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

PowerShell Get-Help on command 'Set-WSManInstance'

PS C:\Users\wahid> Get-Help Set-WSManInstance

NAME

Set-WSManInstance

SYNOPSIS

Modifies the management information that is related to a resource.

SYNTAX

Set-WSManInstance [-ResourceURI] <System.Uri> [[-SelectorSet]
<System.Collections.Hashtable>] [-ApplicationName <System.String>]
[-Authentication {None | Default | Digest | Negotiate | Basic | Kerberos |
ClientCertificate | Credssp}] [-CertificateThumbprint <System.String>]
[-ComputerName <System.String>] [-Credential
<System.Management.Automation.PSCredential>] [-Dialect <System.Uri>]
[-FilePath <System.String>] [-Fragment <System.String>] [-OptionSet
<System.Collections.Hashtable>] [-Port <System.Int32>] [-SessionOption
<Microsoft.WSMan.Management.SessionOption>] [-UseSSL] [-ValueSet
<System.Collections.Hashtable>] [<CommonParameters>]

Set-WSManInstance [-ResourceURI] <System.Uri> [[-SelectorSet]
<System.Collections.Hashtable>] [-Authentication {None | Default | Digest |

Negotiate | Basic | Kerberos | ClientCertificate | Credssp}]
[-CertificateThumbprint <System.String>] [-ConnectionURI <System.Uri>]
[-Credential <System.Management.Automation.PSCredential>] [-Dialect
<System.Uri>] [-FilePath <System.String>] [-Fragment <System.String>]
[-OptionSet <System.Collections.Hashtable>] [-SessionOption
<Microsoft.WSMan.Management.SessionOption>] [-ValueSet
<System.Collections.Hashtable>] [<CommonParameters>]

DESCRIPTION

The `Set-WSManInstance` cmdlet modifies the management information that is related to a resource.

This cmdlet uses the WinRM connection/transport layer to modify the information.

PARAMETERS

-ApplicationName <System.String>

Specifies the application name in the connection. The default value of the ApplicationName parameter is "WSMAN". The complete identifier for the remote endpoint is in the following format:

`<transport>://<server>:<port>/<ApplicationName>`

For example:

`http://server01:8080/WSMAN`

Internet Information Services (IIS), which hosts the session, forwards requests with this endpoint to the specified application. This default setting of `WSMAN` is appropriate for most uses. This parameter is designed to be used when numerous computers establish remote connections

to one computer that is running Windows PowerShell. In this case, IIS hosts Web Services for Management (WS-Management) for efficiency.

-Authentication <Microsoft.WSMan.Management.AuthenticationMechanism>
Specifies the authentication mechanism to be used at the server. Possible values are:

- `Basic`: Basic is a scheme in which the user name and password are sent in clear text to the server or proxy. - `Default` : Use the authentication method implemented by the WS-Management protocol. This is the default. - `Digest`: Digest is a challenge-response scheme that uses a server-specified data string for the challenge. - `Kerberos`: The client computer and the server mutually authenticate by using Kerberos certificates. - `Negotiate`: Negotiate is a challenge-response scheme that negotiates with the server or proxy to determine the scheme to use for authentication. For example, this parameter value allows negotiation to determine whether the Kerberos protocol or NTLM is used. - `CredSSP`: Use Credential Security Support Provider (CredSSP) authentication, which allows the user to delegate credentials. This option is designed for commands that run on one remote computer but collect data from or run additional commands on other remote computers.

> [!CAUTION] > CredSSP delegates the user's credentials from the local computer to a remote computer. This > practice increases the security risk of the remote operation. If the remote computer is > compromised, when credentials are passed to it, the credentials can be used to control the network > session.

-CertificateThumbprint <System.String>
Specifies the digital public key certificate (X509) of a user account that has permission to perform this action. Enter the certificate thumbprint of the certificate.

Certificates are used in client certificate-based authentication. They can be mapped only to local user accounts; they do not work with domain accounts.

To get a certificate thumbprint, use the `Get-Item` or `Get-ChildItem` command in the PowerShell `Cert:` drive.

-ComputerName <System.String>

Specifies the computer against which you want to run the management operation. The value can be a fully qualified domain name, a NetBIOS name, or an IP address. Use the local computer name, `localhost`, or a dot (`.`) to specify the local computer. The local computer is the default.

When the remote computer is in a different domain from the user, you must use a fully qualified domain name. You can pipe a value for this parameter to the cmdlet.

-ConnectionURI <System.Uri>

Specifies the connection endpoint. The format of this string is:

`<Transport>://<Server>:<Port>/<ApplicationName>`

The following string is a properly formatted value for this parameter:

`http://Server01:8080/WSMAN`

The URI must be fully qualified.

-Credential <System.Management.Automation.PSCredential>

Specifies a user account that has permission to perform this action. The default is the current user. Type a user name, such as `User01`, `Domain01\User01`, or `User@Domain.com`. Alternatively, enter a PSCredential object, such as one returned by the `Get-Credential` cmdlet.

When you type a user name, you will be prompted for a password.

-Dialect <System.Uri>

Specifies the dialect to use in the filter predicate. This can be any dialect that is supported by the remote service. The following aliases can be used for the dialect URI:

- `WQL`: `http://schemas.microsoft.com/wbem/wsman/1/WQL`

- `Selector`: `http://schemas.microsoft.com/wbem/wsman/1/wsman/SelectorFilter`

- `Association`: `http://schemas.dmtf.org/wbem/wsman/1/cimbinding/associationFilter`

-FilePath <System.String>

Specifies the path of a file that is used to update a management resource. You specify the management resource by using the ResourceURI parameter and the SelectorSet parameter. For example, the following command uses the FilePath parameter:

`Invoke-WSManAction -Action StopService -ResourceUri wmicimv2/Win32_Service -SelectorSet @{Name="spooler"} -FilePath:c:\input.xml -authentication default`

This command calls the StopService method on the Spooler service by using input from a file. The file, `Input.xml`, contains the following content:

`<p:StopService_INPUT xmlns:p="http://schemas.microsoft.com/wbem/wsman/1/wmi/root/cimv2/Win32_Service" />`

-Fragment <System.String>

Specifies a section inside the instance that is to be updated or retrieved

for the specified operation. For example, to get the status of a spooler service, specify `-Fragment Status`.

-OptionSet <System.Collections.Hashtable>

Passes a set of switches to a service to modify or refine the nature of the request. These are similar to switches used in command-line shells because they are service specific. Any number of options can be specified.

The following example demonstrates the syntax that passes the values `1`, `2`, and `3` for the `a`, `b`, and `c` parameters:

`-OptionSet @{a=1;b=2;c=3}`

-Port <System.Int32>

Specifies the port to use when the client connects to the WinRM service. When the transport is HTTP, the default port is 80. When the transport is HTTPS, the default port is 443.

When you use HTTPS as the transport, the value of the ComputerName parameter must match the server's certificate common name (CN). However, if the SkipCNCheck parameter is specified as part of the SessionOption parameter, then the certificate common name of the server does not have to match the host name of the server. The SkipCNCheck parameter should be used only for trusted machines.

-ResourceURI <System.Uri>

Contains the Uniform Resource Identifier (URI) of the resource class or instance. The URI is used to identify a specific type of resource, such as disks or processes, on a computer.

A URI consists of a prefix and a path to a resource. For example:

`http://schemas.microsoft.com/wbem/wsman/1/wmi/root/cimv2/Win32_LogicalDisk

`

`http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_NumericSensor`

-SelectorSet <System.Collections.Hashtable>
Specifies a set of value pairs that are used to select particular management resource instances. The SelectorSet parameter is used when more than one instance of the resource exists. The value of the SelectorSet parameter must be a hash table. The following example shows how to enter a value for this parameter:

`-SelectorSet @{Name="WinRM";ID="yyy"}`

-SessionOption <Microsoft.WSMan.Management.SessionOption>
Defines a set of extended options for the WS-Management session. Enter a SessionOption object that you create with the `New-WSManSessionOption` cmdlet. For more information about the options that are available, see New-WSManSessionOption (New-WSManSessionOption.md).

-UseSSL <System.Management.Automation.SwitchParameter>
Specifies that the Secure Sockets Layer (SSL) protocol should be used to establish a connection to the remote computer. By default, SSL is not used.

WS-Management encrypts all the Windows PowerShell content that is transmitted over the network. The UseSSL parameter lets you specify the additional protection of HTTPS instead of HTTP. If SSL is not available on the port that is used for the connection and you specify this parameter, the command fails.

-ValueSet <System.Collections.Hashtable>
Specifies a hash table that helps modify a management resource. You specify the management resource by using the ResourceURI parameter and the SelectorSet parameter. The value of the ValueSet parameter must be a hash

table.

<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: Disable a listener on the local computer -----

Set-WSManInstance -ResourceURI winrm/config/listener -SelectorSet

@{address="*";transport="https"} -ValueSet @{Enabled="false"}

cfg :

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

xsi : http://www.w3.org/2001/XMLSchema-instance

lang : en-US

Address : *

Transport : HTTPS

Port : 443

Hostname :

Enabled : false

URLPrefix : wsman

CertificateThumbprint :

ListeningOn : {127.0.0.1, 172.30.168.171, ::1,

2001:4898:0:fff:0:5efe:172.30.168.171...}

This command disables the HTTPS listener on the local computer.

> [!IMPORTANT] > The ValueSet parameter is case-sensitive when matching the

properties specified.

For example, in this command,

This fails: `-ValueSet @{enabled="False"}`

This succeeds: `-ValueSet @{Enabled="False"}`

Example 2: Set the maximum envelope size on the local computer

Set-WSManInstance -ResourceURI winrm/config -ValueSet @{MaxEnvelopeSizekb =
"200"}

cfg : http://schemas.microsoft.com/wbem/wsman/1/config
lang : en-US
MaxEnvelopeSizekb : 200
MaxTimeoutms : 60000
MaxBatchItems : 32000
MaxProviderRequests : 4294967295
Client : Client
Service : Service
Winrs : Winrs

This command sets the MaxEnvelopeSizekb value to 200 on the local computer.

> [!IMPORTANT] > The ValueSet parameter is case-sensitive when matching the
properties specified.

For example, using the above command.

This fails: `-ValueSet @{MaxEnvelopeSizeKB ="200"}`

This succeeds: `-ValueSet @{MaxEnvelopeSizekb ="200"}`

------ Example 3: Disable a listener on a remote computer ------

Set-WSManInstance -ResourceURI winrm/config/listener -ComputerName SERVER02
-SelectorSet @{address="*";transport="https"} -ValueSet @{Enabled="false"}

cfg : http://schemas.microsoft.com/wbem/wsman/1/config/listener

xsi : http://www.w3.org/2001/XMLSchema-instance

lang : en-US

Address : *

Transport : HTTPS

Port : 443

Hostname :

Enabled : false

URLPrefix : wsman

CertificateThumbprint :

ListeningOn : {127.0.0.1, 172.30.168.172, ::1, 2001:4898:0:fff:0:5efe:172.30.168.172...}

This command disables the HTTPS listener on the remote computer SERVER02.

> [!IMPORTANT] > The ValueSet parameter is case-sensitive when matching the properties specified.

For example, using the above command.

This fails: `-ValueSet @{enabled="False"}`

This succeeds: `-ValueSet @{Enabled="False"}`

REMARKS

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

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

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

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