Windows PowerShell Get-Help on Cmdlet 'Set-NetIPsecPhase2AuthSet'

PS:\>Get-HELP Set-NetIPsecPhase2AuthSet -Full

NAME

Set-NetIPsecPhase2AuthSet

SYNOPSIS

Modifies existing phase 2 authentication sets.

SYNTAX

Set-NetIPsecPhase2AuthSet [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-Description <String>] -DisplayGroup <String[]> [-GPOSession <String>] [-NewDisplayName

<String>] [-PassThru] [-PolicyStore <String>] [-Proposal <CimInstance[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

Set-NetIPsecPhase2AuthSet [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-Description <String>] -DisplayName <String[]> [-GPOSession <String>] [-NewDisplayName

<String>] [-PassThru] [-PolicyStore <String>] [-Proposal <CimInstance[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

Set-NetIPsecPhase2AuthSet [-Name] <String[]> [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-Description <String>] [-GPOSession <String>] [-NewDisplayName

<String>] [-PassThru] [-PolicyStore <String>] [-Proposal <CimInstance[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

Set-NetIPsecPhase2AuthSet [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-Description <String>] [-GPOSession <String>] -Group <String[]> [-NewDisplayName <String>]
[-PassThru] [-PolicyStore <String>] [-Proposal <CimInstance[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

Set-NetIPsecPhase2AuthSet [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-Description <String>] -InputObject <CimInstance[]> [-NewDisplayName <String>] [-PassThru]
[-Proposal <CimInstance[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The Set-NetIPsecPhase2AuthSet cmdlet modifies cryptographic set properties of existing main mode cryptographic sets.

This cmdlet gets one or more phase 2 authentication sets to be modified with the Name parameter (default), DisplayName parameter, or by group association using the
Group or DisplayGroup parameter. The Get-NetIPsecPhase2AuthSet cmdlet returns the sets queried by property and pipes the sets into this cmdlet. The remaining
parameters specify the properties of the set to be modified. When a group is specified, all of the sets associated with the group receive the same modifications. The
rule parameters modified using the dot-notation are committed with this cmdlet.

To move a set to a new GPO, copy the existing set by running the Copy-NetIPsecPhase2AuthSet cmdlet with the NewPolicyStore parameter, then remove the old set with the
Remove-NetIPsecPhase2AuthSet cmdlet.

PARAMETERS

-AsJob [<SwitchParameter>]

Runs the cmdlet as a background job. Use this parameter to run commands that take a long time to complete.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-CimSession <CimSession[]>

Runs the cmdlet in a remote session or on a remote computer. Enter a computer name or a session object, such as the output of a New-CimSession

(https://go.microsoft.com/fwlink/p/?LinkId=227967) or [Get-CimSession](https://go.microsoft.com/fwlink/p/?LinkId=227966)cmdlet. The default is the current session

on the local computer.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Confirm [<SwitchParameter>]

Prompts you for confirmation before running the cmdlet.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Description <String>

Specifies that matching IPsec rules of the indicated description are modified. Wildcard characters are accepted. This parameter provides information about the

firewall rule. This parameter specifies the localized, user-facing description of the IPsec rule.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-DisplayGroup <String[]>

Specifies that only matching phase 2 authentication sets of the indicated group association are modified. Wildcard characters are accepted. The Group parameter

specifies the source string for this parameter. If the value for this parameter is a localizable string, then the Group parameter contains an indirect string.

Rule groups can be used to organize rules by influence and allows batch rule modifications. Using the Set-NetIPsecPhase2AuthSet cmdlet, if the group name is

specified for a set of rules, then all of the rules in that group receive the same set of modifications. It is a good practice to specify the Group parameter with

a universal and world-ready indirect @FirewallAPI name. This parameter cannot be specified upon object creation using the New-NetIPsecPhase2AuthSet cmdlet, but

can be modified using dot notation and the Set-NetIPsecPhase2AuthSet cmdlet.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-DisplayName <String[]>

Specifies that only matching firewall rules of the indicated display name are modified. Wildcard characters are accepted. Specifies the localized, user-facing

name of the firewall rule being created. When creating a rule this parameter is required. This parameter value is locale-dependent. If the object is not modified,

this parameter value may change in certain circumstances. When writing scripts in multi-lingual environments, the Name parameter should be used instead, where the

default value is a randomly assigned value. This parameter cannot be set to All.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-GPOSession <String>

Specifies the network GPO from which to retrieve the rules to be modified. This parameter is used in the same way as the PolicyStore parameter. When modifying

GPOs in Windows PowerShellr, each change to a GPO requires the entire GPO to be loaded, modified, and saved back. On a busy Domain Controller (DC), this can be a

slow and resource-heavy operation. A GPO Session loads a domain GPO onto the local computer and makes all changes in a batch, before saving it back. This reduces

the load on the DC and speeds up the Windows PowerShell cmdlets. To load a GPO Session, use the Open-NetGPO cmdlet. To save a GPO Session, use the Save-NetGPO

cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Group <String[]>

Specifies that only matching firewall rules of the indicated group association are modified. Wildcard characters are accepted. This parameter specifies the

source string for the DisplayGroup parameter. If the DisplayGroup parameter value is a localizable string, then this parameter contains an indirect string. Rule

groups can be used to organize rules by influence and allows batch rule modifications. Using the Set-NetIPsecRule cmdlets, if the group name is specified for a

set of rules or sets, then all of the rules or sets in that group receive the same set of modifications. It is good practice to specify this parameter value with

a universal and world-ready indirect @FirewallAPI name. The DisplayGroup parameter cannot be specified upon

object creation using the New-NetIPsecRule cmdlet,

but can be modified using dot-notation and the Set-NetIPsecRule cmdlet.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-InputObject <CimInstance[]>

Specifies the input object that is used in a pipeline command.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-Name <String[]>

Specifies that only matching main mode cryptographic sets of the indicated name are modified. Wildcard characters are accepted. This parameter acts just like a

file name, in that only one rule with a given name may exist in a policy store at a time. During group policy processing and policy merge, rules that have the

same name but come from multiple stores being merged, will overwrite one another so that only one exists. This overwriting behavior is desirable if the rules

serve the same purpose. For instance, all of the firewall rules have specific names, so if an administrator can copy these rules to a GPO, and the rules will

override the local versions on a local computer. GPOs can have precedence. So, if an administrator has a different or more specific rule the same name in a

higher-precedence GPO, then it overrides other rules that exist. The default value is a randomly assigned value. When you want to override the defaults for main

mode encryption, specify the customized parameters and set this parameter value, making this parameter the new default setting for encryption.

Required? true

Position? 0

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-NewDisplayName <String>

Specifies the new display name for an IPsec rule.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PassThru [<SwitchParameter>]

Returns an object representing the item with which you are working. By default, this cmdlet does not generate any output.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-PolicyStore <String>

Specifies the policy store from which to retrieve the sets to be modified. A policy store is a container for firewall and IPsec policy. The acceptable values

for this parameter are:

- PersistentStore: Sometimes called static rules, this store contains the persistent policy for the local computer. This policy is not from GPOs, and has been

created manually or programmatically, during application installation, on the computer. Rules created in this store are attached to the ActiveStore and activated

on the computer immediately. - ActiveStore: This store contains the currently active policy, which is the sum of all policy stores that apply to the computer.

This is the resultant set of policy (RSOP) for the local computer (the sum of all GPOs that apply to the computer), and the local stores (the PersistentStore, the

Static Windows Service Hardening (WSH), and the Configurable WSH). ---- GPOs are also policy stores. Computer GPOs can be specified as follows. ------

`-PolicyStore hostname`.

---- Active Directory GPOs can be specified as follows.

------ `-PolicyStore domain.fqdn.com\GPO_Friendly_Namedomain.fqdn.comGPO_Friendly_Name`.

------ Such as the following.

-------- `-PolicyStore localhost`

-------- `-PolicyStore corp.contoso.com\FirewallPolicy`

---- Active Directory GPOs can be created using the New-GPO cmdlet or the Group Policy Management Console. - RSOP: This read-only store contains the sum of all

GPOs applied to the local computer.

- SystemDefaults: This read-only store contains the default state of firewall rules that ship with Windows Serverr 2012.

- StaticServiceStore: This read-only store contains all the service restrictions that ship with Windows Server 2012.

Optional and product-dependent features are considered part of Windows Server 2012 for the purposes of WFAS. - ConfigurableServiceStore: This read-write store

contains all the service restrictions that are added for third-party services. In addition, network isolation rules that are created for Windows Store application

containers will appear in this policy store. The default value is PersistentStore. This cmdlet cannot be used to add an

object to a policy store. An object can

only be added to a policy store at creation time with the Copy-NetIPsecPhase2AuthSet cmdlet or with the New-NetIPsecPhase2AuthSet cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Proposal <CimInstance[]>

Associates the specified cryptographic proposal to the corresponding cryptographic set to be used in main mode negotiations. Separate multiple entries with a

comma.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ThrottleLimit <Int32>

Specifies the maximum number of concurrent operations that can be established to run the cmdlet. If this parameter is omitted or a value of `0` is entered, then

Windows PowerShellr calculates an optimum throttle limit for the cmdlet based on the number of CIM cmdlets that are running on the computer. The throttle limit

applies only to the current cmdlet, not to the session or to the computer.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-WhatIf [<SwitchParameter>]

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

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

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\NetIPsecMainModePhase2AuthSet[]

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

OUTPUTS

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\NetIPsecMainModePhase2AuthSet[]

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

NOTES

---------- EXAMPLE 1 ----------

PS C:\>$NewCertProposal = New-NetIPsecAuthProposal -User -Cert -Authority "C=US,O=MSFT,CN=Microsoft Root Authority" -AuthorityType Root

PS C:\>Set-NetIPsecPhase2AuthSet -DisplayName "User Certificate Auth Set" -Proposal $NewCertProposal

This example replaces the proposals of an existing authentication set.

---------- EXAMPLE 2 ----------

PS C:\>Set-NetIPsecPhase2AuthSet -DisplayGroup "Authenticate with Certificates" -NewDisplayName "User Authentication Certificates"

This example modifies the display of a phase 1 authentication set.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/netsecurity/set-netipsecphase2authset?view=windowsserver2022-ps&wt.mc_id=ps-gethelp

Copy-NetIPsecPhase2AuthSet

Get-NetIPsecMainModeCryptoSet

Get-NetIPsecPhase2AuthSet

New-NetIPsecMainModeCryptoSet

New-NetIPsecRule

Open-NetGPO

Remove-NetIPsecPhase2AuthSet

Save-NetGPO

Set-NetIPsecMainModeCryptoSet

Set-NetIPsecRule