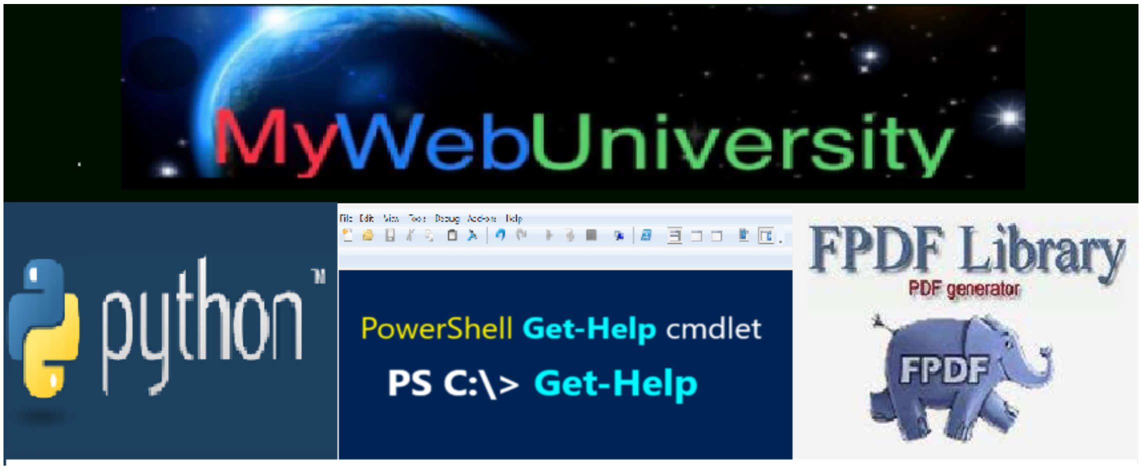

Full credit is given to all the above companies including the Operating System that this PDF file was generated!

Windows PowerShell Get-Help on Cmdlet 'New-WSManSessionOption'

PS:\>Get-HELP New-WSManSessionOption -Full

NAME

New-WSManSessionOption

SYNOPSIS

Creates session option hash table to use as input parameters for WS-Management cmdlets.

SYNTAX

New-WSManSessionOption [-NoEncryption] [-OperationTimeout <System.Int32>] [-ProxyAccessType {ProxyIEConfig | ProxyWinHttpConfig | ProxyAutoDetect |

ProxyNoProxyServer}] [-ProxyAuthentication {Negotiate | Basic | Digest}] [-ProxyCredential <System.Management.Automation.PSCredential>] [-SkipCACheck] [-SkipCNCheck]

[-SkipRevocationCheck] [-SPNPort <System.Int32>] [-UseUTF16] [<CommonParameters>]

DESCRIPTION

The `New-WSManSessionOption` cmdlet creates a WSMan Session option hash table which can be passed to WSMan cmdlets:

- `Get-WSManInstance`

- `Set-WSManInstance`

- `Invoke-WSManAction`

- `Connect-WSMan`

PARAMETERS

-NoEncryption <System.Management.Automation.SwitchParameter>

Indicates that the connection does not use encryption for remote operations over HTTP.

By default, unencrypted traffic is not enabled. It must be enabled in the local configuration.

Required?   false

Position?   named

Default value   False

Accept pipeline input?   False

Accept wildcard characters? false

-OperationTimeout <System.Int32>

Specifies the time-out, in milliseconds, for the WS-Management operation.

Required?   false

Position?   named

Default value   None

Accept pipeline input?   False

Accept wildcard characters? false

-ProxyAccessType <Microsoft.WSMan.Management.ProxyAccessType>

Specifies the mechanism by which the proxy server is located. The acceptable values for this parameter are:

- `ProxyIEConfig` - Use the Internet Explorer proxy configuration for the current user.

- `ProxyWinHttpConfig` - The WSMan client uses the proxy settings configured for WinHTTP, using the

ProxyCfg.exe utility. - `ProxyAutoDetect` - Force auto-detection of a proxy server.

- `ProxyNoProxyServer` - Do not use a proxy server. Resolve all host names locally.

The default value is ProxyIEConfig.

Required?   false

Position?   named

Default value   None

Accept pipeline input?   False

Accept wildcard characters? false

-ProxyAuthentication <Microsoft.WSMan.Management.ProxyAuthentication>

Specifies the authentication method to use at the proxy. The acceptable values for this parameter are:

- `Basic` - Basic is a scheme in which the user name and password are sent in clear-text to the server or proxy. - `Digest` - Digest is a challenge-response

scheme that uses a server-specified data string for the challenge. - `Negotiate` - Negotiate is a challenge-response scheme that negotiates with the server or

proxy to determine which scheme to use for authentication. Examples are the Kerberos protocol and NTLM.

The default value is Negotiate.

Required?   false

Position?   named

Default value   None

Accept pipeline input?   False

Accept wildcard characters? false

-ProxyCredential <System.Management.Automation.PSCredential>

Specifies a user account that has permission to gain access through an intermediate Web proxy.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SkipCACheck <System.Management.Automation.SwitchParameter>

Specifies that, when it connects over HTTPS, the client does not validate that the server certificate is signed by a trusted certification authority (CA). Use this option only when the remote computer is trusted by another method, for example, if the remote computer is part of a network that is physically secure and isolated or the remote computer is listed as a trusted host in the WS-Management configuration.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-SkipCNCheck <System.Management.Automation.SwitchParameter>

Specifies that the certificate common name (CN) of the server does not have to match the host name of the server. This is used only in remote operations using HTTPS. This option should only be used for trusted computers.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-SkipRevocationCheck <System.Management.Automation.SwitchParameter>

Indicates that the connection does not validate the revocation status on the server certificate.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-SPNPort <System.Int32>

Specifies a port number to append to the connection Service Principal Name (SPN) of the remote server. An SPN is used when the authentication mechanism is

Kerberos or Negotiate.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-UseUTF16 <System.Management.Automation.SwitchParameter>

Indicates that the connection encodes the request in UTF16 format instead of UTF8 format. The default is UTF8 encoding.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

<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).

INPUTS

OUTPUTS

Microsoft.WSMan.Management.SessionOption

This cmdlet returns the created WSMan session option as a SessionOption object.

NOTES

- Example 1: Create a connection that uses connection options -

```
PS C:\> $a = New-WSManSessionOption -OperationTimeout 30000

PS C:\> Connect-WSMan -ComputerName "server01" -SessionOption $a

PS C:\> cd wsman:

PS WSMan:\> dir


WSManConfig: Microsoft.WSMan.Management\WSMan::WSMan

ComputerName                           Type

------------                           ----

localhost                              Container

server01                               Container
```

This example creates a connection to the remote server01 computer by using the connection options that are defined by `New-WSManSessionOption`.

The first command uses `New-WSManSessionOption` to store a set of connection setting options in the `$a` variable. In this case, the session options set a connection time out of 30 seconds (30,000 milliseconds).

The second command uses the SessionOption parameter to pass the credentials that are stored in the `$a` variable to `Connect-WSMan`. Then, `Connect-WSMan` connects to

the remote server01 computer by using the specified session options.

`Connect-WSMan` is generally used in the context of the WSMan provider to connect to a remote computer, in this case the server01 computer. However, you can use the

cmdlet to establish connections to remote computers before you change to the WSMan provider. Those connections appear in the ComputerName list.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/microsoft.wsman.management/new-wsmansessionoption?view=powershell-5.1&WT.mc_id=ps-gethelp

Connect-WSMan

Disable-WSManCredSSP

Disconnect-WSMan

Enable-WSManCredSSP

Get-WSManCredSSP

Get-WSManInstance

Invoke-WSManAction

New-WSManInstance

Remove-WSManInstance

Set-WSManInstance

Set-WSManQuickConfig

Test-WSMan