumbprint] <String> ] [[-AuthCAThumbprint]
<String> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMCredentials** cmdlet configures certificate-based authentication for computers that you want to protect but that are in untrusted domains.

Parameters

-Action<Action>

Specifies the action to perform. Valid values for this parameter are:

-- Configure: the configuration uses a new certificate

-- Regenerate: the configuration regenerates the output metadata file

The acceptable values for this parameter are:

Configure	
Regenerate	

Aliases	none
Required?	true
Position?	3
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AuthCAThumbprint<String>

Specifies the thumbprint of a certifying authority in the trust chain of the certificate. If you do not specify this parameter, System Center 2012 – Data Protection Manager (DPM) uses the value Root.

Aliases	none
Required?	false
Position?	6
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of the DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-OutputFilePath<String>

Specifies the location of the output file. Specify this path by using the Set-DPMServer tool to complete configuration for protection agents on the DPM server.

Aliases	none
Required?	true

Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Thumbprint<String>

Specifies the thumbprint to use when you search for the certificate. You must specify this parameter if you specify Configure as the value for the *Action* parameter.

Aliases	none
Required?	false
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Type<AuthenticationType>

Specifies the type of credential that you are adding. This parameter only accepts the single value Certificate.

The acceptable values for this parameter are:

Certificate	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Configure authentication with a new certificate

This command configures authentication for the DPM server named Dpmserver.Contoso.com. The command creates a file in the folder C:\CertMetaData\ with the name CertificateConfiguration_<DPM-server-FQDN>.bin.

```
PS C:\> Set-DPMCredentials -DPMServerName "Dpmserver.Contoso.com" -Type Certificate -Action
Configure -OutputFilePath "C:\CertMetaData\" –Thumbprint
"cf822d9ba1c801ef40d4b31de0cfcb200a8a2496"
```

Example 2: Configure authentication with a regenerated configuration file

This cmdlet configures the DPM server named Dpmserver.Contoso.com by regenerating the lost configuration file in the folder C:\CertMetaData\.

```
PS C:\> Set-DPMCredentials -DPMServerName "Dpmserver.Contoso.com" -Type Certificate -
OutputFilePath "C:\CertMetaData\" -Action Regenerate
```

Set-DPMDatasourceDefaultDiskAllocation

Set-DPMDatasourceDefaultDiskAllocation

Retrieves the amount of disk space that is allocated to protected data.

Syntax

```
Parameter Set: Default
Set-DPMDatasourceDefaultDiskAllocation [-Datasource] <Datasource[]> [-Async] [-
CalculateShrinkThresholds] [-CalculateSize] [-PrimaryDpmServer] [-Tag <Object> ] [
<CommonParameters>]
```

Detailed Description

The **Set-DPMDatasourceDefaultDiskAllocation** cmdlet modifies disk allocation for protected data.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish. This parameter is useful if the cmdlet takes a long time to complete or if you build a graphical user interface (GUI) by using cmdlets. Do not use this parameter if you are working with the Data Protection Manager (DPM) Management Shell.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-CalculateShrinkThresholds

Indicates that DPM must calculate the thresholds to which the recovery point volume can shrink.

This parameter sets the properties **ShadowCopySizeAfterMaxShrink** and **ShadowCopySizeAfterMinShrink**. When you use this parameter, the Get-DiskAllocation cmdlet returns two values. To shrink the recovery point volume, specify a value between the two returned values to the *ShadowCopyArea* parameter of the Set-DiskAllocation cmdlet. However, if you cannot shrink the recovery point volume, the cmdlet throws an exception with the appropriate error code.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-CalculateSize

Indicates that DPM calculates the space allocated on a disk.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource[]>

Specifies a share or volume in a Windows file system, Microsoft SQL Server database, Microsoft Exchange storage group, Microsoft SharePoint farm, Microsoft virtual machine, DPM database, or system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1

Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-PrimaryDpmServer

Indicates that DPM recovers data to a DPM server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tag<Object>

Specifies a custom property that distinguishes the replies to each asynchronous call.

This parameter is useful if you build a GUI by using cmdlets.

Do not use this parameter if you work with the DPM Management Shell.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Set-DPMDatasourceDefaultDiskAllocation detailed`".

Related topics

Edit-DPMDiskAllocation

Set-DPMDatasourceDiskAllocation

Set-DPMDatasourceDiskAllocation

Set-DPMDatasourceDiskAllocation

Changes disk allocation for protected data.

Syntax

```
Parameter Set: Default
Set-DPMDatasourceDiskAllocation [-Datasource] <Datasource> [-ProtectionGroup]
<ProtectionGroup> [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: AutoGrowOption
Set-DPMDatasourceDiskAllocation [-ProtectionGroup] <ProtectionGroup> [-AutoGrow] <Boolean>
[-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Custom
Set-DPMDatasourceDiskAllocation [-Datasource] <Datasource> [-ProtectionGroup]
<ProtectionGroup> -CustomRequirements [-FormatVolumes] [-PassThru] [-ReplicaVolume
<DpmServerVolume> ] [-ShadowCopyVolume <DpmServerVolume> ] [-USNJournalSize <Int64> ] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ExpectedDataSizePerClientInMB
Set-DPMDatasourceDiskAllocation -ExpectedDataSizePerClientInMB <Int64> -ProtectionGroup
<ProtectionGroup> [-Datasource <Datasource> ] [-ShadowCopyArea <Int64> ] [-Confirm] [-
WhatIf] [ <CommonParameters>]

Parameter Set: Manual
Set-DPMDatasourceDiskAllocation [-Datasource] <Datasource> [-ProtectionGroup]
<ProtectionGroup> -Manual [-PassThru] [-ProductionServerJournalSize <Int64> ] [-ReplicaArea
<Int64> ] [-ShadowCopyArea <Int64> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: MigrateDatasourceDataFromDPMToDisk
Set-DPMDatasourceDiskAllocation [-Datasource] <Datasource> [-ProtectionGroup]
<ProtectionGroup> [-MigrateDatasourceDataFromDPM] [-DestinationDiskPool] <Disk[]> [-Confirm]
[-WhatIf] [ <CommonParameters>]

Parameter Set: MigrateDatasourceDataFromDPMToVolume
Set-DPMDatasourceDiskAllocation [-Datasource] <Datasource> [-ProtectionGroup]
<ProtectionGroup> [-MigrateDatasourceDataFromDPM] [-DestinationReplicaVolume]
<DpmServerVolume> [-DestinationShadowCopyVolume] <DpmServerVolume> [-FormatVolumes] [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMDatasourceDiskAllocation** cmdlet changes disk allocation for protected data. By default, Data Protection Manager (DPM) allocates disk space and volumes for the data source. The cmdlet gives you these options for changing disk allocation:

-- Specify the amount of disk space to allot to each data source in a protection group.

-- Specify the volumes to use for each data source in a protection group.

Parameters

-AutoGrow<Boolean>

Indicates whether DPM increases the size of the replica or recovery point volume automatically when it runs out of user space.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-CustomRequirements

Indicates that you can specify replica and shadow copy volumes manually.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource>

Specifies a share or volume in a Windows file system, Microsoft SQL Server database, Microsoft Exchange storage group, Microsoft SharePoint farm, Microsoft virtual machine, DPM database, or system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DestinationDiskPool<Disk[]>

Specifies the set of disks on which DPM allocates volumes to migrate a data source. This parameter is migration specific.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DestinationReplicaVolume<DpmServerVolume>

Specifies the destination volume that the user allocates to migrate replicas of a data source. This parameter is migration specific.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-DestinationShadowCopyVolume<DpmServerVolume>

Specifies the destination volume that the user allocates to migrate shadow copies of a data source. This parameter is migration specific.

Aliases	none
Required?	true
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExpectedDataSizePerClientInMB<Int64>

Specifies the expected size of data that the user will back up from each client computer (in megabytes).

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-FormatVolumes

Indicates that DPM formats the allocated volumes.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Manual

Indicates that you can apply DPM settings manually.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MigrateDatasourceDataFromDPM

Indicates that DPM migrates the data source from the current volumes to a new set of volumes.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that cmdlets can be part of a pipeline. Use this parameter with other DPM cmdlets to return a related object in cases where the cmdlet does not create default output.

Aliases	none
Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProductionServerJournalSize<Int64>

Specifies the journal size of the protected server.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group object.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ReplicaArea<Int64>

Specifies the disk allocation for the replica area of the current data source.

Aliases	none
Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ReplicaVolume<DpmServerVolume>

Specifies a replica volume object. A replica volume is a volume on the DPM server that contains the replica of a protected data source.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ShadowCopyArea<Int64>

Specifies the disk allocation for the shadow copy area of the current data source.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ShadowCopyVolume<DpmServerVolume>

Specifies an object for a shadow copy volume. This object represents the volume that contains the shadow copy.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-USNJournalSize<Int64>

Specifies the journal size for the update sequence number (USN).

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Datasource**

Notes

- For more information, type `Get-Help Set-DPMDatasourceDiskAllocation -detailed`.

Examples

Example 1: Specify disk allocation for a data source

This example specifies the disk allocation for a data source.

The first command gets all protection groups from the DPM server named TestingServer and stores these groups in the variable named $Pg. You cannot edit these protection groups.

The second command gets the protection groups from the $Pg array in editable mode. It stores the editable protection group in the variable named $Mpg.

The third command gets the list of protected and unprotected data from the $Mpg array and stores them in the variable named $Ds.

The fourth command gets the current disk allocation for the second data source in $Ds.

The fifth command sets the disk allocation for the second data source in $Ds from the protection groups in $Mpg.

The sixth command saves the second data source in $Ds to the editable protection group in $Mpg.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "TestingServer"
PS C:\> $Mpg = Get-ModifiableProtectionGroup $Pg
PS C:\> $Ds = Get-Datasource -ProtectionGroup $Mpg
PS C:\> Get-DatasourceDiskAllocation -Datasource $Ds[1]
PS C:\> Set-DPMDatasourceDiskAllocation -Datasource $Ds[1] -ProtectionGroup $Mpg
PS C:\> Set-ProtectionGroup $Mpg
```

Related topics

Set-DPMDatasourceDefaultDiskAllocation

Set-DPMDatasourceProtectionOption

Set-DPMDatasourceProtectionOption

Sets protection options for a DPM data source.

Syntax

```
Parameter Set: AllowClientUserToAddFileSpecs
Set-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> [-
AllowClientUserToAddFileSpecs] <Boolean> [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: AutoConsistencyCheck
Set-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> [-
AutoConsistencyCheck] <Boolean> [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ClientFileSpecOptions
Set-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> [-Path] <String> [-
ClientFileSpecOperation] <ClientFileSpecOperationType> [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: ExchangeOptions
Set-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> [[-TopologyType]
<ProtectionTopologyType> ] [[-PreferredPhysicalNode] <String[]> ] -ExchangeOptions [-
BackUpType <AllowedBackupTypeType> ] [-Datasource <Datasource> ] [-EseutilCheckType
<EseutilTypeType> ] [-PassThru] [-RunEseUtilConsistencyCheck] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: FileExclusionAddOptions
Set-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> [-FileType] <String>
-Add [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: FileExclusionRemoveOptions
Set-DPMDatasourceProtectionOption [-ProtectionGroup] <ProtectionGroup> [-FileType] <String>
-Remove [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMDatasourceProtectionOption** cmdlet sets protection options for a Data Protection Manager (DPM) data source. You can set the following protection options:

-- Exclude all files of a particular type (for example, all MP3 files) from protection.

-- Remove exclusions.

-- Set protection options for a Microsoft Exchange data source.

Parameters

-Add

Indicates that the cmdlet adds a file exclusion to a protection group.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AllowClientUserToAddFileSpecs<Boolean>

Indicates whether the user can specify which folders to protect.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AutoConsistencyCheck<Boolean>

Indicates whether DPM automatically runs consistency checks on the data sources of a protection group if a replica becomes inconsistent.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-BackUpType<AllowedBackupTypeType>

Indicates the type of backup. Valid values for this parameter are:

-- CopyBackup

-- FullBackup.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ClientFileSpecOperation<ClientFileSpecOperationType>

Indicates which folders to add to the inclusion list or remove from the exclusion list. Valid values for this parameter are:

-- AddInclude

-- RemoveInclude

-- AddExclude

-- RemoveExclude

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource>

Specifies a data source object. A data source can be a Windows file system share or volume, SQL Server database, Microsoft Exchange storage group, Microsoft SharePoint farm, virtual machine, DPM database, or system state that is a member of a protection group.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-EseutilCheckType<EseutilTypeType>

Indicates how to run the Exchange database repair tool Eseutil. Valid values for this parameter are:

-- DontRun

-- RunOnLogs

-- RunOnLogsAndDB.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExchangeOptions

Indicates that the options that follow affect only Microsoft Exchange data sources.

Aliases	none
Required?	true
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-FileType<String>

Indicates a file type to include in or exclude from a protection group.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Path<String>

Specifies the path to a folder to add to the inclusion list or remove from the exclusion list.

Aliases	none
Required?	true
Position?	2

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PreferredPhysicalNode<String[]>

Specifies an array of names of servers to protect for Preferred Server topology. This parameter applies only to servers that use cluster continuous replication in Microsoft Exchange Server 2007.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Remove

Indicates that the cmdlet performs a remove operation.

Aliases	none
Required?	true

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RunEseUtilConsistencyCheck

Indicates that the cmdlet runs Eseutil consistency checking. This parameter applies only to Microsoft Exchange data sources.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TopologyType<ProtectionTopologyType>

Specifies the topology type. Valid values for this parameter are:

-- Active

-- Passive

-- Active if Passive Not Available

This parameter applies only to Microsoft Exchange Server 2007 installations that run cluster continuous replication.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Datasource**

Examples

Example 1: Set a consistency check for a protection group

This example sets a consistency check for a protection group.

The first command gets the protection groups from the server named DPMServer02 and stores the result in the $Pg variable.

The second command sets an automatic consistency check to run on all protection groups in $Pg.

```
PS C:\> $Pg = Get-DPMProtectionGroup –DPMServerName "DPMServer02"
PS C:\> Set-DPMDatasourceProtectionOption -ProtectionGroup $Pg -AutoConsistencyCheck $True
```

Related topics

Get-DPMDatasourceProtectionOption

Set-DPMGlobalProperty

Set-DPMGlobalProperty

Sets the global properties for a DPM installation.

Syntax

```
Parameter Set: AllowLocalDataProtection
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -AllowLocalDataProtection <Boolean> [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ConsiderForAutoDeployment
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -ConsiderForAutoDeployment <Boolean> [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ExchangeSCRProtection
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -ExchangeSCRProtection <String> [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: HyperVPagefileExclusions
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -HyperVPagefileExclusions <String> [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: IsNetworkChecksumRequired
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -IsNetworkChecksumRequired <Boolean> [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: KnownVMMServers
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -KnownVMMServers <String> [-Confirm] [-
WhatIf] [ <CommonParameters>]

Parameter Set: LibraryRefreshInterval
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -LibraryRefreshInterval <Int32> [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: MaxCapacityForClientAutoDeployment
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -MaxCapacityForClientAutoDeployment
<Int32> [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: RegisteredWriters
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -RegisteredWriters <Guid[]> [-Confirm] [-
WhatIf] [ <CommonParameters>]

Parameter Set: TruncateSharePointDbLogs
Set-DPMGlobalProperty [[-DPMServerName] <String> ] -TruncateSharePointDbLogs <Boolean> [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMGlobalProperty** cmdlet sets the global properties for a System Center 2012 – Data Protection Manager (DPM) installation.

Parameters

-AllowLocalDataProtection<Boolean>

Indicates whether a DPM server can protect data sources on the same computer on which it is installed.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ConsiderForAutoDeployment<Boolean>

Indicates whether to consider a DPM server for client protection.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExchangeSCRProtection<String>

Specifies the name of an Exchange Standby Continuous Replication (SCR) Server. To specify the names of multiple servers, list them individually, separated by commas.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-HyperVPagefileExclusions<String>

Specifies the name of a VHD pagefile to exclude from backup jobs for a Hyper-V™ data source. To specify the names of multiple VHD pagefiles, list them individually, separated by commas. This parameter also accepts wildcard values.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-IsNetworkChecksumRequired<Boolean>

Indicates whether to apply a checksum to verify data transferred over a network.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-KnownVMMServers<String>

Specifies a Virtual Machine Manager (VMM) server that communicates with DPM to provide support for virtual machine mobility scenarios. If you specify this parameter, DPM stores the VMM server information and configures its services to communicate with the VMM.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-LibraryRefreshInterval<Int32>

Specifies the refresh interval for gathering information about shared libraries.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MaxCapacityForClientAutoDeployment<Int32>

Specifies the maximum number of client computers that a DPM server can protect. Set this value to ensure that DPM protects a reasonable number of client computers if auto-deployment is enabled.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RegisteredWriters<Guid[]>

For internal use only.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TruncateSharePointDbLogs<Boolean>

Indicates whether to truncate SharePoint database logs during replication.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Set-DPMGlobalProperty -detailed`".

Examples

Example 1: Exclude page file VHDs

This command excludes all protected VHDs with names that include the string _pagefile.vhd on the server named Contoso-DPMServer.

```
PS C:\> Set-DPMGlobalProperty -DPMServerName "Contoso-DPMServer" -HyperVPagefileExclusions
"*_pagefile.vhd"
```

Example 2: Exclude VHDs using wildcard expressions exclusions

This command excludes VHD files with the strings mypgf and pagefile.vhd in their names.

```
PS C:\> Set-DPMGlobalProperty -DPMServerName "Contoso-DPMServer" -HyperVPagefileExclusions
"*_pagefile.vhd,*mypgf*.vhd*"
```

Example 3: Remove all VHD exclusions

This command removes all existing exclusions by supplying an empty list to the *HyperVPagefileExclusions* parameter.

```
PS C:\> Set-DPMGlobalProperty -DPMServerName "Contoso-DPMServer" -HyperVPagefileExclusions
""
```

Related topics

Get-DPMGlobalProperty

Set-DPMMaintenanceJobStartTime

Set-DPMMaintenanceJobStartTime

Sets the start time of a maintenance job or stops such a job from running.

Syntax

```
Parameter Set: Reset
Set-DPMMaintenanceJobStartTime [[-DPMServerName] <String> ] [-MaintenanceJob]
{CatalogPruning | LibraryInventory} [[-StartTime] <DateTime> ] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: Remove
Set-DPMMaintenanceJobStartTime [[-DPMServerName] <String> ] [-MaintenanceJob]
{CatalogPruning | LibraryInventory} [-Remove] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMMaintenanceJobStartTime** cmdlet sets the start time of a maintenance job or stops such a job if it is running in System Center 2012 – Data Protection Manager (DPM).

You can reschedule Catalog Pruning and Detailed Inventory jobs by using this cmdlet.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MaintenanceJob<HouseKeepingJobs>

Specifies the maintenance job that the cmdlet performs on the replica. The acceptable values for this parameter are:

-- CatalogPruning. Removes index entries for expired tapes.

-- DetailedInventory. Identifies new tapes and recognizes tapes DPM has seen before by reading the on-media identifier (OMID) on each tape.

The acceptable values for this parameter are:

CatalogPruning	
LibraryInventory	

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Remove

Indicates that the cmdlet stops the job that is currently running.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-StartTime<DateTime>

Specifies the time when the operation begins as a **DateTime** object.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Set-DPMMaintenanceJobStartTime -detailed`".

Examples

Example 1: Set the start time for a job

This command schedules the CatalogPruning maintenance job to run on the DPM server named DPMServer07 at 2 a.m.

```
PS C:\> Set-DPMMaintenanceJobStartTime -DPMServerName "DPMServer07" -MaintenanceJob
CatalogPruning -StartTime 02:00
```

Example 2: Stop a running job

This command stops the LibraryInventory job from running on the DPM server named DPMServer07.

```
PS C:\> Set-DPMMaintenanceJobStartTime -DPMServerName "DPMServer07" -MaintenanceJob
LibraryInventory -Remove
```

Related topics

Set-DPMMaintenanceJobStartTime

Set-DPMPerformanceOptimization

Set-DPMPerformanceOptimization

Enables or disables on-the-wire compression for a DPM protection group.

Syntax

```
Parameter Set: DisableCompression
Set-DPMPerformanceOptimization [-ProtectionGroup] <ProtectionGroup> -DisableCompression [-
PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: EnableCompression
Set-DPMPerformanceOptimization [-ProtectionGroup] <ProtectionGroup> -EnableCompression [-
PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMPerformanceOptimization** cmdlet enables or disables on-the-wire compression of data for a System Center 2012 – Data Protection Manager (DPM) protection group. You can use the **Get-DPMProtectionGroup** cmdlet to get a protection group, and use the **Get-DPMModifiableProtectionGroup** cmdlet to allow changes for a protection group.

On-the-wire compression decreases the amount of data transferred during replica creation, synchronization, and consistency check operations. On-the-wire compression increases CPU usage on both the DPM server and on protected computers.

Parameters

-DisableCompression

Indicates that the cmdlet disables on-the-wire compression of data.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-EnableCompression

Indicates that the cmdlet enables on-the-wire compression of data.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group object. To obtain a protection group object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1

Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Set-DPMPerformanceOptimization -detailed`".

Examples

Example 1: Enable compression for a protection group

This example enables on-the-wire compression of data for all data sources protected in the specified protection group. The first command uses the **Get-DPMProtectionGroup** cmdlet to get the protection group for the DPM server named DPMServer07, and stores that object in the $DPMPG variable.

The second command uses the **Get-DPMModifiableProtectionGroup** cmdlet to get the protection group stored in the $DPMPG variable in an editable format, and stores that object in the $DPMMPG variable.

The third command enables compression for the protection group stored in the $DPMMPG variable.

```
PS C:\> $DPMPG=Get-DPMProtectionGroup -DPMServerName "DPMServer07"
PS C:\> $DPMMPG = Get-DPMModifiableProtectionGroup -ProtectionGroup $DPMPG
PS C:\> Set-DPMPerformanceOptimization -ProtectionGroup $DPMMPG -EnableCompression
```

Related topics

Get-DPMProtectionGroup

Get-DPMModifiableProtectionGroup

Set-DPMPolicyObjective

Set-DPMPolicyObjective

Sets the policy objective for a protection group.

Syntax

```
Parameter Set: ShortTermDisk
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-RetentionRangeDays] <Int32>
[[-SynchronizationFrequencyMinutes] <Int32> ] [-BeforeRecoveryPoint] [-PassThru] [-Confirm]
[-WhatIf] [ <CommonParameters>]

Parameter Set: AlertThresholdInDays
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-AlertThresholdInDays] <Int32>
[-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ColocateDatasources
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-ColocateDatasources] <Boolean>
[-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: LongTermCloud
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-OnlineRetentionRange] <Int32>
[-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: LongTermTape
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-RetentionRange]
<RetentionRange> [-LongTermBackupFrequency] {Daily | Weekly | BiWeekly | Monthly | Quarterly
| HalfYearly | Yearly} [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: LongTermTapeCustom
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-RetentionRangeList]
<RetentionRange[]> [-FrequencyList] <Int32[]> [-GenerationList] {GreatGrandfather |
Grandfather | Father | Son} [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ShortTermTape
Set-DPMPolicyObjective [-ProtectionGroup] <ProtectionGroup> [-RetentionRangeInWeeks] <Int32>
[-ShortTermBackupFrequency] {Daily | Weekly | BiWeekly | Monthly | Quarterly | HalfYearly |
Yearly} [-CreateIncrementals] [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMPolicyObjective** cmdlet sets the policy objective for a System Center 2012 – Data Protection Manager (DPM) protection group. You can set the policy to back up to disk or tape and specify a retention range for the protection group.

DPM sets default schedules automatically if you specify the protection objective by using the **Set-DPMPolicyObjective** cmdlet. To change the default schedules, use the **Get-PolicySchedule** cmdlet followed by the **Set-DPMPolicySchedule** cmdlet.

Parameters

-AlertThresholdInDays<Int32>

Specifies the number of days after the most recent backup that DPM waits before it raises an alert.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-BeforeRecoveryPoint

Indicates that DPM runs synchronization before it creates recovery points.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ColocateDatasources<Boolean>

Indicates whether DPM enables colocation.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-CreateIncrementals

Indicates that DPM creates incremental backups.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-FrequencyList<Int32[]>

Specifies an array of backup frequencies that the three protection objectives define. The frequency list can contain any combination of the following values:

-- 6 (yearly)

-- 5 (half yearly)

-- 4 (quarterly)

-- 3 (monthly)

-- 2 (biweekly)

-- 1 (weekly)

-- 0 (daily)

Aliases	none
Required?	true
Position?	3
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-GenerationList<GenerationType[]>

Specifies an array of generations that the three protection objectives define. The generation list can include any combination of these values:

-- GreatGrandfather

-- Grandfather

-- Father

-- Son

The acceptable values for this parameter are:

GreatGrandfather	
Grandfather	
Father	
Son	

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-LongTermBackupFrequency<BackupFrequency>

Specifies the backup frequency for long-term protection. Vaid values for this parameter are:

-- Daily

-- Weekly

-- BiWeekly

-- Monthly

-- Quarterly

-- HalfYearly

-- Yearly

The acceptable values for this parameter are:

Daily	
Weekly	
BiWeekly	
Monthly	
Quarterly	
HalfYearly	
Yearly	

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-OnlineRetentionRange<Int32>

Specifies the retention range for online protection.

Aliases	none
Required?	true

Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies the name of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-RetentionRange<RetentionRange>

Specifies the duration for which DPM keeps data on tape for long-term protection.

Aliases	none

Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RetentionRangeDays<Int32>

Specifies the number of days for which DPM keeps a replica.

Aliases	RetentionRangeInDays
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RetentionRangeInWeeks<Int32>

Specifies the number of weeks for which DPM keeps a replica.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RetentionRangeList<RetentionRange[]>

Specifies an array of retention periods that the three protection objectives define.

Aliases	none

Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ShortTermBackupFrequency<BackupFrequency>

Specifies the backup frequency for short-term protection. Valid values for this parameter are:

-- Weekly

-- Daily

-- Weekly

-- BiWeekly

-- Monthly

-- Quarterly

-- HalfYearly

-- Yearly

The acceptable values for this parameter are:

Daily	
Weekly	
BiWeekly	
Monthly	
Quarterly	
HalfYearly	
Yearly	

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-SynchronizationFrequencyMinutes<Int32>

Specifies the frequency of synchronization, in minutes.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Schedule**

Notes

- For more information, type "`Get-Help Set-DPMPolicyObjective -detailed`".
  For technical information, type "`Get-Help Set-DPMPolicyObjective -full`".

Examples

Example 1: Set the policy objective for a protection group

This example sets the policy objective for a protection group.

The first command gets the protection group objects from the DPM server named DPMServer02 and stores the protection group objects in the $Pg variable.

The second command uses standard array notation to specify the second member of the $Pg array (at element 1) in editable format, and stores the protection group in the $Mpg variable.

The third command sets the policy objective for the protection group in the $Mpg variable to a retention range of 10 days and synchronization frequency of 6 hours (360 minutes).

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Mpg = Get-ModifiableProtectionGroup $Pg[0]
PS C:\> Set-DPMPolicyObjective $Mpg -RetentionRangeInDays 10 -SynchronizationFrequency 360
```

Example 2: Set the retention range for a DPM server

This example sets the retention range for frequency DPM server.

The first command creates a **RetentionRange** object.

The second command gets protection groups from the DPM server named DPMServer02 and stores the protection groups in the $Pg variable.

The third command uses standard array notation to specify the first member of the $Pg array (at position 0) in editable form, and stores the editable protection group in the $Mpg variable.

The fourth command sets the retention range for the protection group to 360 days and the long-term backup frequency to weekly.

```
PS C:\> $rr = new-object -TypeName
Microsoft.Internal.EnterpriseStorage.Dls.UI.ObjectModel.OMCommon.RetentionRange -
ArgumentList 1,"months"
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
```

```
PS C:\> $Mpg = Get-ModifiableProtectionGroup $Pg[0]
PS C:\> Set-DPMPolicyObjective -RetentionRange 360 -LongTermBackupFrequency Weekly $Mpg
```

Example 3: Set a retention range

This example shows another way to set a retention range for a DPM server.

The first command gets protection groups from the DPM server named DPMServer02 and stores the protection groups in the $Pg variable.

The second command uses standard array notation to specify the first member (at position 0) of the $Pg array in editable form, and stores the editable protection group in the $Mpg variable.

The third command sets the retention range for the protection group to 12 weeks and the short-term backup frequency to daily.

```
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Mpg = Get-ModifiableProtectionGroup $Pg[0]
PS C:\> Set-DPMPolicyObjective -RetentionRangeInWeeks 12 -ShortTermBackupFrequency Daily
$Mpg
```

Related topics

Get-DPMPolicyObjective

Get-DPMPolicySchedule

Set-DPMPolicySchedule

Set-DPMPolicySchedule

Set-DPMPolicySchedule

Sets the schedule for protection jobs.

Syntax

```
Parameter Set: CheckDataIntegrity
Set-DPMPolicySchedule [-ProtectionGroup] <ProtectionGroup> [-JobType] <ProtectionJobType> [-
PassThru] [-Remove] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ModifySchedule
Set-DPMPolicySchedule [-ProtectionGroup] <ProtectionGroup> [-Schedule] <Schedule> [-
PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: OffsetSchedule
Set-DPMPolicySchedule [-ProtectionGroup] <ProtectionGroup> [-OffsetInMinutes] <Int32> [-
PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMPolicySchedule** cmdlet sets the schedule for protection jobs in System Center 2012 – Data Protection Manager (DPM).

DPM sets default schedules automatically if you specify the protection objective by using the **Set-PolicyObjective** cmdlet. To change the default schedules, use the **Get-PolicySchedule** cmdlet followed by the **Set-DPMPolicySchedule** cmdlet.

You can set the following schedules.

For express full and shadow copy schedule:

[-TimesOfDay] <TimesOfDay> [-DaysOfWeek]<DaysOfWeek>

Example: 6:00 AM, 12:00 AM We,Th

For monthly, half-yearly, and yearly archive schedules:

[-StartTime] <StartTime> [-RelativeInterval <RelativeInterval>] [-DayOfWeek <DayOfWeek>]

Example: -StartTime 1/1/2003 6:00 AM -RelativeInterval First -DayOfWeek Sa

For daily archive schedule:

[-TimeOfDay] <TimeOfDay>

Example: -StartTime 6:00 AM

For quarterly archive schedule:

[-StartTime] <StartTime>

Example: -StartTime 1/1/2003 6:00 AM

The input schedule determines the frequency.

Parameters

-JobType<ProtectionJobType>

Indicates the type of job for which you set options. This parameter can take the single value ConsistencyCheck.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-OffsetInMinutes<Int32>

Specifies the time, in minutes, by which to offset the start time of a job.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Remove

Indicates that the cmdlet removes a schedule for a protection operation.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Schedule<Schedule>

Specifies a schedule object. A schedule describes the recurrence of a backup job. Each job type has one schedule, which DPM triggers. You create a default schedule by using the **Set-PolicyObjective** cmdlet.

Customize a default schedule by using the **Get-PolicySchedule** cmdlet and passing the schedule to **Set-DPMPolicySchedule**, specifying the backup recurrence.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Get-PolicySchedule -detailed`".
  For technical information, type "`Get-Help Get-PolicySchedule -full`".

Examples

Example 1: Create a synchronization schedule for a protection group

This example creates a synchronization schedule for a protection group. DPM runs synchronization for the protection group every Monday at 2:00 A.M.

The first command gets the protection group from the DPM server named DPMServer02, and stores the result in the $Pg variable.

The second command gets the short-term schedule for performing integrity checks from the protection group stored in $Pg, and stores the result in the $ShadowCopysch variable.

The third command sets synchronization to run at 2:00 A.M. every Monday.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $ShadowCopysch = Get-PolicySchedule $Pg -ShortTerm
PS C:\> Set-DPMPolicySchedule $Pg $ShadowCopysch -DaysOfWeek mo -TimesOfDay 02:00
```

Example 2: Change a synchronization schedule for a protection group

This example changes a synchronization schedule for a protection group.

The first command gets protection groups from the DPM server named ContosoDPM1, and stores the result in the $Pg variable.

The second command gets the long-term synchronization schedule for the second protection group (at element 1) stored in $Pg, and stores the result in the $SC variable.

The third command gets the second protection group in editable form stored in $Pg, and stores the result in the $Mpg variable.

The fourth command sets the policy schedule for the third schedule (at element 2) stored in $SC to run at 7:00 P.M. daily.

The fifth command uses the **Set-DPMProtectionGroup** cmdlet to store the protection group that has the modified synchronization schedule.

```
PS C:\> $Pg = Get-DPMProtectionGroup -dpmservername "ContosoDPM1"
PS C:\> $SC = Get-DPMPolicySchedule $Pg[1] -longterm
PS C:\> $Mpg = Get-DPMModifiableProtectionGroup -ProtectionGroup $Pg[1]
```

```
PS C:\> Set-DPMPolicySchedule -ProtectionGroup $Mpg -Schedule $sc[2] -TimeOfDay "07:00 PM"
PS C:\> Set-DPMProtectionGroup -ProtectionGroup $Mpg
```

Example 3: Schedule an integrity check for a DPM server

This example creates a schedule for checking the integrity of a DPM server.

The first command gets the protection group from the DPM server named ContosoDPM1, and stores the result in the $Pg variable.

The second command gets the protection group in editable form stored in $Pg, and stores the result in the $Mpg variable.

The third command schedules a data integrity check to run on the server every Monday at 6:00 P.M.

The fourth command uses the **Set-DPMProtectionGroup** cmdlet to store the protection group that has the new integrity-check schedule.

```
PS C:\> $Pg = Get-DPMProtectionGroup "ContosoDPM1"
PS C:\> $Mpg = Get-DPMModifiableProtectionGroup -ProtectionGroup $Pg
PS C:\> Set-DPMPolicySchedule -ProtectionGroup $Mpg -JobType CheckDataIntegrity -DaysOfWeek
mo -TimesOfDay 18:00
PS C:\> Set-DPMProtectionGroup -ProtectionGroup $Mpg
```

Related topics

Get-DPMPolicySchedule

Get-DPMPolicyObjective

Set-DPMPolicyObjective

Set-DPMProtectionGroup

Set-DPMProtectionGroup

Saves all the actions performed on the protection group on the DPM server.

Syntax

```
Parameter Set: Default
Set-DPMProtectionGroup [-ProtectionGroup] <ProtectionGroup> [-Async] [-TranslateDSList
<Datasource[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMProtectionGroup** cmdlet saves all the actions that you performed on the protection group on the System Center 2012 – Data Protection Manager (DPM) server.

Until you run the **Set-DPMProtectionGroup** cmdlet, the actions that you take on a modifiable protection group or a new protection group exist only in the memory. For example, when you create a protection group, use this cmdlet as the final step in the process.

Use the following sequence of steps to modify a protection group:

-- 1. Get-DPMProtectionGroup

-- 2. Get-DPMModifiableProtectionGroup

-- 3. Perform actions on the protection group

-- 4. Set-DPMProtectionGroup

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-TranslateDSList<Datasource[]>

Specifies an array of data sources. The translation of the data sources that you specify is forced. This helps to regenerate jobs.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Set-DPMProtectionGroup -detailed`".

Examples

Example 1: Save a protection group

This example saves a protection group on the DPM server.

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command saves the protection group stored in the $Pg variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Set-DPMProtectionGroup -ProtectionGroup $Pg
```

Related topics

Update-DPMProtectionGroup

Get-DPMProtectionGroup

Rename-DPMProtectionGroup

New-DPMProtectionGroup

Rename-DPMProtectionGroup

Set-DPMProtectionJobStartTime

Set-DPMProtectionJobStartTime

Sets the start time of a protection job.

Syntax

```
Parameter Set: Edit
Set-DPMProtectionJobStartTime [-ProtectionGroup] <ProtectionGroup> [-JobType]
<ProtectionJobType> [-StartTime] <DateTime> [-MaximumDurationInHours] <Int32> [-PassThru] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: OffsetSchedule
Set-DPMProtectionJobStartTime [-ProtectionGroup] <ProtectionGroup> [-CatalogOffset] <Int32>
[-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Remove
Set-DPMProtectionJobStartTime [-ProtectionGroup] <ProtectionGroup> [-JobType]
<ProtectionJobType> -Remove [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMProtectionJobStartTime** cmdlet sets the start time of a protection job. The only type of protection job that is currently available is consistency check. Use the *Remove* parameter to stop a protection job.

Parameters

-CatalogOffset<Int32>

Specifies the number of minutes, after the first scheduled recovery point creation time of the SharePoint farm, that DPM creates the catalog.

Aliases	none
Required?	true
Position?	2
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-JobType<ProtectionJobType>

Specifies the type of job for which options are being set. The only valid value for this parameter is ConsistencyCheck.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-MaximumDurationInHours<Int32>

Specify the maximum number of hours that DPM can run a job.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Remove

Indicates that DPM stops the protection job.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-StartTime<DateTime>

Specifies a start time for the protection job.

Aliases	none

Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Get-ProtectionJobStartTime -detailed`".

Examples

Example 1: Schedule a consistency check on a protection group

This example schedules a consistency check to run on a protection group every day.

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command makes the protection group stored in the $Pg variable modifiable, and stores the result in the $Mpg variable.

The third command sets the start time of the protection job stored in the $Mpg variable. The command schedules for a consistency check to run on the protection group at 2 a.m. every day.

The fourth command uses the **Set-DPMProtectionGroup** cmdlet to save the action that you performed on the protection group.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Mpg = Get-DPMModifiableProtectionGroup -ProtectionGroup $Pg
PS C:\> Set-DPMProtectionJobStartTime -ProtectionGroup -JobType ConsistencyCheck -StartTime
02:00
PS C:\> Set-DPMProtectionGroup $Mpg
```

Related topics

Get-ProtectionGroup

Get-ModifiableProtectionGroup

Set-ProtectionGroup

Set-DPMProtectionType

Set-DPMProtectionType

Sets the protection type for a protection group.

Syntax

```
Parameter Set: Default
Set-DPMProtectionType [-ProtectionGroup] <ProtectionGroup> [-LongTerm <LongTermProtection> ]
[-PassThru] [-ShortTerm <DataLocation> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMProtectionType** cmdlet sets the protection type for a protection group.

You can use the *ShortTerm* and *LongTerm* parameters individually or in combination to define protection types.

You can set the following combinations for a protection type:

-- ShortTerm Disk: Disk to Disk

-- ShortTerm Tape: Disk to Tape

-- LongTerm Tape: Disk to Tape (long-term)

-- ShortTerm Disk -LongTerm: Disk to Disk to Tape

-- ShortTerm Tape -LongTerm: Disk to Tape to Tape

Parameters

-LongTerm<LongTermProtection>

Specifies that the protection group is set to long-term tape protection. The acceptable values for this parameter are:

-- Tape

-- Online

-- OnlineAndTape

Aliases	none
Required?	false

Position?	named
Default Value	Tape
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-ShortTerm<DataLocation>

Specifies that the protection group is on disk, on tape, or on neither, if you do not specify a value. Valid values are: disk, tape.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Set-DPMProtectionType -detailed`".

Examples

Example 1: Set the protection type to short term on disk

This example sets the protection type for the protection group to short-term on disk.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command sets the protection type for the protection group stored in the $Pg variable to short-term on disk.

```
PS C:\> $Pg = New-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Pg Set-DPMProtectionType -ProtectionGroup $Pg -ShortTerm Disk
```

Example 2: Set the protection type to disk to disk to tape

This example sets the protection type for the protection group to short-term disk to disk to tape.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command sets the protection type for the protection group stored in the $Pg variable to disk to disk to tape.

```
PS C:\> $Pg = New- DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> Set-DPMProtectionType -ProtectionGroup $Pg -ShortTerm Disk –LongTerm Tape
```

Related topics

New-ProtectionGroup

Set-DPMProtectionGroup

Set-DPMReplicaCreationMethod

Set-DPMReplicaCreationMethod

Sets the replica creation method for disk-based protection.

Syntax

```
Parameter Set: Now
Set-DPMReplicaCreationMethod [-ProtectionGroup] <ProtectionGroup> -Now [-PassThru] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Later
Set-DPMReplicaCreationMethod [-ProtectionGroup] <ProtectionGroup> -Later <DateTime> [-
PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Manual
Set-DPMReplicaCreationMethod [-ProtectionGroup] <ProtectionGroup> -Manual [-PassThru] [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMReplicaCreationMethod** cmdlet sets the replica creation method for disk-based protection. You can set Now, Later, and Manual as the replica creation method. If you do not specify a value, the default replica creation method is Now.

If you specify Now for disk, System Center 2012 – Data Protection Manager (DPM) starts replication as soon as you create the protection group.

The **Set-DPMReplicaCreationMethod** cmdlet is the second step in changing the replication method that you specified for a protection group. You must first get the replica creation method by using the **Get-DPMReplicaCreationMethod** cmdlet.

You can use this cmdlet to set the replica creation method only if you set the protection type to short-term for disk and long-term for online. You can use the **Set-DPMProtectionType** to set the protection type for a protection group. If you specify any other type of protection type, this cmdlet results in an error. This cmdlet does not apply to tape-based protection.

Parameters

-Later<DateTime>

Specifies the time at which DPM performs the operation.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Manual

Indicates that you will apply the settings manually.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Now

Indicates that DPM creates the replica immediately.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet generates output. By default, this cmdlet does not generate output. You can use the *PassThru* parameter in order to use the cmdlet in a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Set-DPMReplicaCreationMethod -detailed`".

Examples

Example 1: Set the replica creation method

This example sets the replica creation method for a protection group.

The first command gets the protection group on the DPM server named DPMServer02 and stores the results in the $Pg variable.

The second command makes the protection group stored in the $Pg variable modifiable, and stores the result in the $Mpg variable.

The third command sets the replica creation method to Later for the protection group stored in the $Mpg variable to create a replica on April 23, 2013.

The fourth command uses the **Set-DPMProtectionGroup** cmdlet to save the action that you performed on the protection group stored in the $Mpg variable.

```
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Mpg = Get-DPMModifiableProtectionGroup -ProtectionGroup $Pg
PS C:\> Set-DPMReplicaCreationMethod -ProtectionGroup $Mpg -Later "23 April 2013"
PS C:\> Set-DPMProtectionGroup $Mpg
```

Related topics

Get-DPMProtectionGroup

Get-DPMReplicaCreationMethod

Set-DPMProtectionType

Set-DPMRole

Set-DPMRole

Saves changes to a DPM role.

Syntax

```
Parameter Set: Default
Set-DPMRole [-DpmRole] <DpmRole> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMRole** cmdlet saves changes you make to a System Center 2012 – Data Protection Manager (DPM) role. DPM roles allow Microsoft SQL Server database owners to recover databases without assistance from a DPM administrator.

You can change the name or description of a DPM by using the **Rename-DPMRole** cmdlet. Use the **Add-DPMSecurityGroup** cmdlet to add the role to appropriate security groups. Use the **Add-DPMRecoveryItem** cmdlet to specify instances of SQL Server and SQL Server databases that DPM protects. Use the **Add-DPMRecoveryTarget** cmdlet to specify target computers running SQL Server.

Parameters

-DpmRole<DpmRole>

Specifies a DPM role object. To obtain a DPM role object, use the **Get-DPMRole** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Set-DPMRole -detailed`".

Examples

Example 1: Save a change to a role

This example changes the name of a DPM role and saves that change.

The first command gets the role by using the **Get-DpmRole** cmdlet and stores it in the $DpmRole variable. The command makes the role editable.

The second command renames the role stored in the $DpmRole variable by using the **Rename-DPMRole** cmdlet.

The third command saves the change made in the second command for the role stored in the $DpmRole variable.

```
PS C:\> $DpmRole = Get-DpmRole –Name "OpsMgrSQL" -Editable
PS C:\> Rename-DPMRole -DpmRole $DpmRole -Name "OpsMgrSQL23"
PS C:\> Set-DPMRole -DPMRole $DpmRole
```

Related topics

Get-DPMRole

New-DPMRole

Remove-DPMRole

Rename-DPMRole

Add-DPMSecurityGroup

Add-DPMRecoveryItem

Add-DPMRecoveryTarget

Set-DPMTape

Set-DPMTape

Marks a tape as Archive, Cleaner, Free or Not Free.

Syntax

```
Parameter Set: Archive
Set-DPMTape [-Tape] <Media[]> -Archive [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: Cleaner
Set-DPMTape [-Tape] <Media[]> -Cleaner [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: Free
Set-DPMTape [-Tape] <Media[]> -Free [-PassThru] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: Unfree
Set-DPMTape [-Tape] <Media[]> -NotFree [-PassThru] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Set-DPMTape** cmdlet marks a tape as Archive, Cleaner, Free or Not Free.

Parameters

-Archive

Indicates that the media is an archive tape.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-Cleaner

Indicates that the media is a cleaning tape.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Free

Indicates that the media is free for use.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-NotFree

Indicates that the media is not free for use.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet is part of a pipeline. Use this parameter with System Center 2012 – Data Protection Manager (DPM) commands to return a related object in cases where there is no default output.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media[]>

Specifies an array of tape objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Tape**

Notes

- For more information, type "`Get-Help Set-DPMTape -detailed`".

Examples

Example 1: Mark a tape as free

This example marks a tape as Free.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library for the server, and stores the result in the $DPMLib variable.

The second command uses the **Get-Tape** cmdlet to retrieve the tape object, and stores the result in the $Tape variable.

The third command uses the **Set-DPMTape** cmdlet to set the second tape as Free.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $Tape = Get-Tape -DPMLibrary $DPMLib
PS C:\> Set-DPMTape -Tape $Tape[1] -Free
```

Related topics

Add-DPMTape

Get-DPMLibrary

Get-DPMTape

Remove-DPMTape

Set-DPMTapeBackupOption

Set-DPMTapeBackupOption

Modifies the tape backup and library options for a DPM protection group.

Syntax

```
Parameter Set: LibraryOptions
Set-DPMTapeBackupOption [-ProtectionGroup] <ProtectionGroup> -BackupLibrary <Library> -
DrivesAllocated <Int32> [-PassThru] [-PGSet <PGSet> ] [-RemovePGSet] [-TapeCopyLibrary
<Library> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: LongTerm
Set-DPMTapeBackupOption [-ProtectionGroup] <ProtectionGroup> -LongTerm [-CompressData] [-
EncryptData] [-PassThru] [-PerformIntegritycheck] [-PGSet <PGSet> ] [-RemovePGSet] [-
Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ShortTerm
Set-DPMTapeBackupOption [-ProtectionGroup] <ProtectionGroup> -ShortTerm [-CompressData] [-
EncryptData] [-PassThru] [-PerformIntegritycheck] [-PGSet <PGSet> ] [-RemovePGSet] [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-DPMTapeBackupOption** cmdlet modifies the tape backup options, such as library and drive information, and short-term and long-term tape backup options for a System Center 2012 – Data Protection Manager (DPM) protection group.

Parameters

-BackupLibrary<Library>

Specifies a backup library.

Aliases	none
Required?	true
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-CompressData

Indicates that the backup process compresses the data to reduce storage needs. Do not use compression and encryption together.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DrivesAllocated<Int32>

Specifies the number of drives to allocate to the protection group.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-EncryptData

Indicates that the backup process encrypts the data during backup. Do not use compression and encryption together.

Aliases	none
Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-LongTerm

Indicates that the protection group uses long-term tape protection. Do not use this parameter with the **ShortTerm** parameter. You must set *LongTerm* and *ShortTerm* options in separate commands.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PassThru

Indicates that the cmdlet is part of a pipeline.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PerformIntegritycheck

Indicates that the backup process performs an integrity check on the tape backup.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PGSet<PGSet>

Specifies the name of a protection group set.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies the name of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-RemovePGSet

Indicates that the cmdlet removes the protection group set as an option.

Aliases	none

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ShortTerm

Indicates that the protection group will be on disk, on tape, or on neither, if nothing is specified. Do not use this parameter with the *LongTerm* parameter. You must set LongTerm and ShortTerm options in separate commands.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TapeCopyLibrary<Library>

Specifies a library object. Use the secondary tape library to make copies of the tape in the backup library.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **ProtectionGroup**

Notes

- For more information, type "`Get-Help Set-DPMTapeBackupOption -detailed`".

Examples

Example 1: Set tape backup options for short-term backup

This example sets the tape backup options to short-term backup with encryption.

The first command uses the **Get-ProtectionGroup** cmdlet and stores the result in the variable named $PGroup.

The second command uses the **Set-DPMTapeBackupOption** cmdlet to set the *ShortTerm* and *Encryption* parameters.

```
PS C:\> $PGroup = Get-ProtectionGroup -DPMServerName "Contoso-DPMServer"
PS C:\> Set-DPMTapeBackupOption -ProtectionGroup $PGroup -ShortTerm -EncryptData
```

Example 2: Set tape backup options for a protection group

This example sets the tape backup option for the protection group in the variable named $MPGroup.

The first command uses the **Get-ProtectionGroup** cmdlet and stores the result in the variable named $PGroup.

The second command uses the **Get-ModifiableProtectionGroup** cmdlet to retrieve the settings for the first protection group, and stores the result in the variable named $MPGroup.

The third command uses the **Get-DPMLibrary** cmdlet and stores the result in the variable named $Lib.

The fourth command uses the **Set-DPMTapeBackupOption** cmdlet to set the *BackupLibrary* and *DrivesAllocated* parameters.

The fifth command uses the **Set-ProtectionGroup** cmdlet to set the protection group to the value in the $MPGroup variable.

```
PS C:\> $PGroup = Get-ProtectionGroup "Contoso-DPMServer"
PS C:\> $MPGroup = Get-ModifiableProtectionGroup -ProtectionGroup $PGroup[0]
PS C:\> $Lib = Get-DPMLibrary "Contoso-DPMServer"
PS C:\> Set-DPMTapeBackupOption -ProtectionGroup $MPGroup -BackupLibrary $Lib -
DrivesAllocated 1
PS C:\> Set-ProtectionGroup $MPGroup
```

Related topics

Get-DPMTapeBackupOption

Start-DPMAutoProtection

Start-DPMAutoProtection

Adds new SQL Server databases to a DPM protection group.

Syntax

```
Parameter Set: Default
Start-DPMAutoProtection [[-DPMServerName] <String> ] [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Start-DPMAutoProtection** cmdlet adds new Microsoft SQL Server databases to a System Center 2012 – Data Protection Manager (DPM) protection group. The cmdlet searches for new SQL Server databases that belong to SQL Server instances configured for auto-protection, and adds those databases to the protection group for a specified DPM server.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	ComputerName,CN
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Start-DPMAutoProtection -detailed`".

Related topics

Get-DPMAutoProtectIntent

Set-DPMAutoProtectIntent

Start-DPMCloudRegistration

Start-DPMCloudRegistration

Registers a DPM server with Windows Azure Online Backup service to enable online protection features.

Syntax

```
Parameter Set: Default
Start-DPMCloudRegistration [[-DPMServerName] <String> ] [-Credential] <PSCredential> [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Start-DPMCloudRegistration cmdlet registers a Data Protection Manager (DPM) server with Windows Azure Online Backup service to enable online protection features.

Parameters

-Credential<PSCredential>

Specifies the user name and password for the administrator account of the Windows Azure Online Backup subscription.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of the DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Register a DPM server with Windows Azure Online Backup

This example registers a DPM server with Windows Azure Online Backup.

The first command converts the string myPassword to a secure string and assigns the secure string to the variable named $Secpasswd.

The second command creates a new **PSCredential** object from the username sjones@contoso.com and the secure password in $Secpasswd, then assigns the object to the variable named $Mycreds.

The third command registers the DPM server TestingServer with Windows Azure Online Backup service by using the credential in $Mycreds.

```
PS C:\> $Secpasswd = ConvertTo-SecureString "myPassword" -AsPlainText -Force
PS C:\> $Mycreds = New-Object System.Management.Automation.PSCredential
("sjones@contoso.com", $Secpasswd)
PS C:\> Start-DPMCloudRegistration -DPMServerName "TestingServer" -Credential $Mycreds
```

Related topics

Start-DPMCloudUnregistration

Start-DPMCloudUnregistration

Start-DPMCloudUnregistration

Unregisters a DPM server from Windows Azure Online Backup service.

Syntax

```
Parameter Set: Default
Start-DPMCloudUnregistration [[-DPMServerName] <String> ] [-Credential] <PSCredential> [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMCloudUnregistration** cmdlet unregisters a System Center 2012 – Data Protection Manager (DPM) server from the Windows Azure Online Backup service. The cmdlet deletes all online data and frees up storage space that the DPM server used.

Parameters

-Credential<PSCredential>

Specifies the user name and password for the administrator account of the Windows Azure Online Backup subscription.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-DPMServerName<String>

Specifies the name of the DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Unregister a DPM server from Windows Azure Online Backup

This example unregisters a DPM server from Windows Azure Online Backup.

The first command converts the string myPassword to a secure string and assigns the secure string to the variable named $Secpasswd.

The second command creates a new **PSCredential** object from the username sjones@contoso.com and the secure password in $Secpasswd, then assigns the object to the variable named $Mycreds.

The third command unregisters the DPM server named TestingServer from the Windows Azure Online Backup service by using the credential in $Mycreds.

```
PS C:\> $Secpasswd = ConvertTo-SecureString "myPassword" -AsPlainText -Force
PS C:\> $Mycreds = New-Object System.Management.Automation.PSCredential
("sjones@contoso.com", $Secpasswd)
PS C:\> Start-DPMCloudUnregistration -DPMServerName "TestingServer" -Credential $Mycreds
```

Start-DPMCreateCatalog

Start-DPMCreateCatalog

Creates a catalog for a data source.

Syntax

```
Parameter Set: Datasource
Start-DPMCreateCatalog [-Datasource] <Datasource> [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMCreateCatalog** cmdlet creates a catalog for a data source.

In System Center 2012 – Data Protection Manager (DPM) you can generate catalogs only for SharePoint farms. A catalog is a list of all URLs in the farm. You must use a catalog for item-level recovery.

Parameters

-Datasource<Datasource>

Specifies a data source object. A data source can be a file system share or volume for Windows, Microsoft SQL Server database, Microsoft Exchange storage group, Microsoft SharePoint farm, Microsoft Virtual Machine, DPM database, or system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation. Use this parameter along with the **Async** parameter to build a GUI by using cmdlets. Do not use this parameter if you are working with the DPM Management Shell.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Start-DPMCreateCatalog -detailed`".

Examples

- `PS C:\>`

Start-DPMDatasourceConsistencyCheck

Start-DPMDatasourceConsistencyCheck

Performs a consistency check on a DPM data source.

Syntax

```
Parameter Set: Datasource
Start-DPMDatasourceConsistencyCheck [-Datasource] <Datasource> [[-AdhocJobsContext]
<AdhocJobsContext> ] [-ForcedFullCC] [-HeavyWeight] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ProtectionGroup
Start-DPMDatasourceConsistencyCheck [-ProtectionGroup] <ProtectionGroup> [[-
AdhocJobsContext] <AdhocJobsContext> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMDatasourceConsistencyCheck** cmdlet performs a consistency check on a Data Protection Manager (DPM) data source. You can run consistency checks on a specific data source or on all data sources in a protection group that are in an inconsistent state.

Parameters

-AdhocJobsContext<AdhocJobsContext>

Specifies the context details of the ad hoc job. Do not use this parameter from the Windows PowerShell command line.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource>

Specifies a data source object. A data source may be a file system share or volume for Windows, Microsoft SQL Server database, Microsoft Exchange storage group, Microsoft SharePoint farm, Microsoft virtual machine, DPM database, or system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-ForcedFullCC

Indicates that the cmdlet performs a heavyweight consistency check on all databases in the farm, not just on the databases in an inconsistent state. This option is specific to SharePoint.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-HeavyWeight

Indicates that the cmdlet performs a heavyweight consistency check. A heavyweight consistency check checksums the contents of each file. This parameter affects only file servers; the cmdlet always performs heavyweight consistency checks on application servers.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the to the user. Use the *JobStateChangedEventHandler* parameter along with the *Async* parameterto build a graphical user interface (GUI) by using cmdlets. Do not use it when you work with the DPM Management Shell.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group object.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Start a consistency check on a data source

This example starts a consistency check on a data source.

The first command gets the protection groups from the server named TestingServer and stores them in the variable named $Pg.

The second command gets the data source for the first protection group (stored in position 0) in $Pg and stores the data source in the variable named $Ds.

The third command runs a consistency check on the data source that is stored in $Ds.

```
PS C:\> $Pg = Get-ProtectionGroup –DPMServerName TestingServer
PS C:\> $Ds = Get-Datasource $Pg[0]
PS C:\> Start-DPMDatasourceConsistencyCheck -Datasource $Ds
```

Related topics

Get-Datasource

Get-ProtectionGroup

Start-DPMDiskRescan

Start-DPMDiskRescan

Scans for new disks or disk configuration changes on a DPM server.

Syntax

```
Parameter Set: Default
Start-DPMDiskRescan [[-DPMServerName] <String> ] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMDiskRescan** cmdlet scans for new disks or disk configuration changes on a System Center 2012 – Data Protection Manager (DPM) server.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMDiskRescan -detailed`"

Examples

Example 1: Scan a disk for configuration changes

This command scans a DPM server for new disks or changes to existing disk configurations.

```
PS C:\> Start-DPMDiskRescan -DPMServerName "Contoso-DPMServer"
```

Start-DPMLibraryInventory

Start-DPMLibraryInventory

Starts an inventory of tapes in a DPM library.

Syntax

```
Parameter Set: FastInventory
Start-DPMLibraryInventory [-DPMLibrary] <Library> [-FastInventory] [-
JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: DetailedInventory
Start-DPMLibraryInventory [-DPMLibrary] <Library> -DetailedInventory [-
JobStateChangedEventHandler <JobStateChangedEventHandler> ] [-Tape <Media[]> ] [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMLibraryInventory** cmdlet starts an inventory of the tapes in a System Center 2012 – Data Protection Manager (DPM) library. DPM detects any tape, with or without a bar code, in a library. You can choose either a detailed inventory or a fast inventory.

For a detailed inventory, DPM reads the header area of the tapes in a library to identify the on-media identifier (OMID) on each tape.

For a fast inventory of a library that has a bar code reader, DPM reads the bar code of each tape in the library. If a library does not have a bar code reader or a tape does not have a bar code, DPM reads the header area of the tapes to identify the OMID.

Parameters

-DetailedInventory

Indicates that DPM performs a detailed inventory on the specified library.

Aliases	none
Required?	true
Position?	named

Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-FastInventory

Indicates that DPM performs a fast inventory on the specified library.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for **Job.StateChanged** events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false

Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media[]>

Specifies an array of tape objects. To obtain a tape object, use the **Get-DPMTape** cmdlet.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMLibraryInventory -detailed`".

Examples

Example 1: Perform a fast inventory

This example performs a fast inventory on a tape library for the DPM server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the library associated with the server, and stores that object in the $DPMLib variable.

The second command starts an inventory on the library object stored in the $DPMLib variable. The command uses the *FastInventory* parameter, so the command starts a fast inventory.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Start-DPMLibraryInventory -DPMLibrary $DPMLib -FastInventory
```

Example 2: Perform a detailed inventory

This example performs a detailed inventory on a tape library for the DPM server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the library associated with the server, and stores that object in the $DPMLib variable.

The second command starts an inventory on the library object stored in the $DPMLib variable. The command uses the *DetailedInventory* parameter, so the command starts a detailed inventory.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Start-DPMLibraryInventory -DPMLibrary $DPMLib -DetailedInventory
```

Related topics

Get-DPMLibrary

Get-DPMTape

Start-DPMLibraryRescan

Start-DPMLibraryRescan

Starts a scan to identify new libraries and update existing libraries.

Syntax

```
Parameter Set: Full
Start-DPMLibraryRescan [[-DPMServerName] <String> ] [-Full] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [ <CommonParameters>]

Parameter Set: Quick
Start-DPMLibraryRescan [[-DPMServerName] <String> ] -Quick [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [ <CommonParameters>]

Parameter Set: RefreshOnly
Start-DPMLibraryRescan [[-DPMServerName] <String> ] -RefreshOnly [-
JobStateChangedEventHandler <JobStateChangedEventHandler> ] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMLibraryRescan** cmdlet starts a scan for a System Center 2012 – Data Protection Manager (DPM) server to identify new libraries and stand-alone tape drives, and to update information for existing libraries. You can select one of the following parameters:

-- Full. The cmdlet checks for new libraries and stand-alone tape drives and refreshes all libraries and drives.

-- Quick. The cmdlet checks for new libraries and stand-alone tape drives.

-- RefreshOnly. The cmdlet refreshes known drives.

If you do not specify any of these parameters, the cmdlet performs a full scan.

Parameters

-DPMServerName<String>

Specifies the name of a DPM server.

Aliases	none
Required?	false

Position?	1
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Full

Indicates that the cmdlet starts a full library rescan.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for changes in the status of a job. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Quick

Indicates that the cmdlet starts a quick library rescan.

Aliases	none

Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RefreshOnly

Indicates that the cmdlet starts a refresh library scan.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMLibraryRescan -detailed`".

Examples

Example 1: Perform a full rescan

This command starts a full rescan of the libraries on the DPMserver named DPMServer07.

```
PS C:\> Start-DPMLibraryRescan -DPMServerName "DPMServer07" -Full
```

Start-DPMOnline

Start-DPMOnline

Syntax

```
Parameter Set:
Start-DPMOnline [- ] < > [- ] < > [- ] < > [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

This cmdlet has been deprecated.

Parameters

-<>

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-<>

Aliases	none
Required?	true
Position?	1
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-<>

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

```
PS C:\>
```

Start-DPMOnlineRecatalog

Start-DPMOnlineRecatalog

Returns a detailed list of data for a DPM recovery point.

Syntax

```
Parameter Set: Default
Start-DPMOnlineRecatalog [-RecoveryPoint] <RecoverySource> [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-RecoveryPointLocation <RecoverySourceLocation> ] [-
Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMOnlineRecatalog** cmdlet returns a detailed list of data for a System Center 2012 – Data Protection Manager (DPM) recovery point. You can use this information to perform partial recoveries or recovery of selected files.

Parameters

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for changes in the status of a job. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryPoint<RecoverySource>

Specifies a recovery point object. To obtain a recovery point object, use the **Get-DPMRecoveryPoint** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoveryPointLocation<RecoverySourceLocation>

Specifies a recovery point location object. To obtain a recovery point location object, use the **Get-DPMRecoveryPointLocation** cmdlet. You need to specify the location of recovery point if a recovery item exists in more than one recovery point.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMOnlineRecatalog -detailed`".

Examples

Example 1: Start online catalog

This example performs on online catalog action of a recovery point on a DPM server. The first command uses the **Get-ProtectionGroup** to get a protection group for the DPM server named DPMServer07, and stores the protection group object in the $DPMPG variable.

The second command uses the **Get-DPMDatasource** cmdlet to get a data source for the protection group object specified in the $DPMPG variable, and stores the data source object in the $DPMDS variable.

The third command uses the **Get-DPMRecoveryPoint** cmdlet to get recovery points for the data source object stored in the $DPMDS variable, and stores the recovery point objects in the $DPMRP variable.

The fourth command uses the **Get-DPMRecoveryPointLocation** cmdlet to get a recovery point location for the recovery point objects stored in the $DPMRP variable, and stores the recovery point location object in the $DPMRPL variable.

The fifth command starts a catalog action for a recovery point stored in the $DPMRP variable. The command uses standard array syntax to select the second member of that array. The command specifies the recovery point location as the object stored in the $DPMRPL variable.

```
PS C:\> $DPMPG = Get-DPMProtectionGroup -DPMServerName "DPMServer07"
PS C:\> $DPMDS = Get-DPMDatasource -ProtectionGroup $DPMPG
PS C:\> $DPMRP = Get-DPMRecoveryPoint -Datasource $DPMDS
PS C:\> $DPMRPL = Get-DPMRecoveryPointLocation -RecoveryPoint $DPMRP
PS C:\> Start-DPMOnlineRecatalog -RecoveryPoint $DPMRP[1] -RecoveryPointLocation $DPMRPL
```

Related topics

Get-DPMRecoveryPoint

Get-DPMRecoveryPointLocation

Get-DPMProtectionGroup

Get-DPMDatasource

Start-DPMProductionServerSwitchProtection

Start-DPMProductionServerSwitchProtection

Syntax

```
Parameter Set:
Start-DPMProductionServerSwitchProtection [- ] < > [- ] < > [- ] < > [- ] < > [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

This cmdlet has been deprecated.

Parameters

-<>

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-<>

Aliases	none
Required?	true
Position?	4
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-<>

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue, ByPropertyName)
Accept Wildcard Characters?	false

-<>

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

```
PS C:\>
```

Start-DPMSwitchProtection

Start-DPMSwitchProtection

Switches protection of data sources to a secondary DPM server.

Syntax

```
Parameter Set: Default
Start-DPMSwitchProtection [-ProtectionGroup] <ProtectionGroup> -Datasource <Datasource[]> [-
Async] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMSwitchProtection** cmdlet switches protection of a set of data sources to a secondary System Center 2012 – Data Protection Manager (DPM) server. Run this cmdlet on the secondary DPM server.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Datasource<Datasource[]>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.

-- Microsoft Exchange storage group.

-- Microsoft SharePoint farm.

-- Microsoft Virtual Machine.

-- DPM database.

-- A system state that is a member of a protection group.

Aliases	none
Required?	true
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a protection group object. To obtain a protection group object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Start-DPMSwitchProtection -detailed`".

Examples

Example 1: Switch protection for a data source to the secondary server

This example switches protection for a data source to a secondary DPM server. The first command gets the data protection group for a specified server by using the **Get-ProtectionGroup** cmdlet, and stores it in the $PG variable.

The second command uses the **Get-Datasource** cmdlet to obtain a data source. The command uses standard array syntax to specify the first member of the $PG array variable.

The third command switches protection to the secondary server for the data source stored in the $DS variable that is part of the protection group stored in the $PG variable. Run this example on the secondary server.

```
PS C:\> $PG = Get-ProtectionGroup -DPMServerName DPMServer073
PS C:\>$DS = Get-Datasource $PG[0]
PS C:\>Start-DPMSwitchProtection -ProtectionGroup $PG -Datasource $DS
```

Related topics

Get-DPMProtectionGroup

Get-DPMDatasource

Start-DPMTapeDriveCleaning

Start-DPMTapeDriveCleaning

Starts a job to clean a tape drive.

Syntax

```
Parameter Set: Default
Start-DPMTapeDriveCleaning [-TapeDrive] <Drive[]> [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMTapeDriveCleaning** cmdlet starts a job to clean a tape drive.

Parameters

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation to the user.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-TapeDrive<Drive[]>

Specifies an array of tape drives.

Aliases	none

Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMTapeDriveCleaning -detailed`".

Examples

Example 1: Clean a tape drive

This example performs a tape drive cleaning operation.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library, and stores the result in the variable named $DPMLib.

The second command uses the **Get-TapeDrive** cmdlet to retrieve the list of the tape drives in the library, and stores the result in the variable named $TDrive.

The third command uses the **Start-DPMTapeDriveCleaning** cmdlet to clean the tape drive.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $TDrive = Get-TapeDrive -DPMLibrary $DPMLib
PS C:\> Start-DPMTapeDriveCleaning -TapeDrive $TDrive
```

Related topics

Get-DPMLibrary

Start-DPMTapeErase

Start-DPMTapeErase

Erases the contents of a tape.

Syntax

```
Parameter Set: Default
Start-DPMTapeErase [-Tape] <Media[]> [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMTapeErase** cmdlet erases the contents of a tape in System Center 2012 – Data Protection Manager (DPM).

You can reuse an expired tape in DPM without first erasing the data on the tape.

Parameters

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation to the user.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media[]>

Specifies an array of tape objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMTapeErase -detailed`".

Examples

Example 1: Erase the data on a tape

This example erases the data on a tape.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the library, and stores the result in the variable named $DPMLib.

The second command uses the **Get-Tape** cmdlet to retrieve the list of the tapes in the library, and stores the result in the variable named $Tape.

The third command uses the **Start-DPMTapeErase** cmdlet to erase the data on the third tape object stored in $Tape.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $Tape = Get-Tape -DPMLibrary $DPMLib
PS C:\> Start-DPMTapeErase -Tape $Tape[2]
```

Related topics

Get-DPMLibrary

Start-DPMTapeRecatalog

Start-DPMTapeRecatalog

Recatalogs the data on a tape.

Syntax

```
Parameter Set: Default
Start-DPMTapeRecatalog [-Tape] <Media[]> [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Start-DPMTapeRecatalog** cmdlet recatalogs the data on a tape. System Center 2012 – Data Protection Manager (DPM) performs this operation before starting data recovery data from an imported tape.

A recatalog operation determines the recovery points for data on the tape.

Parameters

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Tape<Media[]>

Specifies an array of tape objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **Job**

Notes

- For more information, type "`Get-Help Start-DPMTapeRecatalog -detailed`".

Examples

Example 1: Recatalog data on a tape

This example recatalogs the data on an imported tape.

The first command uses the **Get-DPMLibrary** cmdlet to retrieve the name of the library on the server, and stores the result in the $DPMLib variable.

The second command uses the **Get-Tape** cmdlet to retrieve a list of the tapes in the library, and stores the result in the $Tape variable.

The third command uses the **Start-DPMTapeRecatalog** cmdlet to recatalog the data.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "Contoso-DPMServer"
PS C:\> $Tape = Get-Tape -DPMLibrary $DPMLib
PS C:\> Start-DPMTapeRecatalog -Tape $Tape[2]
```

Related topics

Get-DPMLibrary

Stop-DPMJob

Stop-DPMJob

Stops a running DPM job.

Syntax

```
Parameter Set: CancelDpmJob
Stop-DPMJob [-Job] <Job[]> [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: CancelDpmTask
Stop-DPMJob [-Task] <Task[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Stop-DPMJob** cmdlet stops a running job in System Center 2012 – Data Protection Manager (DPM).

Parameters

-Job<Job[]>

Specifies an array of job objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Task<Task[]>

Specifies an array of task objects.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

Example 1: Stop all DPM jobs

This example stops all jobs for a protection group.

The first command uses the **Get-DPMProtectionGroup** cmdlet to retrieve the protection group, and then stores the result in the variable named $PGroup.

The second command uses the **Get-DPMJob** cmdlet to retrieve the list of running jobs for the protection group, and then stores the result in the variable named $Job.

The third command uses the **Stop-DPMJob** cmdlet to stop the currently running jobs stored in the variable $Job.

```
PS C:\> $PGroup = Get-DPMProtectionGroup -DPMServerName "Contoso-DPMServer"
PS C:\> $Job = Get-DPMJob -ProtectionGroup $PGroup -Status InProgress
PS C:\> Stop-DPMJob $Job
```

Related topics

Get-DPMJob

Restart-DPMJob

Stop-DPMOnline

Stop-DPMOnline

Syntax

```
Parameter Set:
Stop-DPMOnline [- ] < > [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

This cmdlet has been deprecated.

Parameters

-<>

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

```
PS C:\>
```

```
PS C:\>
```

Test-DPMTapeData

Test-DPMTapeData

Verifies a data set for a recovery point.

Syntax

```
Parameter Set: Default
Test-DPMTapeData [-RecoveryPoint] <RecoverySource> [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-RecoveryPointLocation <RecoverySourceLocation> ] [
<CommonParameters>]
```

Detailed Description

The **Test-DPMTapeData** cmdlet verifies a data set for a recovery point in System Center 2012 – Data Protection Manager (DPM). A recovery point is a point-in-time copy of a replica stored on a DPM server.

Parameters

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Indicates that the cmdlet displays the status of the operation to the user.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-RecoveryPoint<RecoverySource>

Specifies a recovery point.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-RecoveryPointLocation<RecoverySourceLocation>

Specifies the location of a recovery point. A recovery item may exist in more than one location for the same point in time, such as on a disk and tape, or on two separate tapes.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Outputs

The output type is the type of the objects that the cmdlet emits.

- **RecoveryPoint**

Notes

- For more information, type "`Get-Help Test-DPMTapeData -detailed`".

Examples

Example 1: Verify a recovery point

This example verifies a recovery point at a specified location.

The first command uses the **Get-ProtectionGroup** cmdlet to retrieve the protection group on the DPM server, and then stores the result in the variable named $PGroup.

The second command uses the **Get-Datasource** cmdlet to retrieve the data source for the protection group, and then stores the result in the variable named $DSource.

The third command uses the **Get-RecoveryPoint** cmdlet to retrieve the recovery point for the data source, and then stores the result in the variable named $RPoint.

The fourth command uses the **Get-RecoveryPointLocation** cmdlet to retrieve the recovery point location, and then stores the result in the variable named $RS1.

The fifth command uses the **Test-DPMTapeData** cmdlet to verify a data set for the second recovery point.

```
PS C:\> $PGroup = Get-ProtectionGroup -DPMServerName "Contoso-DPMServer"
PS C:\> $DSource = Get-Datasource -ProtectionGroup $PGroup
PS C:\> $RPoint = Get-RecoveryPoint -Datasource $DSource
PS C:\> $RS1 = Get-RecoveryPointLocation -RecoveryPoint $RPoint[1]
PS C:\> Test-DPMTapeData -RecoveryPoint $RPoint[1] -RecoveryPointLocation $RS1
```

Related topics

Copy-DPMTapeData

Unlock-DPMLibraryDoor

Unlock-DPMLibraryDoor

Unlocks the door of a DPM library.

Syntax

```
Parameter Set: Default
Unlock-DPMLibraryDoor [-DPMLibrary] <Library> [[-Timeout] <Int32> ] [-Async] [-
DoorAccessJobStateChangeEventHandler <DoorAccessJobStateChangeEventHandler> ] [-Confirm] [-
WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Unlock-DPMLibraryDoor** cmdlet unlocks the door of a System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DoorAccessJobStateChangeEventHandler<DoorAccessJobStateChangeEventHandler>

Specifies an event handler for certain door access events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Timeout<Int32>

Specifies a time-out period, in seconds, for the unlock action. If the operation times out, the door locks automatically.

Aliases	none
Required?	false
Position?	2
Default Value	none

Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Unlock-DPMLibraryDoor -detailed`".

Examples

Example 1: Unlock a library door

This example unlocks a library door for a library associated with the server named DPMServer07. The first command gets a library associated with the server, and stores the library object in the $DPMLib variable.

The second command unlocks the library door for the library object stored in the $DPMLib variable.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Unlock-DPMLibraryDoor -DPMLibrary $DPMLib
```

Related topics

Get-DPMLibrary

Lock-DPMLibraryDoor

Unlock-DPMLibraryIEPort

Unlock-DPMLibraryIEPort

Unlocks the IE port for a DPM library.

Syntax

```
Parameter Set: Default
Unlock-DPMLibraryIEPort [-DPMLibrary] <Library> [-Async] [-JobStateChangedEventHandler
<JobStateChangedEventHandler> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Unlock-DPMLibraryIEPort** cmdlet unlocks the insert/eject (IE) port for a System Center 2012 – Data Protection Manager (DPM) library.

Parameters

-Async

Indicates that the command runs asynchronously. When you run a command asynchronously, the command prompt returns immediately even if the job takes an extended time to finish.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-DPMLibrary<Library>

Specifies a DPM library object. To obtain a DPM library object, use the **Get-DPMLibrary** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-JobStateChangedEventHandler<JobStateChangedEventHandler>

Specifies an event handler for certain IE port events. You can use this parameter to update a GUI that runs in Windows PowerShell, but do not use it in the Windows PowerShell console.

Aliases	none
Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Unlock-DPMLibraryIEPort -detailed`".

Examples

Example 1: Unlock an IE port

This example unlocks an IE port for a library associated with a server named DPMServer07. The first command uses the **Get-DPMLibrary** cmdlet to get the library for the named server, and stores library object in the $DPMLib variable.

The second command unlocks the object stored in the $DPMLib variable.

```
PS C:\> $DPMLib = Get-DPMLibrary -DPMServerName "DPMServer07"
PS C:\> Unlock-DPMLibraryDoor -DPMLibrary $DPMLib
```

Related topics

Lock-DPMLibraryIEPort

Get-DPMLibrary

Update-DPMJob

Update-DPMJob

Syntax

```
Parameter Set:
Update-DPMJob [-Job] < > [ <CommonParameters>]
```

Detailed Description

This cmdlet has been deprecated.

Parameters

-Job<>

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Examples

```
PS C:\>
```

Update-DPMPGSet

Update-DPMPGSet

Updates and saves changes to a PG set.

Syntax

```
Parameter Set: AllParams
Update-DPMPGSet [-PGSet] <PGSet> [-Name] <String> [-WritePeriodUnit] <TimeUnit> [[-
AllowDifferentRetentionPeriods]] [-WritePeriodValue] <UInt32> [-ExpiryToleranceUnit]
<TimeUnit> [-ExpiryToleranceValue] <UInt32> [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: AddPG
Update-DPMPGSet [-PGSet] <PGSet> [-Add] <ProtectionGroup> [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: AllowDiffRetnPeriodsFlag
Update-DPMPGSet [-PGSet] <PGSet> [-AllowDifferentRetentionPeriods] [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: ChangeName
Update-DPMPGSet [-PGSet] <PGSet> [-Name] <String> [-Confirm] [-WhatIf] [ <CommonParameters>]

Parameter Set: ChangeTimePeriods
Update-DPMPGSet [-PGSet] <PGSet> [-WritePeriodUnit] <TimeUnit> [-WritePeriodValue] <UInt32>
[-ExpiryToleranceUnit] <TimeUnit> [-ExpiryToleranceValue] <UInt32> [-Confirm] [-WhatIf] [
<CommonParameters>]

Parameter Set: RemovePG
Update-DPMPGSet [-PGSet] <PGSet> [-Remove] <ProtectionGroup> [-Confirm] [-WhatIf] [
<CommonParameters>]
```

Detailed Description

The **Update-DPMPGSet** cmdlet updates and saves changes to a System Center 2012 – Data Protection Manager (DPM) protection group (PG) set. A DPM PG is a collection of protection groups that you collocate on the same tape.

Parameters

-Add<ProtectionGroup>

Specifies a protection group. The cmdlet adds this PG to the PG set.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-AllowDifferentRetentionPeriods

Indicates that protection groups with different retention periods can be part of the same PG set.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExpiryToleranceUnit<TimeUnit>

Specifies the measurement unit for expiry tolerance.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ExpiryToleranceValue<UInt32>

Specifies the maximum length of time for which an expired recovery point can remain on a tape until DPM marks the tape as expired.

Aliases	none
Required?	true
Position?	5
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Name<String>

Specifies a new name for the PG set.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-PGSet<PGSet>

Specifies a PG set to update.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	True (ByValue)

Accept Wildcard Characters?	false

-Remove<ProtectionGroup>

Specifies a protection group. The cmdlet removes this PG from the PG set.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WritePeriodUnit<TimeUnit>

Specifies the measurement unit for the write period.

Aliases	none
Required?	true
Position?	2
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WritePeriodValue<UInt32>

Specifies the length of time for which a tape is available for writing new backups. DPM marks the tape as Offsite Ready after this interval.

Aliases	none
Required?	true
Position?	3
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Update-DPMPGSet -detailed`".

Examples

Example 1: Update write period and expiry tolerance

This example updates the first PG set from PG sets on the on the server DPMServer07 with values for write period and expiry tolerance.

The first command uses the **Get-DPMPGSet** cmdlet to get the PG sets for the specified server and stores them in the $PgSet variable.

The second command specifies the first member of the $PgSet variable by using standard array notation. The command updates values for write period and expiry tolerance.

```
PS C:\> $PgSet = Get-DPMPGSet -DPMServerName "DPMServer07"
PS C:\> Update-DPMPGSet -PGSet $PgSet[0] -Name "PGSset3" -WritePeriodUnit day -
WritePeriodValue 60 -ExpiryToleranceUnit Day -ExpiryToleranceValue 10
```

Example 2: Add a PG to a PG set

This example adds a protection group from DPMServer07 to the first protection group set on that DPM server.

The first command uses the **Get-DPMPGSet** cmdlet to get the PG sets for the specified server and stores them in the $PgSet variable.

The second command uses the **Get-DPMProtectionGroup** cmdlet to get a PG from the specified DPM server that has a name that contains PG1 and stores it in the $Pg variable.

The third command specifies the first member of the $PgSet variable by using standard array notation. The command updates that set to contain the PG stored in the $Pg variable.

```
PS C:\> $PgSet = Get-DPMPGSet -DPMServerName "DPMServer07"
PS C:\> $Pg = Get-DPMProtectionGroup -DPMServerName "DPMServer07" | where {($_.friendlyname)
-match "PG1" }
PS C:\> Update-DPMPGSet -PGSet $PgSet[0] -Add $Pg
```

Example 3: Remove a PG from a PG set

This example removes the first protection group set from the list of protection group sets on the DPM server TestingServer.

The first command uses the **Get-DPMPGSet** cmdlet to get the PG sets for the specified server and stores them in the $PgSet variable.

The second command uses the **Get-DPMProtectionGroup** cmdlet to get a PG from the specified DPM server that has a name that contains PG1 and stores it in the $Pg variable.

The third command specifies the first member of the $PgSet variable by using standard array notation. The command updates that set to no longer contain the PG stored in the $Pg variable.

```
PS C:\> $PgSet = Get-DPMPGSet -DPMServerName "DPMServer07"
PS C:\> $Pg = Get-ProtectionGroup -DPMServerName "DPMServer07" | where { ($_.friendlyname) -
match "PG1" }
PS C:\> Update-DPMPGSet -PGSet $PgSet[0] -Remove $Pg
```

Related topics

Get-DPMPGSet

New-DPMPGSet

Remove-DPMPGSet

Get-DPMProtectionGroup

Update-DPMProductionServer

Update-DPMProductionServer

Gets updated information about a protected computer.

Syntax

```
Parameter Set: Default
Update-DPMProductionServer [-ProductionServer] <ProductionServer> [ <CommonParameters>]
```

Detailed Description

The **Update-DPMProductionServer** cmdlet gets updated information about a computer on which a DPM protection agent is installed.

Parameters

-ProductionServer<ProductionServer>

Specifies a computer on which a DPM protection agent is installed.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Update-DPMProductionServer -detailed`".

Related topics

Get-DPMProductionServer

Enable-DPMProductionServer

Disable-DPMProductionServer

Update-DPMProtectionGroup

Update-DPMProtectionGroup

Updates a protection group configuration.

Syntax

```
Parameter Set: Default
Update-DPMProtectionGroup [-ProtectionGroup] <ProtectionGroup> [[-Datasource] <Datasource[]>
] [[-Inquire]] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Update-DPMProtectionGroup** cmdlet refreshes the protection group configuration to update any changes to protected data sources. This cmdlet modifies the protection group without changing any settings to reflect changes to protected data sources.

Parameters

-Datasource<Datasource[]>

Specifies an array of data source objects. Data source objects include the following:

-- Windows file system share or volume.

-- Microsoft SQL Server database.

-- Microsoft Exchange storage group.

-- Microsoft SharePoint Server farm.

-- Microsoft Virtual Machine.

-- Data Protection Manager (DPM) database.

-- A system state that is a member of a protection group.

Aliases	none
Required?	false
Position?	2
Default Value	none
Accept Pipeline Input?	true (ByValue)

Accept Wildcard Characters?	false

-Inquire

Indicates that DPM must perform a fresh inquiry.

Aliases	none
Required?	false
Position?	3
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-ProtectionGroup<ProtectionGroup>

Specifies a **ProtectionGroup** object. To obtain a **ProtectionGroup** object, use the **Get-DPMProtectionGroup** cmdlet.

Aliases	none
Required?	true
Position?	1
Default Value	none
Accept Pipeline Input?	true (ByValue)
Accept Wildcard Characters?	false

-Confirm

Prompts you for confirmation before executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false

Accept Wildcard Characters?	false

-WhatIf

Describes what would happen if you executed the command without actually executing the command.

Required?	false
Position?	named
Default Value	none
Accept Pipeline Input?	false
Accept Wildcard Characters?	false

<CommonParameters>

This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable. For more information, see about_CommonParameters.

Notes

- For more information, type "`Get-Help Update-DPMProtectionGroup -detailed`".

Examples

Example 1: Update a production group

This example refreshes a protection group configuration to update any changes to protected data sources.

The first command gets the protection group on the DPM server named DPMServer02 and store the results in the $Pg variable.

The second command gets the list of protected and unprotected data in the protection group stored in the $Pg variable. The command stores the results in the $Ds variable.

The third command updates the protection group stored in the $Pg variable with the data source stored in the $Ds variable.

```
PS C:\> $Pg=Get-DPMProtectionGroup -DPMServerName "DPMServer02"
PS C:\> $Ds=Get-DPMDatasource -ProtectionGroup $Pg
PS C:\> Update-DPMProtectionGroup -ProtectionGroup $Pg -Datasource $Ds[0] -Inquire
```

Related topics

Get-DPMProtectionGroup

Get-DPMDatasource

Set-DPMProtectionGroup

New-DPMProtectionGroup

Rename-DPMProtectionGroup