



python



PowerShell

FPDF Library  
PDF generator

*Full credit is given to the above companies including the OS that this PDF file was generated!*

### **PowerShell Get-Help on command 'Set-PSSessionConfiguration'**

**PS C:\Users\wahid> Get-Help Set-PSSessionConfiguration**

#### NAME

Set-PSSessionConfiguration

#### SYNOPSIS

Changes the properties of a registered session configuration.

#### SYNTAX

```
Set-PSSessionConfiguration [-Name] <System.String> [-AccessMode {Disabled |  
Local | Remote}] [-ApplicationBase <System.String>] [-Force]  
[-MaximumReceivedDataSizePerCommandMB <System.Nullable`1[System.Double]>]  
[-MaximumReceivedObjectSizeMB <System.Nullable`1[System.Double]>]  
[-ModulesToImport <System.Object[]>] [-NoServiceRestart] [-PSVersion  
<System.Version>] [-RunAsCredential  
<System.Management.Automation.PSCredential>] [-SecurityDescriptorSddl  
<System.String>] [-SessionTypeOption  
<System.Management.Automation.PSSessionTypeOption>]  
[-ShowSecurityDescriptorUI] [-StartupScript <System.String>]  
[-ThreadApartmentState {STA | MTA | Unknown}] [-ThreadOptions {Default |  
UseNewThread | ReuseThread | UseCurrentThread}] [-TransportOption  
<System.Management.Automation.PSTransportOption>] [-UseSharedProcess]
```

[-Confirm] [-WhatIf] [<CommonParameters>]

Set-PSSessionConfiguration [-Name] <System.String> [-AssemblyName]  
<System.String> [-ConfigurationTypeName] <System.String> [-AccessMode  
{Disabled | Local | Remote}] [-ApplicationBase <System.String>] [-Force]  
[-MaximumReceivedDataSizePerCommandMB <System.Nullable`1[System.Double]>]  
[-MaximumReceivedObjectSizeMB <System.Nullable`1[System.Double]>]  
[-ModulesToImport <System.Object[]>] [-NoServiceRestart] [-PSVersion  
<System.Version>] [-RunAsCredential  
<System.Management.Automation.PSCredential>] [-SecurityDescriptorSddl  
<System.String>] [-SessionTypeOption  
<System.Management.Automation.PSSessionTypeOption>]  
[-ShowSecurityDescriptorUI] [-StartupScript <System.String>]  
[-ThreadApartmentState {STA | MTA | Unknown}] [-ThreadOptions {Default |  
UseNewThread | ReuseThread | UseCurrentThread}] [-TransportOption  
<System.Management.Automation.PSTransportOption>] [-UseSharedProcess]  
[-Confirm] [-WhatIf] [<CommonParameters>]

Set-PSSessionConfiguration [-Name] <System.String> [-AccessMode {Disabled |  
Local | Remote}] [-Force] [-MaximumReceivedDataSizePerCommandMB  
<System.Nullable`1[System.Double]>] [-MaximumReceivedObjectSizeMB  
<System.Nullable`1[System.Double]>] [-NoServiceRestart] -Path <System.String>  
[-RunAsCredential <System.Management.Automation.PSCredential>]  
[-SecurityDescriptorSddl <System.String>] [-ShowSecurityDescriptorUI]  
[-StartupScript <System.String>] [-ThreadApartmentState {STA | MTA | Unknown}]  
[-ThreadOptions {Default | UseNewThread | ReuseThread | UseCurrentThread}]  
[-TransportOption <System.Management.Automation.PSTransportOption>]  
[-UseSharedProcess] [-Confirm] [-WhatIf] [<CommonParameters>]

## DESCRIPTION

The `Set-PSSessionConfiguration` cmdlet changes the properties of the session configurations on the local computer.

Use the Name parameter to identify the session configuration that you want to change. Use the other parameters to specify new values for the properties of the session configuration. To delete a property value from the configuration, and use the default value, enter an empty string (``) or a value of ``$Null`` for the corresponding parameter.

Starting in PowerShell 3.0, you can use a session configuration file to define a session configuration. This feature provides a simple and discoverable method for setting and changing the properties of sessions that use the session configuration. To specify a session configuration file, use the Path parameter of ``Set-PSSessionConfiguration``. For information about session configuration files, see [about\\_Session\\_Configuration\\_Files](#) (About/about\_Session\_Configuration\_Files.md). For information about how to create and modify a session configuration file, see the ``New-PSSessionConfigurationFile`` cmdlet.

Session configurations define the environment of remote sessions ( `PSSessions` ) that connect to the local computer. Every `PSSession` uses a session configuration. The session configuration determines the features of the `PSSession` , such as the modules that are available in the session, the cmdlets that are permitted to run, the language mode, quotas, and timeouts. The security descriptor of the session configuration determines who can use the session configuration to connect to the local computer. For more information about session configurations, see [about\\_Session\\_Configurations](#) (About/about\_Session\_Configurations.md).

To see the properties of a session configuration, use the ``Get-PSSessionConfiguration`` cmdlet or the WSMAN Provider. For more information about the WSMAN Provider, type ``Get-Help WSMAN``.

-AccessMode

<System.Management.Automation.Runspaces.PSSessionConfigurationAccessMode>

Enables and disables the session configuration and determines whether it can be used for remote or local sessions on the computer. The acceptable values for this parameter are:

- `Disabled`. Disables the session configuration. It cannot be used for remote or local access to the computer. This value sets the Enabled property of the session configuration

(`WSMan:<ComputerName>\PlugIn<SessionConfigurationName>\Enabled`) to

`False`. - `Local`. Adds a Network\_Deny\_All entry to security descriptor

of the session configuration. Users of the local computer can use the session configuration to create a local loopback session on the same

computer, but remote users are denied access. - `Remote`. Removes Deny\_All

and Network\_Deny\_All entries from the security descriptors of the

session configuration. Users of local and remote computers can use the

session configuration to create sessions and run commands on this

computer.

The default value is Remote .

Other cmdlets can override the value of this parameter later. For example,

the `Enable-PSRemoting` cmdlet enables all session configurations on the

computer and permits remote access to them, and the `Disable-PSRemoting`

cmdlet permits only local access to all session configurations on the

computer.

This parameter was introduced in PowerShell 3.0.

-ApplicationBase <System.String>

Specifies the path of the assembly file (\*.dll) that is specified in the value of the AssemblyName parameter.

-AssemblyName <System.String>

Specifies the assembly name. This cmdlet creates a session configuration based on a class that is defined in an assembly.

Enter the filename or full path of an assembly `.dll` file that defines a session configuration. If you enter only the file name, you can enter the path in the value of the `ApplicationBase` parameter.

-ConfigurationTypeName <System.String>

Specifies the type of the session configuration that is defined in the assembly in the `AssemblyName` parameter. The type that you specify must implement the `System.Management.Automation.Remoting.PSSessionConfiguration` class.

This parameter is required when you specify an assembly name.

-Force <System.Management.Automation.SwitchParameter>

Suppresses all user prompts, and restarts the WinRM service without prompting. Restarting the service makes the configuration change effective.

To prevent a restart and suppress the restart prompt, use the `NoServiceRestart` parameter.

-MaximumReceivedDataSizePerCommandMB <System.Nullable`1[System.Double]>

Specifies the limit on the amount of data that can be sent to this computer in any single remote command. Enter the data size in megabytes (MB). The default is `50`.

If a data size limit is defined in the configuration type that is specified in the `ConfigurationTypeName` parameter, the limit in the configuration type is used. The value of this parameter is ignored.

-MaximumReceivedObjectSizeMB <System.Nullable`1[System.Double]>

Specifies the limits on the amount of data that can be sent to this computer in any single object. Enter the data size in megabytes. The default is `10`.

If an object size limit is defined in the configuration type that is specified in the ConfigurationTypeName parameter, the limit in the configuration type is used. The value of this parameter is ignored.

**-ModulesToImport <System.Object[]>**

Specifies the modules and snap-ins that are automatically imported into sessions that use the session configuration. Enter the module and snap-in names.

By default, only the Microsoft.PowerShell.Core module is imported into sessions, but unless the cmdlets are excluded, you can use the `Import-Module` and `Add-PSSnapin` cmdlets to add modules and snap-ins to the session.

The modules specified in this parameter value are imported in additions to modules specified in the session configuration file (`New-PSSessionConfigurationFile`). However, settings in the session configuration file can hide the commands exported by modules or prevent users from using them.

The modules specified in this parameter value replace the list of modules specified with the ModulesToImport parameter of the `Register-PSSessionConfiguration` cmdlet.

This parameter was introduced in PowerShell 3.0.

**-Name <System.String>**

Specifies the name of the session configuration that you want to change.

You cannot use this parameter to change the name of the session configuration.

`-NoServiceRestart <System.Management.Automation.SwitchParameter>`

Does not restart the WinRM service, and suppresses the prompt to restart the service.

By default, when you run `Set-PSSessionConfiguration``, you are prompted to restart the WinRM service to make the new session configuration effective. Until the WinRM service is restarted, the new session configuration is not effective.

To restart the WinRM service without prompting, use the Force parameter. To restart the WinRM service manually, use the `Restart-Service`` cmdlet.

`-Path <System.String>`

Specifies the path of a session configuration file (`.pssc``), such as one created by the `New-PSSessionConfigurationFile`` cmdlet. If you omit the path, the default is the current directory.

For information about how to modify a session configuration file, see the help topic for the `New-PSSessionConfigurationFile`` cmdlet.

This parameter was introduced in PowerShell 3.0.

`-PSVersion <System.Version>`

Specifies the version of PowerShell in sessions that use this session configuration.

The value of this parameter takes precedence over the value of the `PowerShellVersion` key in the session configuration file.

This parameter was introduced in PowerShell 3.0.

`-RunAsCredential <System.Management.Automation.PSCredential>`

Specifies credentials for commands in the session. By default, commands run with the permissions of the current user.

This parameter was introduced in PowerShell 3.0.

`-SecurityDescriptorSddl <System.String>`

Specifies a different Security Descriptor Definition Language (SDDL) string for the configuration.

This string determines the permissions that are required to use the new session configuration. To use a session configuration in a session, users must have at least ``Execute(Invoke)`` permission for the configuration.

To use the default security descriptor for the configuration, enter an empty string (`""`) or a value of ``$Null``. The default is the root SDDL in the ``WSMan:`` drive.

If the security descriptor is complex, consider using the `ShowSecurityDescriptorUI` parameter instead of this one. You cannot use both parameters in the same command.

`-SessionTypeOption <System.Management.Automation.PSSessionTypeOption>`

Specifies type-specific options for the session configuration. Enter a session type options object, such as the `PSWorkflowExecutionOption` object that the ``New-PSWorkflowExecutionOption`` cmdlet returns.

The options of sessions that use the session configuration are determined by the values of session options and the session configuration options.

Unless specified, options set in the session, such as with the ``New-PSSessionOption`` cmdlet, take precedence over options set in the session configuration. However, session option values cannot exceed



maximum values set in the session configuration.

This parameter was introduced in PowerShell 3.0.

#### `-ShowSecurityDescriptorUI <System.Management.Automation.SwitchParameter>`

When this parameter is used the cmdlet opens a property sheet that helps you create a new SDDL for the session configuration. The property sheet appears after you run the ``Set-PSSessionConfiguration`` command and then restart the WinRM service.

When you set permissions to the configuration, remember that users must have at least ``Execute(Invoke)`` permission to use the session configuration in a session.

You cannot use the `SecurityDescriptorSDDL` parameter and this parameter in the same command.

#### `-StartupScript <System.String>`

Specifies the startup script for the configuration. Enter the fully qualified path of a PowerShell script. The specified script runs in the new session that uses the session configuration.

To delete a startup script from a session configuration, enter an empty string (``""``) or a value of ``$Null``.

You can use a startup script to further configure the user session. If the script generates an error, even a non-terminating error, the session is not created and the ``New-PSSession`` command fails.

#### `-ThreadApartmentState <System.Threading.ApartmentState>`

Specifies the apartment state of the threading module to be used.

Acceptable values are:

- `Unknown`

- `MTA`

- `STA`

-ThreadOptions <System.Management.Automation.Runspaces.PSThreadOptions>

Specifies the thread options setting in the configuration. This setting defines how threads are created and used when a command is executed in the session. The acceptable values for this parameter are:

- `Default`

- `ReuseThread`

- `UseCurrentThread`

- `UseNewThread`

The default value is `UseCurrentThread`.

For more information, see [PSThreadOptions Enumeration](#)

([/dotnet/api/system.management.automation.runspaces.psthreadoptions](#)).

-TransportOption <System.Management.Automation.PSTransportOption>

Specifies the transport options for the session configuration. Enter a transport options object, such as the [WSManConfigurationOption](#) object that the `New-PSTransportOption` cmdlet returns.

The options of sessions that use the session configuration are determined by the values of session options and the session configuration options.

Unless specified, options set in the session, such as with the

`New-PSSessionOption` cmdlet, take precedence over options set in the session configuration. However, session option values cannot exceed maximum values set in the session configuration.

This parameter was introduced in PowerShell 3.0.

`-UseSharedProcess <System.Management.Automation.SwitchParameter>`

Use only one process to host all sessions that are started by the same user and use the same session configuration. By default, each session is hosted in its own process.

This parameter was introduced in PowerShell 3.0.

`-Confirm <System.Management.Automation.SwitchParameter>`

Prompts you for confirmation before running the cmdlet.

`-WhatIf <System.Management.Automation.SwitchParameter>`

Shows what would happen if the cmdlet runs. The cmdlet is not run.

`<CommonParameters>`

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see `about_CommonParameters` (<https://go.microsoft.com/fwlink/?LinkID=113216>).

----- Example 1: Change the thread apartment state -----

```
PS C:\> Set-PSSessionConfiguration -Name "MaintenanceShell"  
-ThreadApartmentState STA
```

This command changes the thread apartment state in the MaintenanceShell configuration to STA. The change is effective when you restart the WinRM service.

----- Example 2: Create and change a session configuration -----

```
Register-PSSessionConfiguration -Name "AdminShell" -AssemblyName  
"C:\Shells\AdminShell.dll" -ConfigurationTypeName "AdminClass"  
Set-PSSessionConfiguration -Name "AdminShell" -StartupScript "AdminConfig.ps1"  
Set-PSSessionConfiguration -Name "AdminShell" -StartupScript $Null
```

----- Example 3: Display results -----

```
Set-PSSessionConfiguration -Name "IncObj" -MaximumReceivedObjectSizeMB 20
```

```
WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Plugin\IncObj\Initiali  
zationParameters
```

ParamName	ParamValue
-----------	------------

-----	-----
psmaximumreceivedobjectsizemb	20

```
"Restart WinRM service"
```

```
WinRM service need to be restarted to make the changes effective. Do you want  
to run the command "restart-service winrm"?
```

```
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y
```

----- Example 3: Display results in different ways -----

```
Set-PSSessionConfiguration -Name "MaintenanceShell" -StartupScript  
"C:\ps-test\Maintenance.ps1"
```

```
WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Plugin\MaintenanceShel  
\InitializationParameters
```

ParamName      ParamValue

-----      -----

startupscript    c:\ps-test\Mainte...

"Restart WinRM service"

WinRM service need to be restarted to make the changes effective. Do you want  
to run

the command "restart-service winrm"?

[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y

Get-PSSessionConfiguration MaintenanceShell | Format-List -Property \*

xmlns            :

http://schemas.microsoft.com/wbem/wsman/1/config/PluginConfiguration

Name            : MaintenanceShell

Filename        : %windir%\system32\pwrshplugin.dll

SDKVersion      : 1

XmlRenderingType : text

lang            : en-US

PSVersion       : 2.0

startupscript   : c:\ps-test\Maintenance.ps1

ResourceUri     : http://schemas.microsoft.com/powershell/MaintenanceShell

SupportsOptions : true

ExactMatch      : true

Capability       : {Shell}

Permission       :

Get-ChildItem WSMAN:\localhost\Plugin\MaintenanceShell\InitializationParameters

ParamName      ParamValue

-----      -----

PSVersion      2.0

startupscript   c:\ps-test\Maintenance.ps1

## REMARKS

To see the examples, type: "get-help Set-PSSessionConfiguration -examples".

For more information, type: "get-help Set-PSSessionConfiguration -detailed".

For technical information, type: "get-help Set-PSSessionConfiguration -full".

For online help, type: "get-help Set-PSSessionConfiguration -online"