

Cmdlet Reference for Windows Azure Pack SQL Server Cmdlets

Microsoft Corporation

Published: November 1, 2013

Applies To

Windows Azure Pack for Windows Server

Feedback

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

Copyright

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet website references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. You may modify this document for your internal, reference purposes.

© 2013 Microsoft Corporation. All rights reserved.

Microsoft, Active Directory, Bing, Excel, Hyper-V, Internet Explorer, Silverlight, SQL Server, Windows, Windows Intune, Windows PowerShell, Windows Server, and Windows Vista are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.

Revision History

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

Contents

Add-MgmtSvcSqlHostingServer	4
Add-MgmtSvcSqlServerGroup	9
Get-MgmtSvcSqlDatabase	13
Get-MgmtSvcSqlHostingServer	17
Get-MgmtSvcSqlHostingServerByGroup	21
Get-MgmtSvcSqlHostingServerMetric	24
Get-MgmtSvcSqlServerGroup	28
Remove-MgmtSvcSqlHostingServer	31
Remove-MgmtSvcSqlServerGroup	35
Set-MgmtSvcSqlHostingServer	38
Test-MgmtSvcSqlHostingServer	43

Add-MgmtSvcSqlHostingServer

Add-MgmtSvcSqlHostingServer

Adds a SQL hosting server to Windows Azure Pack.

Syntax

Parameter Set: ByProperties

```
Add-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [-Name] <String> [-TotalSpaceMB] <Int32> [-User] <PSCredential> [-ServerGroupId] <String> [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: ByObject

```
Add-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [[-HostingServer] <SqlHostingServer> ] [-ServerGroupId] <String> [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-MgmtSvcSqlHostingServer** adds a SQL hosting server to Windows Azure Pack for Windows Server.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-HostingServer<SqlHostingServer>

Specifies a SQL hosting server object.

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

-Name<String>

Specifies the name of a SQL server.

If you want your application databases to be publically accessible, ensure that you use a publically-accessible IP address or FQDN.

Aliases	none
Required?	true
Position?	3
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)

Accept Wildcard Characters?	false
-----------------------------	-------

-ServerGroupId<String>

Specifies the ID for a SQL server group.

Aliases	none
Required?	true
Position?	6
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

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

-TotalSpaceMB<Int32>

Specifies the size, in megabytes (MB) of the hosting server.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)

Accept Wildcard Characters?	false
-----------------------------	-------

-User<PSCredential>

Specifies a user account and password as a PSCredential object. To create a PSCredential object, use the **Get-Credential** cmdlet.

Aliases	none
Required?	true
Position?	5
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: Add a SQL hosting server

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

The first command prompts the user for credentials which are stored in the \$Creds variable.

The second command uses the credentials provided in the first command to add the SQL server named SQLServer01.Contoso.com to the SQL server group with the ID of g5sho0.

```
PS C:\> $Creds = Get-Credential
```

```
PS C:\> Add-MgmtSvcSqlHostingServer -AdminUri "https://Computer01:30004" -Token $Token -  
Name "SQLServer01.Contoso.com" -TotalSpaceMB 2048 -ServerGroupId "g5sho0" -User $Creds
```

Related topics

[Get-MgmtSvcSqlHostingServer](#)

[Set-MgmtSvcSqlHostingServer](#)

[Test-MgmtSvcSqlHostingServer](#)

[Remove-MgmtSvcSqlHostingServer](#)

Add-MgmtSvcSqlServerGroup

Add-MgmtSvcSqlServerGroup

Adds a SQL server group to Windows Azure Pack.

Syntax

Parameter Set: ByProperties

```
Add-MgmtSvcSqlServerGroup [-AdminUri] <Uri> [-Token] <String> [-GroupName] <String> [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: ByObject

```
Add-MgmtSvcSqlServerGroup [-AdminUri] <Uri> [-Token] <String> [[-ServerGroup] <SqlServerGroup> ] [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Add-MgmtSvcSqlServerGroup** adds a SQL server group to Windows Azure Pack for Windows Server.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-GroupName<String>

Specifies a name for the SQL server group.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-ServerGroup<SqlServerGroup>

Specifies a SQL server group object.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByValue)
Accept Wildcard Characters?	false

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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: Add a SQL server group

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This example adds a SQL server group named SQL Group 01.

```
PS C:\> Add-MgmtSvcSqlServerGroup -AdminUri "https://Computer01:30004" -Token $Token -
GroupName "SQL Group 01"
```

Related topics

[Get-MgmtSvcSqlServerGroup](#)

[Remove-MgmtSvcSqlServerGroup](#)

Get-MgmtSvcSqlDatabase

Get-MgmtSvcSqlDatabase

Gets a SQL database.

Syntax

Parameter Set: Default

```
Get-MgmtSvcSqlDatabase [-AdminUri] <Uri> [-Token] <String> [-HostingServerId] <String> [[-Name] <String> ] [-Descending] [-DisableCertificateValidation] [-First <Int32> ] [-Skip <Int32> ] [ <CommonParameters>]
```

Detailed Description

The **Get-MgmtSvcSqlDatabase** cmdlet gets SQL databases. By default, all SQL databases for a hosting server are returned. To get a specific SQL database, use the *Name* parameter.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-Descending

Indicates that the returned databases are displayed in descending order.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-First<Int32>

Gets only the specified number of SQL databases.

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

-HostingServerId<String>

Specifies the ID of a SQL hosting server.

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

-Name<String>

Specifies the name of a SQL database.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Skip<Int32>

Skips the specified number of SQL databases.

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

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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](#).

Examples

Example 1: Get a SQL database by name

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command gets the database named DB01 on the hosting server with the ID u37k25.

```
PS C:\> Get-MgmtSvcSqlDatabase -AdminUri "https://Computer01:30004" -Token $Token -  
HostingServerId "u37k25" -Name "DB01"
```

Related topics

[Get-MgmtSvcSqlHostingServer](#)

Get-MgmtSvcSqlHostingServer

Get-MgmtSvcSqlHostingServer

Gets a SQL hosting server.

Syntax

Parameter Set: Default

```
Get-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [[-Name] <String> ] [-Descending] [-DisableCertificateValidation] [-First <Int32> ] [-Skip <Int32> ] [ <CommonParameters>]
```

Detailed Description

The **Get-MgmtSvcSqlHostingServer** cmdlet gets a SQL hosting server. By default, all hosting servers are returned. To get a specific hosting server, use the *Name* parameter. You can also get a specified number of servers by using the *First* and *Skip* parameters.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-Descending

Indicates that the returned servers are displayed in descending order.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-First<Int32>

Gets only the specified number of SQL hosting servers.

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

-Name<String>

Specifies the name of a SQL hosting server.

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

-Skip<Int32>

Skips the specified number of SQL hosting servers.

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

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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](#).

Examples

Example 1: Get a specific SQL hosting server by name

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command gets the SQL hosting server named SQLServer01.Contoso.com.

```
PS C:\> Get-MgmtSvcSqlHostingServer -AdminUri "https://Computer01:30004" -Token $Token -Name  
" SQLServer01.Contoso.com"
```

Related topics

[Add-MgmtSvcSqlHostingServer](#)

[Set-MgmtSvcSqlHostingServer](#)

[Test-MgmtSvcSqlHostingServer](#)

[Remove-MgmtSvcSqlHostingServer](#)

[Get-MgmtSvcSqlHostingServerByGroup](#)

Get-MgmtSvcSqlHostingServerByGroup

Get-MgmtSvcSqlHostingServerByGroup

Gets SQL hosting servers by SQL server group.

Syntax

Parameter Set: Default

```
Get-MgmtSvcSqlHostingServerByGroup [-AdminUri] <Uri> [-Token] <String> [-ServerGroupId] <String> [[-Name] <String[]> ] [-DisableCertificateValidation] [ <CommonParameters>]
```

Detailed Description

The **Get-MgmtSvcSqlHostingServerByGroup** gets SQL hosting servers by server group. By default, all SQL hosting servers for a group are returned. To get a specific SQL hosting server, use the *Name* parameter.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-Name<String[]>

Specifies an array of SQL hosting server names.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-ServerGroupId<String>

Specifies the ID of a SQL server group.

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

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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](#).

Examples

Example 1: Get all SQL hosting servers for a group

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command gets all SQL hosting servers in the group with the ID of g5sho0.

```
PS C:\> Get-MgmtSvcSqlHostingServerByGroup -AdminUri "https://Computer01:30004" -Token $Token -ServerGroupId "g5sho0"
```

Related topics

[Get-MgmtSvcSqlHostingServer](#)

Get-MgmtSvcSqlHostingServerMetric

Get-MgmtSvcSqlHostingServerMetric

Gets metrics for a SQL hosting server.

Syntax

Parameter Set: Default

```
Get-MgmtSvcSqlHostingServerMetric [-AdminUri] <Uri> [-Token] <String> [-HostingServerId] <String> [[-MetricName] <String[]> ] [[-StartTime] <DateTime> ] [[-EndTime] <DateTime> ] [-DisableCertificateValidation] [ <CommonParameters>]
```

Detailed Description

The **Get-MgmtSvcSqlHostingServerMetric** cmdlet gets metrics for a SQL hosting server. By default, all metrics for a specified SQL hosting server are returned. To get a specific metric, use the *MetricName* parameter. You can also narrow your results by using the *StartTime* and *EndTime* parameters to specify a date range.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-EndTime<DateTime>

Specifies the end time of the date range as a DateTime object. To create a DateTime object, use the **Get-Date** cmdlet.

Aliases	none
Required?	false
Position?	6
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-HostingServerId<String>

Specifies the ID of a SQL hosting server.

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

-MetricName<String[]>

Specifies an array of metric names. You can get the following metrics: DatabaseCount, TotalAllottedSpace.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-StartTime<DateTime>

Specifies the start time of the date range as a DateTime object. To create a DateTime object, use the **Get-Date** cmdlet.

Aliases	none
Required?	false
Position?	5
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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](#).

Examples

Example 1: Get the metrics for DatabaseCount

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command gets the DatabaseCount metrics for the hosting server with the ID of u37k25.

```
PS C:\> Get-MgmtSvcSqlHostingServerMetric -AdminUri "https://Computer01:30004" -Token $Token  
-HostingServerId "u37k25" -MetricName DatabaseCount
```

Related topics

[Get-MgmtSvcSqlHostingServer](#)

Get-MgmtSvcSqlServerGroup

Get-MgmtSvcSqlServerGroup

Gets a SQL server group.

Syntax

Parameter Set: Default

```
Get-MgmtSvcSqlServerGroup [-AdminUri] <Uri> [-Token] <String> [[-GroupName] <String[]> ] [-DisableCertificateValidation] [ <CommonParameters>]
```

Detailed Description

The **Get-MgmtSvcSqlServerGroup** cmdlet gets SQL server groups. By default, all SQL server groups are returned. To get a specific SQL server group, use the *GroupName* parameter.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-GroupName<String[]>

Specifies an array of SQL server group names.

Aliases	none
Required?	false
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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](#).

Examples

Example 1: Get a specific SQL server group

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command gets the SQL server group named SQL Group 01.

```
PS C:\> Get-MgmtSvcSqlServerGroup -AdminUri "https://Computer01:30004" -Token $Token -
GroupName "SQL Group 01"
```

Related topics

[Add-MgmtSvcSqlServerGroup](#)

[Remove-MgmtSvcSqlServerGroup](#)

Remove-MgmtSvcSqlHostingServer

Remove-MgmtSvcSqlHostingServer

Removes a SQL hosting server from Windows Azure Pack.

Syntax

Parameter Set: Default

```
Remove-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [-HostingServerId] <String> [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-MgmtSvcSqlHostingServer** cmdlet removes a hosting server from Windows Azure Pack for Windows Server.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-HostingServerId<String>

Specifies the ID of a SQL hosting server.

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

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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: Remove a SQL hosting server

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command removes the SQL hosting server with the ID of u37k25.

```
PS C:\> Remove-MgmtSvcSqlHostingServer -AdminUri "https://Computer01:30004" -Token $Token -
HostingServerId "u37k25"
```

Related topics

[Add-MgmtSvcSqlHostingServer](#)

[Get-MgmtSvcSqlHostingServer](#)

[Set-MgmtSvcSqlHostingServer](#)

[Test-MgmtSvcSqlHostingServer](#)

Remove-MgmtSvcSqlServerGroup

Remove-MgmtSvcSqlServerGroup

Removes a SQL server group from Windows Azure Pack.

Syntax

Parameter Set: Default

```
Remove-MgmtSvcSqlServerGroup [-AdminUri] <Uri> [-Token] <String> [-ServerGroupId] <String> [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Remove-MgmtSvcSqlServerGroup** cmdlet removes a SQL server group from Windows Azure Pack for Windows Server.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.

If you specify this parameter, you can use self-signed certificates.

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

-ServerGroupId<String>

Specifies the ID of a SQL server group.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

Aliases	none
Required?	true
Position?	2
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: Remove a SQL server group

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

This command removes the SQL server group with the ID g5sho0.

```
PS C:\> Remove-MgmtSvcSqlServerGroup -AdminUri "https://Computer01:30004" -Token $Token -
ServerGroupId "g5sho0"
```

Related topics

[Add-MgmtSvcSqlServerGroup](#)

[Get-MgmtSvcSqlServerGroup](#)

Set-MgmtSvcSqlHostingServer

Set-MgmtSvcSqlHostingServer

Updates a SQL hosting server.

Syntax

Parameter Set: ByProperties

```
Set-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [-Name] <String> [-TotalSpaceMB] <Int32> [-User] <PSCredential> [-HostingServerId] <String> [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: ByObject

```
Set-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [[-HostingServer] <SqlHostingServer> ] [-DisableCertificateValidation] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The **Set-MgmtSvcSqlHostingServer** cmdlet updates a SQL hosting server.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.
If you specify this parameter, you can use self-signed certificates.

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

-HostingServer<SqlHostingServer>

Specifies a SQL hosting server object.

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

-HostingServerId<String>

Specifies the ID of a SQL hosting server.

Aliases	none
Required?	true
Position?	6
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Name<String>

Specifies the name of a SQL hosting server.

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

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

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

-TotalSpaceMB<Int32>

Specifies the size, in megabytes (MB) of the hosting server.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-User<PSCredential>

Specifies a user account and password as a PSCredential object. To create a PSCredential object, use the **Get-Credential** cmdlet.

Aliases	none
Required?	true
Position?	5
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: Update a SQL hosting server

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

The first command prompts the user for credentials and stores the provided user name and password in the \$Credential variable.

The second command updates the total space to 4096 MB for the hosting server named SQLServer01.Contoso.com.

```
PS C:\> $Creds = Get-Credential
```

```
PS C:\> Set-MgmtSvcSqlHostingServer -AdminUri "https://Computer01:30004" -Token $Token -  
Name "SQLServer01.Contoso.com" -TotalSpaceMB 4096 -User $Creds -HostingServerId "u37k25"
```

Related topics

[Add-MgmtSvcSqlHostingServer](#)

[Get-MgmtSvcSqlHostingServer](#)

[Test-MgmtSvcSqlHostingServer](#)

[Remove-MgmtSvcSqlHostingServer](#)

Test-MgmtSvcSqlHostingServer

Test-MgmtSvcSqlHostingServer

Verifies a SQL hosting server.

Syntax

Parameter Set: ByProperties

```
Test-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [-Name] <String> [-TotalSpaceMB] <Int32> [-User] <PSCredential> [-ServerGroupId] <String> [-DisableCertificateValidation] [ <CommonParameters>]
```

Parameter Set: ByObject

```
Test-MgmtSvcSqlHostingServer [-AdminUri] <Uri> [-Token] <String> [[-HostingServer] <SqlHostingServer> ] [-ServerGroupId] <String> [-DisableCertificateValidation] [ <CommonParameters>]
```

Detailed Description

The **Test-MgmtSvcSqlHostingServer** cmdlet verifies a SQL hosting server.

Parameters

-AdminUri<Uri>

Specifies the URI of the Windows Azure Pack administrator API. Use the following format: `https://<computer>:<port>`, where `<computer>` is the computer on which the administrator API is installed.

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

-DisableCertificateValidation

Disables certificate validation for the Windows Azure Pack installation.
If you specify this parameter, you can use self-signed certificates.

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

-HostingServer<SqlHostingServer>

Specifies a SQL hosting server object.

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

-Name<String>

Specifies the name of a SQL server.

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

-ServerGroupId<String>

Specifies the ID for a SQL server group.

Aliases	none
Required?	true
Position?	6
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-Token<String>

Specifies an identity token. To create a token, use the **Get-MgmtSvcToken** cmdlet.

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

-TotalSpaceMB<Int32>

Specifies the size, in megabytes (MB) of the hosting server.

Aliases	none
Required?	true
Position?	4
Default Value	none
Accept Pipeline Input?	True (ByPropertyName)
Accept Wildcard Characters?	false

-User<PSCredential>

Specifies a user account and password as a PSCredential object. To create a PSCredential object, use the **Get-Credential** cmdlet.

Aliases	none
Required?	true
Position?	5
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](#).

Examples

Example 1: Verify a SQL hosting server

NOTE: This example assumes that you have created a token by using **Get-MgmtSvcToken** and have stored it in a variable named \$Token.

The first command prompts the user for credentials and stores the provided user name and password in the \$Credential variable.

The second command tests the SQL hosting server named SQLServer01.Contoso.com, using the credentials provided in the first command.

```
PS C:\> $Creds = Get-Credential
```

```
PS C:\> Test-MgmtSvcSqlHostingServer -AdminUri "https://Computer01:30004" -Token $Token -  
Name "SQLServer01.Contoso.com" -TotalSpaceMB 2048 -User $Creds -ServerGroupId "g5sho0"
```

Related topics

[Add-MgmtSvcSqlHostingServer](#)

[Get-MgmtSvcSqlHostingServer](#)

[Set-MgmtSvcSqlHostingServer](#)

[Remove-MgmtSvcSqlHostingServer](#)

