



Cmdlet Reference for Service Provider Foundation in System Center 2012 SP1

Microsoft Corporation

Published: January 15, 2013

Applies To

System Center 2012 Service Pack 1 (SP1) - Orchestrator

Feedback

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

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet Web site 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.

This document is confidential and proprietary to Microsoft. It is disclosed and can be used only pursuant to a non-disclosure agreement.

© 2011 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
January 15, 2013	Original Release of this guide.

Contents

Get-SCSPFConnectionString	5
Get-SCSPFExtensibleEventHandler	6
Get-SCSPFOffer	8
Get-SCSPFServer	12
Get-SCSPFStamp	15
Get-SCSPFTenant.....	20
Get-SCSPFTenantUserRole.....	25
Get-SCSPFTrustedIssuer.....	28
New-SCSPFETWEvent	31
New-SCSPFOffer	33
New-SCSPFServer.....	38
New-SCSPFStamp.....	41
New-SCSPFTenant	45
New-SCSPFTenantUserRole	50
New-SCSPFTrustedIssuer	53
Remove-SCSPFOffer	56
Remove-SCSPFServer.....	58
Remove-SCSPFStamp.....	60
Remove-SCSPFTenant	62
Remove-SCSPFTenantUserRole	64
Remove-SCSPFTrustedIssuer	66
Set-SCSPFConnectionString.....	68
Set-SCSPFExtensibleEventHandler.....	70

Set-SCSPFOffer	73
Set-SCSPFServer.....	76
Set-SCSPFStamp.....	79
Set-SCSPFTenant	83
Start-SCOrchestratorRunbook.....	87

Get-SCSPFConnectionString

Get-SCSPFConnectionString

Gets the connection string to the Service Provider Foundation database. Requires running Windows PowerShell as Administrator.

Syntax

```
Get-SCSPFConnectionString [ <CommonParameters> ]
```

Detailed Description

The Get-SCSPFConnectionString cmdlet gets the database connection string to the Service Provider Foundation database. The database can be on a different server than the server that is running Service Provider Foundation. This cmdlet requires running Windows PowerShell as Administrator.

Parameters

<CommonParameters>

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

Examples

1: Get the connection string.

This command gets the connection string to the Service Provider Foundation database.

```
PS C:\> Get-SCSPFConnectionString
```

Related topics

[Set-SCSPFConnectionString](#)

Get-SCSPFExtensibleEventHandler

Get-SCSPFExtensibleEventHandler

Gets an extensible event in Service Provider Foundation. Internal use only.

Syntax

Parameter Set: Default

```
Get-SCSPFExtensibleEventHandler -EventName <String> [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFExtensibleEventHandler cmdlet supports the extensibility features in Service Provider Foundation and is not meant to be used directly by Administrators.

Parameters

-EventName<String>

Specifies either **VirtualMachineCreated** or **ServiceCreated** for the event that invokes an Orchestrator runbook. These are the only permitted values for this parameter.

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](#).

Examples

PS C:\> # The Get-SCSPFExtensibleEventHandler cmdlet is for internal use only.

Related topics

[Set-SCSPFExtensibleEventHandler](#)

Get-SCSPFOffer

Get-SCSPFOffer

Gets one or more offer objects.

Syntax

Parameter Set: FromOfferIdParameterSetName

```
Get-SCSPFOffer -ID <Guid[]> [ <CommonParameters>]
```

Parameter Set: FromOfferNameParameterSetName

```
Get-SCSPFOffer -Name <String[]> [ <CommonParameters>]
```

Parameter Set: FromOfferStampParameterSetName

```
Get-SCSPFOffer -Stamp <Stamp> [ <CommonParameters>]
```

Parameter Set: FromOfferTenantParameterSetName

```
Get-SCSPFOffer -Tenant <Tenant> [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFOffer cmdlet gets one or more offer objects in Service Provider Foundation. To create a new offer, use the New-SCSPFOffer cmdlet.

Parameters

-ID<Guid[]>

Specifies one or more identifiers (as a GUID) for a specific object.

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

-Name<String[]>

Specifies the name of one or more offers.

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

-Stamp<Stamp>

Specifies a stamp for which this cmdlet will return associated offers. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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

-Tenant<Tenant>

Specifies a tenant for which this cmdlet will return associated offers. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Get all offers.

This command lists all the existing offers.

```
PS C:\> Get-SCSPFOffer
```

2: Get an offer by its ID.

This command gets an offer by its ID.

```
PS C:\> $offr = Get-SCSPFOffer -ID aeea340e-abe4-478b-af3c-e2c95ccb7a1e
```

3: Get an offer by its name.

This command gets an offer by its name.

```
PS C:\> $offr = Get-SCSPFOffer -Name "Beginner"
```

4: Get an offer that is associated to a particular stamp.

The first command gets a stamp.

The second command gets an offer from the stamp.

```
PS C:\> $stmp = Get-SCSPFStamp -Name "Silver"
```

```
PS C:\> $offr = Get-SCSPFOffer -Stamp $stmp
```

5: Get an offer that is associated to a particular tenant.

The first command gets a tenant.

The second command gets an offer from the tenant.

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "Contoso"
```

```
PS C:\> $offr = Get-SCSPFOffer -Tenant $tnnt
```

Related topics

[New-SCSPFOffer](#)

[Set-SCSPFOffer](#)

[Remove-SCSPFOffer](#)

Get-SCSPFServer

Get-SCSPFServer

Gets one or more server objects.

Syntax

Parameter Set: Empty

```
Get-SCSPFServer [-ServerType <String> ] [ <CommonParameters>]
```

Parameter Set: FromServerIdParameterSetName

```
Get-SCSPFServer -ID <Guid[]> [ <CommonParameters>]
```

Parameter Set: FromServerNameParameterSetName

```
Get-SCSPFServer -Name <String[]> [ <CommonParameters>]
```

Parameter Set: FromServerStampParameterSetName

```
Get-SCSPFServer -Stamp <Stamp> [-ServerType <String> ] [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFServer cmdlet gets one or more server objects in Service Provider Foundation. In the current version of Service Provider Foundation, these are VMM server objects. To create a new server, use the New-SCSPFServer cmdlet.

Parameters

-ID<Guid[]>

Specifies one or more identifiers (as a GUID) for a specific object.

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

-Name<String[]>

Specifies the name of one or more VMM servers.

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

-ServerType<String>

Specifies the type of server. You can specify only 0 for Virtual Machine Manager (VMM) as the server type, other values are reserved for future use.

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

-Stamp<Stamp>

Specifies a stamp for which this cmdlet will return associated servers. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Get all servers.

This command lists all the defined servers.

```
PS C:\> Get-SCSPFServer
```

2: Get a server by its ID.

This command gets a server by its ID.

```
PS C:\> $srvr = Get-SCSPFServer -ID f50e36aa-216f-4ade-a2c0-95cf17b93ee4
```

3: Get a server by its name.

This command gets a server by its name.

```
PS C:\> $srvr = Get-SCSPFServer -Name "VMA1aska"
```

4: Get the server that is associated with a stamp.

This first command gets a stamp.

The second command gets the server associated with the stamp.

```
PS C:\> $stmp = Get-SCSPFStamp -Name "Silver"
```

```
PS C:\> $srvr = Get-SCSPFServer -Stamp $stmp
```

Related topics

[New-SCSPFServer](#)

[Set-SCSPFServer](#)

[Remove-SCSPFServer](#)

Get-SCSPFStamp

Get-SCSPFStamp

Gets one or more stamp objects.

Syntax

Parameter Set: FromStampDataCenterParameterSetName

```
Get-SCSPFStamp -DataCenter <DataCenter> [ <CommonParameters>]
```

Parameter Set: FromStampIdAndServerParameterSetName

```
Get-SCSPFStamp -ID <Guid[]> -Server <Server> [ <CommonParameters>]
```

Parameter Set: FromStampIdAndTenantParameterSetName

```
Get-SCSPFStamp -ID <Guid[]> -Tenant <Tenant> [ <CommonParameters>]
```

Parameter Set: FromStampIdParameterSetName

```
Get-SCSPFStamp -ID <Guid[]> [ <CommonParameters>]
```

Parameter Set: FromStampNameAndServerParameterSetName

```
Get-SCSPFStamp -Name <String[]> -Server <Server> [ <CommonParameters>]
```

Parameter Set: FromStampNameAndTenantParameterSetName

```
Get-SCSPFStamp -Name <String[]> -Tenant <Tenant> [ <CommonParameters>]
```

Parameter Set: FromStampNameParameterSetName

```
Get-SCSPFStamp -Name <String[]> [ <CommonParameters>]
```

Parameter Set: FromStampOfferParameterSetName

```
Get-SCSPFStamp -Offer <Offer> [ <CommonParameters>]
```

Parameter Set: FromStampServerParameterSetName

```
Get-SCSPFStamp -Server <Server> [ <CommonParameters>]
```

Parameter Set: FromStampTenantParameterSetName

```
Get-SCSPFStamp -Tenant <Tenant> [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFStamp cmdlet gets one or more stamp objects in Service Provider Foundation. To create a stamp, use the New-SCSPFStamp cmdlet.

Parameters

-DataCenter<DataCenter>

Specifies the name of a DataCenter object. To obtain a DataCenter object, use the New-Object cmdlet with Microsoft.SystemCenter.Foundation.SPFDData.Types.DataCenter specified for its TypeName parameter. Then set its Name property to the name of the your data center, as shown in the following example.

```
PS C:\> $DS = New-Object Microsoft.SystemCenter.Foundation.SPFDData.Types.DataCenter
PS C:\> $DS.Name = "DataCenter_NorthAmerica"
```

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

-ID<Guid[]>

Specifies one or more identifiers (as a GUID) for a specific object.

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

-Name<String[]>

Specifies the name of one or more stamps.

Aliases	none
Required?	true

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

-Offer<Offer>

Specifies an offer for which this cmdlet will return associated stamps. To obtain an offer, use the Get-SCSPFOffer cmdlet.

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

-Server<Server>

Specifies a server for which this cmdlet will return associated stamps. To obtain a server, use the Get-SCSPFServer cmdlet.

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

-Tenant<Tenant>

Specifies a tenant for which this cmdlet will return associated stamps. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Get all stamps.

This command lists all stamps.

```
PS C:\> Get-SCSPFStamp
```

2. Get a stamp by its ID.

This command gets a stamp by its ID.

```
PS C:\> $stmp = Get-SCSPFStamp -ID db656655-68ff-4a21-bd1d-0a06bdbbc762f
```

3. Get a stamp by its name.

This command gets a stamp by its name.

```
PS C:\> $stmp = Get-SCSPFStamp -Name "Ultimate"
```

4: Get the stamps that are associated with an offer.

The first command gets an offer. The second command gets the stamps associated with the offer.

```
PS C:\> $offer = Get-SCSPFOffer -Name "Beginner"
```

```
PS C:\> $stmp = Get-SCSPFStamp -Offer $offer
```

5: Get the stamp that is associated with a server.

The first command gets a server.

The second command gets the stamp that is associated with the server.

```
PS C:\> $srvr = Get-SCSPFServer -Name "serverVids7"
```

```
PS C:\> $stmps = Get-SCSPFStamp -Server $srvr
```

6: Get the stamps that are associated with a tenant.

The first command gets a tenant.

The second command gets the stamps associated with the tenant.

```
PS C:\> $Contoso = Get-SCSPFTenant -Name "Contoso"
```

```
PS C:\> $stmps = Get-SCSPFStamp -Tenant $Contoso
```

Related topics

[Set-SCSPFStamp](#)

[New-SCSPFStamp](#)

[Remove-SCSPFStamp](#)

Get-SCSPFTenant

Get-SCSPFTenant

Gets one or more tenant objects.

Syntax

Parameter Set: FromTenantIdParameterSetName

```
Get-SCSPFTenant -ID <Guid[]> [ <CommonParameters>]
```

Parameter Set: FromTenantIssuerParameterSetName

```
Get-SCSPFTenant -Issuer <Issuer> [ <CommonParameters>]
```

Parameter Set: FromTenantNameParameterSetName

```
Get-SCSPFTenant -Name <String[]> [ <CommonParameters>]
```

Parameter Set: FromTenantOfferParameterSetName

```
Get-SCSPFTenant -Offer <Offer> [ <CommonParameters>]
```

Parameter Set: FromTenantRoleParameterSetName

```
Get-SCSPFTenant -Role <Role> [ <CommonParameters>]
```

Parameter Set: FromTenantStampParameterSetName

```
Get-SCSPFTenant -Stamp <Stamp> [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFTenant cmdlet gets one or more tenant objects in Service Provider Foundation. To create a tenant, use the New-SCSPFTenant command.

Parameters

-ID<Guid[]>

Specifies one or more identifiers (as a GUID) for a specific object.

Aliases	none
Required?	true
Position?	named

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

-Issuer<Issuer>

Specifies a trusted issuer object. To obtain a trusted issuer, use the Get-SCSPFTrustedIssuer cmdlet.

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

-Name<String[]>

Specifies the name of one or more tenants.

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

-Offer<Offer>

Specifies an offer for which this cmdlet will return associated tenants. To obtain an offer, use the Get-SCSPFOffer cmdlet.

Aliases	none
Required?	true

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

-Role<Role>

Specifies an tenant user role for which this cmdlet will return associated tenants. To obtain a tenant user role, use the Get-SCSPFTenantUserRole cmdlet.

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

-Stamp<Stamp>

Specifies an stamp for which this cmdlet will return associated tenants. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Get all tenants.

This command displays the IDs and name of the existing tenants.

```
PS C:\> Get-SCSPFTenant | Format-Table -Property ID, Name
```

2: Get a tenant by its ID.

This command gets a tenant by its ID.

```
PS C:\> $tnnt = Get-SCSPFTenant -ID e5310370-ab2d-4d51-8336-176999c9cc1e
```

3: Get the tenant that is associated with a trusted issuer.

The first command gets a trusted issuer. The second command gets a tenant associated with the issuer.

```
PS C:\> $ti = Get-SCSPFTrustedIssuer -Name "Litware"
```

```
PS C:\> $tnnt = Get-SCSPFTenant -Issuer $ti
```

4: Get a tenant by its name.

This command gets a tenant by its name.

```
PS C:\> $contoso = Get-SCSPFTenant -Name "Contoso"
```

5: Get the tenants that are associated with an offer.

This command gets a tenant associated to an offer. It first gets a specified offer and then uses the pipeline operator to pass the offer to the Get-SCSPFTenant cmdlet to get the associated tenant.

```
PS C:\> Get-SCSPFOffer -Name "Executive" | Get-SCSPFTenant
```

6: Get the tenant that is associated with a tenant user role.

The first command gets a tenant user role.

The second command gets the tenants associated with the tenant user role.

```
PS C:\> $role_supervisor = Get-SCSPFTenantUserRole -ID e6f9f901-2883-4478-a64e-00c58a423516
```

```
PS C:\> $tnnts = Get-SCSPFTenant -Role $role_supervisor
```

7: Get the tenants that are associated with a stamp.

The first command gets a stamp.

The second command gets the tenants associated with the stamp.

```
PS C:\> $stamp_silver = Get-SCSPFStamp -Name "Silver"
```

```
PS C:\> $tnnts = Get-SCSPFTenant -Stamp $stamp_silver
```

Related topics

[New-SCSPFTenant](#)

[Set-SCSPFTenant](#)

[Remove-SCSPFTenant](#)

Get-SCSPFTenantUserRole

Get-SCSPFTenantUserRole

Gets one or more tenant user role objects.

Syntax

Parameter Set: FromRoleIdParameterSetName

```
Get-SCSPFTenantUserRole [-ID] <Guid[]> [ <CommonParameters>]
```

Parameter Set: FromRoleTenantParameterSetName

```
Get-SCSPFTenantUserRole [-Tenant] <Tenant> [[-Name] <String[]> ] [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFTenantUserRole cmdlet gets one or more tenant role objects in Service Provider Foundation. To create a new tenant user role, use the New-SCSPFTenantUserRole cmdlet.

Parameters

-ID<Guid[]>

Specifies one or more identifiers (as a GUID) for a specific object.

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

-Name<String[]>

Specifies the name of one or more tenant user roles.

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

-Tenant<Tenant>

Specifies a tenant for which this cmdlet will return associated tenant user roles. To obtain a tenant, use the Get-SCSPFTenant 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](#).

Examples

1: Get all tenant user roles.

This command lists all tenant user roles.

```
PS C:\> Get-SCSPFTenantUserRole
```

2: Get a tenant user role by its ID.

This command gets a tenant user role by its ID.

```
PS C:\> $tur = Get-SCSPFTenantUserRole -ID e6f9f901-2883-4478-a64e-00c58a423516
```

3: Get the tenant user role that is associated with a tenant.

The first command gets a tenant.

The second command gets the tenant user roles associated with a tenant.

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "Contoso"
```

```
PS C:\> $tur = Get-SCSPFTenantUserRole -Tenant $tnnt
```

Related topics

[New-SCSPFTenantUserRole](#)

[Remove-SCSPFTenantUserRole](#)

Get-SCSPFTrustedIssuer

Get-SCSPFTrustedIssuer

Gets one or more trusted issuer objects.

Syntax

Parameter Set: FromIssuerIdParameterSetName

```
Get-SCSPFTrustedIssuer -ID <Guid[]> [ <CommonParameters>]
```

Parameter Set: FromIssuerNameParameterSetName

```
Get-SCSPFTrustedIssuer -Name <String[]> [ <CommonParameters>]
```

Parameter Set: FromIssuerTenantParameterSetName

```
Get-SCSPFTrustedIssuer -Tenant <Tenant> [ <CommonParameters>]
```

Detailed Description

The Get-SCSPFTrustedIssuer cmdlet gets one or more trusted issuer objects in Service Provider Foundation. To create a trusted issuer, use the New-SCSPFTrustedIssuer cmdlet.

Parameters

-ID<Guid[]>

Specifies one or more identifiers (as a GUID) for a specific object.

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

-Name<String[]>

Specifies the name of one or more trusted issuers.

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

-Tenant<Tenant>

Specifies a tenant for which this cmdlet will return associated trusted issuers. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Get all trusted issuers.

This command lists all the trusted issuers.

```
PS C:\> Get-SCSPFTrustedIssuer
```

2: Get a trusted issuer by its ID.

This command gets a trusted issuer by its ID.

```
PS C:\> $ti = Get-SCSPFTrustedIssuer -ID c9f59be2-dce5-4221-ad60-2204dc457bd8
```

3: Get a trusted issuer by its name.

This command gets a trusted issuer by its name.

```
PS C:\> $ti = Get-SCSPFTrustedIssuer -Name "Woodgrove"
```

4: Get the trusted issuer that is associated with a tenant.

The first command gets a tenant.

The second command gets the trusted issuer associated with the tenant.

```
PS C:\> $Contoso = Get-SCSPFTenant -Name "Contoso"
```

```
PS C:\> $ti = Get-SCSPFTrustedIssuer -Tenant $Contoso
```

Related topics

[Remove-SCSPFTrustedIssuer](#)

[New-SCSPFTrustedIssuer](#)

New-SCSPFETWEvent

New-SCSPFETWEvent

Adds a new event for ETW logging. Internal use only.

Syntax

Parameter Set: Default

```
New-SCSPFETWEvent -EventName <String> -ProviderName <String> [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

Detailed Description

The New-SCSPFETWEvent cmdlet adds an event for Event Tracing for Windows (ETW) to the event log. This cmdlet is for internal use only and is not meant to be used directly by Administrators.

Parameters

-EventName<String>

Specifies the name of the event to add to the ETW event log. Internal use only.

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

-ProviderName<String>

Specifies an event provider. Internal use only.

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](#).

Examples

PS C:\> # The New-SCSPFETWEvent cmdlet is for internal use only.

New-SCSPFOffer

New-SCSPFOffer

Creates an offer that specifies the stamps available to a tenant administrator user role.

Syntax

Parameter Set: Empty

```
New-SCSPFOffer -Name <String> [-Actions <String> ] [-Clouds <String> ] [-Networks <String> ]  
[-Stamps <Stamp[]> ] [-Templates <String> ] [-Tenants <Tenant[]> ] [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

Detailed Description

The New-SCSPFOffer cmdlet defines stamps available to a tenant administrator user role in Service Provider Foundation. Each tenant administrator that is associated with the offer will have access to the stamps that are specified in the offer.

You can associate an offer with multiple tenants, but a specific tenant can be associated with only one offer. However, a stamp can be associated with multiple offers.

Parameters

-Actions<String>

To be disregarded. Service Provider Foundation does not use this parameter.

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

-Clouds<String>

To be disregarded. Service Provider Foundation does not use this parameter.

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

-Name<String>

Specifies a name for the offer.

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

-Networks<String>

To be disregarded. Service Provider Foundation does not use this parameter.

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

-Stamps<Stamp[]>

Specifies the name of one or more stamp objects to associate with the new offer. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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

-Templates<String>

To be disregarded. Service Provider Foundation does not use this parameter.

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

-Tenants<Tenant[]>

Specifies the name of one or more tenant objects to associate with the new offer. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Create an offer.

The first command gets a collection of tenants.

The second command gets a collection of stamps.

The third command creates a new offer associated with the tenants and stamps.

```
PS C:\> $tnnts = Get-SCSPFTenant | Where-Object {$_.Name -ne Administrator}
```

```
PS C:\> $stamps = Get-SCSPFStamp
```

```
PS C:\> New-SCSPFOffer -Name "Enterprise" -Stamps $stamps -Tenants $tnnts
```

Related topics

[Get-SCSPFOffer](#)

[Set-SCSPFOffer](#)

[Remove-SCSPFOffer](#)

New-SCSPFServer

New-SCSPFServer

Adds a reference to a Virtual Machine Manager (VMM) server in the Service Provider Foundation database.

Syntax

Parameter Set: Empty

```
New-SCSPFServer -Name <String> -ServerType <String> [-Stamps <Stamp[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The New-SCSPFServer cmdlet adds a server to the Service Provider Foundation database. Service Provider Foundation currently supports only VMM servers with the ServerType parameter set to 0 for 'VMM'.

Parameters

-Name<String>

Specifies the name of the VMM server. Specify the fully qualified domain name (FQDN).

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

-ServerType<String>

Specifies the type of server. You can specify only 0 for Virtual Machine Manager (VMM) as the server type, other values are reserved for future use.

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

-Stamps<Stamp[]>

Specifies the name of one or more stamp objects to associate with the new server. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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](#).

Examples

1: Add a server.

The first command gets a stamp.

The second command creates a server with a specified name, a VMM server type, and associates the server with the stamp.

```
PS C:\> $stmp = Get-SCSPFStamp -Name "UltimatePro"
```

```
PS C:\> New-SCSPFServer -Name "server23G" -ServerType 0 -Stamps $stmp
```

Related topics

[Get-SCSPFServer](#)

[Set-SCSPFServer](#)

[Remove-SCSPFServer](#)

New-SCSPFStamp

New-SCSPFStamp

Creates a logical scale unit for servers and instances of System Center components.

Syntax

Parameter Set: Empty

```
New-SCSPFStamp -Name <String> [-DataCenter <DataCenter> ] [-Offers <Offer[]> ] [-Servers <Server[]> ] [-Tenants <Tenant[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The New-SCSPFStamp cmdlet creates a new stamp, a logical scale unit that specifies a System Center component instance running on a server. As tenants need additional capacity, additional stamps can be provided. Currently, Service Provider Foundation supports only one type of stamp: one VMM instance on a server.

Parameters

-DataCenter<DataCenter>

Specifies the name of a DataCenter object. To obtain a DataCenter object, use the New-Object cmdlet with Microsoft.SystemCenter.Foundation.SPFDData.Types.DataCenter specified for its TypeName parameter. Then set its Name property to the name of the your data center, as shown in the following example.

```
PS C:\> $DS = New-Object Microsoft.SystemCenter.Foundation.SPFDData.Types.DataCenter
PS C:\> $DS.Name = "DataCenter_NorthAmerica"
```

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

-Name<String>

Specifies a name for the stamp.

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

-Offers<Offer[]>

Specifies the name of one or more offer objects to associate with the new stamp. To obtain an offer, use the Get-SCSPFOffer cmdlet.

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

-Servers<Server[]>

Specifies the name of one or more server objects to associate with the new stamp. To obtain a server, use the Get-SCSPFServer cmdlet.

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

-Tenants<Tenant[]>

Specifies the name of one or more tenant objects to associate with the new stamp. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Create a stamp.

This command creates a new stamp.

```
PS C:\> New-SCSPFStamp -Name "Introductory"
```

Related topics

[Set-SCSPFStamp](#)

[Get-SCSPFStamp](#)

[Remove-SCSPFStamp](#)

New-SCSPFTenant

New-SCSPFTenant

Creates a new tenant for a hoster.

Syntax

Parameter Set: Empty

```
New-SCSPFTenant -Name <String> [-AccountStatus <String> ] [-Offer <Offer> ] [-Stamps <Stamp[]> ] [-SubscriptionId <Guid> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: FromTenantCertParameterSetName

```
New-SCSPFTenant -Certificate <String> -IssuerName <String> -Name <String> [-AccountStatus <String> ] [-Offer <Offer> ] [-Stamps <Stamp[]> ] [-SubscriptionId <Guid> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: FromTenantIssuereParameterSetName

```
New-SCSPFTenant -IssuerName <String> -Key <String> -Name <String> [-AccountStatus <String> ] [-Offer <Offer> ] [-Stamps <Stamp[]> ] [-SubscriptionId <Guid> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The New-SCSPFTenant cmdlet creates a new tenant in Service Provider Foundation. If you are a service provider, these are your paying customers. If you are using Service Provider Foundation in private cloud, these are the business units of your organization.

Parameters

-AccountStatus<String>

Sets the status of a tenant. Specify 0 for Active, 1 for Suspended.

Aliases	none
Required?	false
Position?	named
Default Value	none

Accept Pipeline Input?	false
Accept Wildcard Characters?	false

-Certificate<String>

Specifies the path to the certificate file for the tenant.

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

-IssuerName<String>

Specifies the name of the party that issued the certificate for the tenant. This parameter is not to be confused with a trusted issuer object, as it is metadata to verify a token submitted by the tenant.

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

-Key<String>

Specifies the public key that validates a signed token submitted by a tenant in claims-based authentication.

Aliases	none
Required?	true

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

-Name<String>

Specifies the name to give to the tenant.

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

-Offer<Offer>

Specifies the name of one or more offer objects to associate with the new tenant. To obtain an offer, use the Get-SCSPFOffer cmdlet.

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

-Stamps<Stamp[]>

Specifies the name of one or more stamp objects to associate with the new tenant. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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

-SubscriptionId<Guid>

Specifies the subscription identifier (as a GUID) for a tenant. This value cannot be changed after the tenant is created.

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](#).

Examples

1: Onboard a tenant.

The first two commands create the \$cert variable that contains the tenant's issued certificate.

The third command creates the \$key variable that contains the certificate's public key.

The fourth command creates the \$subID variable that contains a GUID for the subscription ID.

The fifth command creates the tenant with the name of the trusted issuer of the certificate, the public key, a specified name, and the subscription ID.

```
PS C:\> $path = "C:\Temp\ADatum29D.cer"
```

```
PS C:\> $cert = New-Object  
System.Security.Cryptography.X509Certificates.X509Certificate2($path)
```

```
PS C:\> $key = [Convert]::ToBase64String($cert.RawData())
```

```
PS C:\> $subID = [System.Guid]::NewGuid().ToString()
```

```
PS C:\> New-SCSPFTenant -IssuerName "Woodgrove" -Key $key -Name "ADatum" -SubscriptionID  
$subID
```

Related topics

[Remove-SCSPFTenant](#)

[Set-SCSPFTenant](#)

[Get-SCSPFTenant](#)

New-SCSPFTenantUserRole

New-SCSPFTenantUserRole

Creates an association between a tenant and one or more self-service user roles.

Syntax

Parameter Set: Default

```
New-SCSPFTenantUserRole -Name <String[]> -Tenant <Tenant> [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

Detailed Description

The New-SCSPFTenantUserRole cmdlet creates an association between a tenant and one or more self-service user roles in Service Provider Foundation. The self-service roles must already have been created in Virtual Machine Manager (VMM) or by using the New-SCUserRole cmdlet.

You cannot use this cmdlet to create an association with a tenant administrator user role. The tenant administrator user role is created automatically when the tenant is created by the [New-SCSPFTenant](#) cmdlet.

Parameters

-Name<String[]>

Specifies the name of one or more tenant user roles.

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

-Tenant<Tenant>

Specifies a tenant object to associate with the new tenant user role. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Create a tenant user role.

The first command defines an array of self-service user roles.

The second command gets a tenant.

The third command creates the tenant user role for the tenant with its individual self-service user roles.

```
PS C:\> $roles = "Assistant", "Admin", "Developer"
```

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "Contoso"
```

```
PS C:\> New-SCSPFTenantUserRole -Name $roles -Tenant $tnnt
```

Related topics

[Get-SCSPFTenantUserRole](#)

[Remove-SCSPFTenantUserRole](#)

New-SCSPFTrustedIssuer

New-SCSPFTrustedIssuer

Creates a trusted issuer for claims-based authentication.

Syntax

Parameter Set: Empty

```
New-SCSPFTrustedIssuer -Key <String> -Name <String> [-Tenant <Tenant> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The New-SCSPFTrustedIssuer cmdlet creates certification credentials for claims-based authentication in Service Provider Foundation. The hoster obtains the public key of the certificate's key value pair from the tenant (or on behalf of a tenant). The trusted issuer object is then used to validate signed tokens from the tenant so that access to resources can be granted.

Parameters

-Key<String>

Specifies the public key that validates a signed token submitted by a tenant in claims-based authentication.

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

-Name<String>

Specifies a name to give to the trusted issuer.

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

-Tenant<Tenant>

Specifies a tenant object to associate with the new trusted issuer. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Create a trusted issuer.

This command creates a trusted issuer with a previously defined key and a specified name.

```
PS C:\> New-SCSPFTrustedIssuer -Key $key -Name "Trey"
```

Related topics

[Remove-SCSPFTrustedIssuer](#)

[Get-SCSPFTrustedIssuer](#)

Remove-SCSPFOffer

Remove-SCSPFOffer

Removes one or more offer objects.

Syntax

Parameter Set: FromOfferParameterSetName

```
Remove-SCSPFOffer -Offer <Offer[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Remove-SCSPFOffer cmdlet removes one or more offers from the Service Provider Foundation database.

Parameters

-Offer<Offer[]>

Specifies the name of one or more offer objects. To obtain an offer, use the Get-SCSPFOffer cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Remove an offer.

The first command gets the offer object by its ID and stores the object in the \$offer variable.

The second command removes the offer.

```
PS C:\> $offer = Get-SCSPFOffer -ID 64cc59ff-ae54-41f8-b6bd-c811361df6aa
```

```
PS C:\> Remove-SCSPFOffer -Offer $offer
```

Related topics

[New-SCSPFOffer](#)

[Set-SCSPFOffer](#)

[Get-SCSPFOffer](#)

Remove-SCSPFServer

Remove-SCSPFServer

Removes one or more server objects.

Syntax

Parameter Set: FromServerParameterSetName

```
Remove-SCSPFServer -Server <Server[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Remove-SCSPFServer cmdlet removes one or more VMM servers from the Service Provider Foundation database.

Parameters

-Server<Server[]>

Specifies the name of one or more VMM server objects. To obtain a server, use the Get-SCSPFServer cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Remove a server.

The first command gets the server object named server23G and stores the object in the \$srvr variable. The second command removes the server.

```
PS C:\> $srvr = Get-SCSPFServer -Name "server23G"
```

```
PS C:\> Remove-SCSPFServer -Server $srvr
```

Related topics

[New-SCSPFServer](#)

[Set-SCSPFServer](#)

[Get-SCSPFServer](#)

Remove-SCSPFStamp

Remove-SCSPFStamp

Removes one or more stamp objects.

Syntax

Parameter Set: FromStampParameterSetName

```
Remove-SCSPFStamp -Stamp <Stamp[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Remove-SCSPFStamp cmdlet removes one or more stamps from the Service Provider Foundation database.

Parameters

-Stamp<Stamp[]>

Specifies the name of one or more stamp objects. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Remove a stamp.

The first command gets the stamp object named DeLuxe and stores the object in the \$stmp variable.

The second command removes the stamp.

```
PS C:\> $stmp = Get-SCSPFStamp -Name "DeLuxe"
```

```
PS C:\> Remove-SCSPFStamp -Stamp $stmp
```

Related topics

[Set-SCSPFStamp](#)

[New-SCSPFStamp](#)

[Get-SCSPFStamp](#)

Remove-SCSPFTenant

Remove-SCSPFTenant

Removes one or more tenant objects.

Syntax

Parameter Set: Default

```
Remove-SCSPFTenant -Tenant <Tenant[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Remove-SCSPFTenant cmdlet removes one or more tenants from the Service Provider Foundation database.

Parameters

-Tenant<Tenant[]>

Specifies the name of one or more tenant objects. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Remove a tenant.

The first command gets the tenant object named Fabrikam and stores the object in the \$tnnt variable. The second command removes the tenant.

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "Fabrikam"
```

```
PS C:\> Remove-SCSPFTenant -Tenant $tnnt
```

Related topics

[New-SCSPFTenant](#)

[Set-SCSPFTenant](#)

[Get-SCSPFTenant](#)

Remove-SCSPFTenantUserRole

Remove-SCSPFTenantUserRole

Removes one or more tenant user role objects.

Syntax

Parameter Set: Default

```
Remove-SCSPFTenantUserRole -UserRole <Role[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Remove-SCSPFTenantUserRole cmdlet removes one or more tenant user roles from the Service Provider Foundation database.

Parameters

-UserRole<Role[]>

Specifies the name of one or more tenant user role objects. To obtain a tenant user role, use the Get-SCSPFTenantUserRole cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Remove a tenant user role.

The first command gets the tenant user role object by its ID and stores the object in the \$tur variable.

The second command removes the tenant user role.

```
PS C:\> $tur = Get-SCSPFTenantUserRole -ID 9adb708d-f47e-4dda-9e56-91e1eb2808df
```

```
PS C:\> Remove-SCSPFTenantUserRole -UserRole $tur
```

Related topics

[New-SCSPFTenantUserRole](#)

[Get-SCSPFTenantUserRole](#)

Remove-SCSPFTrustedIssuer

Remove-SCSPFTrustedIssuer

Removes one or more trusted issuer objects.

Syntax

Parameter Set: Default

```
Remove-SCSPFTrustedIssuer -Issuer <Issuer[]> [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Remove-SCSPFTrustedIssuer cmdlet removes one or more trusted issuers from the Service Provider Foundation database.

Parameters

-Issuer<Issuer[]>

Specifies one or more trusted issuer objects. To obtain a trusted issuer object, use the Get-SCSPFTrustedIssuer cmdlet.

Aliases	none
Required?	true
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](#).

Examples

1: Remove a trusted issuer.

The first command gets the trusted issuer object by its ID and stores the object in the \$ti variable.

The second command removes the trusted issuer.

```
PS C:\> $ti = Get-SCSPFTrustedIssuer -ID 938d2412-ba64-497e-b97d-12991087d66c
```

```
PS C:\> Remove-SCSPFTrustedIssuer -Issuer $ti
```

Related topics

[New-SCSPFTrustedIssuer](#)

[Get-SCSPFTrustedIssuer](#)

Set-SCSPFConnectionString

Set-SCSPFConnectionString

Sets the connection string to the Service Provider Foundation database. Requires running Windows PowerShell as Administrator.

Syntax

```
Parameter Set: FromConnectionStringConnectionStringParameterSetName  
Set-SCSPFConnectionString -ConnectionString <String> [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

Detailed Description

The Set-SCSPFConnectionString cmdlet sets the database connection string to the Service Provider Foundation database that is used by the server running Service Provider Foundation. The database can be on a separate server. This cmdlet requires running Windows PowerShell as Administrator.

Parameters

-ConnectionString<String>

Specifies the database connection string to a Service Provider Foundation database.

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](#).

Examples

1: Set the connection string.

This command sets the connection string to the Service Provider Foundation database on a server named serverSFPDB_7.

```
PS C:\> Set-SCSPFConnectionString -ConnectionString "Data
Source=serverSFPDB_7,1433;Database=SCSPFDB;Integrated
Security=True;MultipleActiveResultSets=True;"
```

Related topics

[Get-SCSPFConnectionString](#)

Set-SCSPFExtensibleEventHandler

Set-SCSPFExtensibleEventHandler

Sets an extensible event in Service Provider Foundation.

Syntax

Parameter Set: Default

```
Set-SCSPFExtensibleEventHandler -EventName <String> [-Enable] [-OrchestratorUri <String> ]  
[-RunbookPath <String> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Set-SCSPFExtensibleEventHandler cmdlet enables access to extensibility features in Service Provider Foundation. You can set an Orchestrator Runbook to be run at the completion of **CreateVirtualMachined** and **CreatedService** events.

Parameters

-Enable

Enables the extensible event handler. Specify this parameter by itself to enable the handler. There is no associated value. Omit this parameter to disable the handler.

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

-EventName<String>

Specifies either **VirtualMachineCreated** or **ServiceCreated** for the event that invokes an Orchestrator runbook. These are the only permitted values for this parameter.

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

-OrchestratorUri<String>

Specifies the URI to the Orchestrator web service as specified in the web.config file at C:\Program Files (x86)\Microsoft System Center 2012\Orchestrator\Orchestration Console.

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

-RunbookPath<String>

Specifies the path to a runbook in System Center 2012 - Orchestrator.

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](#).

Examples

1: Set an event handler.

This command enables the event handler to invoke a Runbook.

```
PS C:\> Set-SCSPFExtensibleEventHandler -EventName "VirtualMachineCreated" -OrchestratorUri  
"http://scxspf-ws8-24:82/Orchestrator2012/Orchestrator.svc" -RunbookPath "\SPF  
Runbooks\Extensibility\VM Created" -Enable
```

Set-SCSPFOffer

Set-SCSPFOffer

Associates an offer with tenants and stamps.

Syntax

Parameter Set: Default

```
Set-SCSPFOffer -Offer <Offer> [-Stamps <Stamp[]> ] [-Tenants <Tenant[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Set-SCSPFOffer cmdlet associates an offer with one or more tenants, and with one or more stamps in Service Provider Foundation. An offer specifies the stamps available to a tenant administrator role. You can associate an offer with multiple tenants, but a specific tenant can be associated with only one offer. However, a stamp can be associated with multiple offers.

If you want to add an association of a stamp to an offer without changing the stamps currently associated with the offer, use the [Set-SCSPFStamp](#) cmdlet.

Parameters

-Offer<Offer>

Specifies the offer object to be set to new values by the other parameters in this cmdlet. To obtain an offer, use the Get-SCSPFOffer cmdlet.

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

-Stamps<Stamp[]>

Specifies one or more stamps to associate with the offer. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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

-Tenants<Tenant[]>

Specifies one or more tenants to associate with the offer. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

If you set a tenant that already is associated with an offer to a new offer, that tenant will be associated with the new offer.

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](#).

Examples

1: Set an offer to a tenant and stamp.

The first three commands get an offer, a tenant, and a stamp.

The fourth command associates the tenant and stamp with the offer.

```
PS C:\> $offer = Get-SCSPFOffer -ID 64cc59ff-ae54-41f8-b6bd-c811361df6aa
```

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "ADatum"
```

```
PS C:\> $stamp = Get-SCSPFStamp -Name "Platinum"
```

```
PS C:\> Set-SCSPFOffer -Offer $offer -Stamps $stamp -Tenants $tnnt
```

Related topics

[New-SCSPFOffer](#)

[Get-SCSPFOffer](#)

[Remove-SCSPFOffer](#)

Set-SCSPFServer

Set-SCSPFServer

Associates a server with one or more stamps.

Syntax

Parameter Set: Default

```
Set-SCSPFServer -Server <Server> [-Stamps <Stamp[]> ] [-Confirm] [-WhatIf] [  
<CommonParameters>]
```

Detailed Description

The Set-SCSPFServer cmdlet associates a server with one or more stamps in Service Provider Foundation. However, you can associate a VMM server with only one stamp, and Service Provider Foundation supports only VMM servers.

If you want to add an association of a stamp to a server without changing the stamps currently associated with the server, use the [Set-SCSPFStamp](#) cmdlet.

Parameters

-Server<Server>

Specifies the server object to be set to new values by the other parameters in this cmdlet. To obtain a server, use the Get-SCSPFServer cmdlet.

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

-Stamps<Stamp[]>

Specifies one or more stamps to associate with the server. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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](#).

Examples

1: Set a server to a stamp.

The first two commands get a stamp and a server.

The third command associates the server with the stamp.

```
PS C:\> $stamp_silver = Get-SCSPFStamp -Name "Silver"
```

```
PS C:\> $server = Get-SCSPFServer -Name "serverD7"
```

```
PS C:\> Set-SCSPFServer -Server $server -Stamps $stamp_silver
```

Related topics

[New-SCSPFServer](#)

[Get-SCSPFServer](#)

[Remove-SCSPFServer](#)

Set-SCSPFStamp

Set-SCSPFStamp

Associates a stamp with offers, servers, and tenants.

Syntax

Parameter Set: Default

```
Set-SCSPFStamp -Stamp <Stamp> [-Offers <Offer[]> ] [-Servers <Server[]> ] [-Tenants <Tenant[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Set-SCSPFStamp cmdlet associates a stamp with one or more servers, one or more tenants, and one or more offers in Service Provider Foundation.

If you want to add an offer, server, or tenant to a stamp without changing the currently associated objects, use the [Set-SCSPFOffer](#), [Set-SCSPFServer](#), or [Set-SCSPFTenant](#) cmdlets.

Parameters

-Offers<Offer[]>

Specifies one or more offers to associate with the stamp. To obtain an offer, use the Get-SCSPFOffer cmdlet.

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

-Servers<Server[]>

Specifies one or more servers to associate with the stamp. To obtain a server, use the Get-SCSPFServer cmdlet.

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

-Stamp<Stamp>

Specifies the stamp object to be set to new values by the other parameters in this cmdlet. To obtain a stamp, use the Get-SCSPFStamp cmdlet.

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

-Tenants<Tenant[]>

Specifies the name of one or more tenant objects. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Set a stamp to a tenant.

The first two commands get a stamp and a tenant.

The third command associates the tenant with the stamp.

```
PS C:\> $stamp = Get-SCSPFStamp -ID db656655-68ff-4a21-bd1d-0a06bdbc762f
```

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "AdventureWorks"
```

```
PS C:\> Set-SCSPFStamp -Stamp $stamp -Tenants $tnnt
```

Related topics

[Get-SCSPFStamp](#)

[New-SCSPFStamp](#)

[Remove-SCSPFStamp](#)

Set-SCSPFTenant

Set-SCSPFTenant

Sets resources and data for a tenant.

Syntax

Parameter Set: Empty

```
Set-SCSPFTenant -Tenant <Tenant> [-AccountStatus <String> ] [-Offer <Offer> ] [-Stamps <Stamp[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Parameter Set: FromTenantIssuereParameterSetName

```
Set-SCSPFTenant -IssuerName <String> -Key <String> -Tenant <Tenant> [-AccountStatus <String> ] [-Offer <Offer> ] [-Stamps <Stamp[]> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Set-SCSPFTenant cmdlet associates a tenant in Service Provider Foundation with various data, including an offer, stamps, and values for claims-based authentication.

If you want to add an association of a stamp to a tenant without changing the stamps currently associated with the tenant, use the [Set-SCSPFStamp](#) cmdlet.

Parameters

-AccountStatus<String>

Sets the status of a tenant. Specify 0 for Active, 1 for Suspended.

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

-IssuerName<String>

Specifies the name of the party that issued the certificate for the tenant. This parameter is not to be confused with a trusted issuer object, as it is metadata to verify a token submitted by the tenant.

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

-Key<String>

Specifies the public key that validates a signed token submitted by a tenant in claims-based authentication.

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

-Offer<Offer>

Specifies an offer to associate with the tenant. To obtain an offer, use the Get-SCSPFOffer cmdlet.

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

-Stamps<Stamp[]>

Specifies one or more stamps to associate with the tenant. To obtain a stamp, use the New-SCSPFStamp cmdlet.

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

-Tenant<Tenant>

Specifies the tenant object to be set to new values by the other parameters in this cmdlet. To obtain a tenant, use the Get-SCSPFTenant cmdlet.

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](#).

Examples

1: Set a tenant's account status.

This command sets the account status for a tenant to 1 (Suspended).

```
PS C:\> Set-SCSPFTenant -Tenant (Get-SCSPFTenant -Name "ADatum") -AccountStatus 1
```

2: Set a tenant to collection of stamps.

The first two commands get a tenant and a collection of stamps.

The third command associates the tenant with the stamps.

```
PS C:\> $tnnt = Get-SCSPFTenant -Name "Northwind"
PS C:\> $stmps = Get-SCSPFServer | Where-Object {
>> $_.Name -contains "silver">> }
PS C:\> Set-SCSPFTenant -Tenant $tnnt -Stamps $stmps
```

Related topics

[New-SCSPFTenant](#)

[Remove-SCSPFTenant](#)

[Get-SCSPFTenant](#)

Start-SCOrchestratorRunbook

Start-SCOrchestratorRunbook

Invokes a runbook in System Center 2012 - Orchestrator. Internal use only.

Syntax

Parameter Set: Default

```
Start-SCOrchestratorRunbook -RunbookPath <String> -Uri <String> [-RunbookParameters <Hashtable> ] [-Confirm] [-WhatIf] [ <CommonParameters>]
```

Detailed Description

The Start-SCOrchestratorRunbook cmdlet invokes the runbook as specified by the RunbookPath parameter for the Set-SCSPFExtensibleEventHandler command. This action is performed automatically by Service Provider Foundation and is not meant to be used by Administrators.

Parameters

-RunbookParameters<Hashtable>

Specifies the parameters for a runbook in System Center 2012 - Orchestrator.

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

-RunbookPath<String>

Specifies the path to a runbook in Orchestrator.

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

-Uri<String>

Specifies the Uri to the runbook in Orchestrator.

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](#).

Examples

PS C:\> # The Start-SCOrchestratorRunbook cmdlet is for internal use only.