Windows PowerShell Get-Help on Cmdlet 'Update-NetIPsecRule'

PS:\>Get-HELP Update-NetIPsecRule -Full

NAME

Update-NetIPsecRule

SYNOPSIS

Updates an IPsec rule by adding or removing a set of IP addresses.

SYNTAX

Update-NetIPsecRule -Action {Add | Delete} [-AsJob] [-CimSession <CimSession[]>] [-Confirm] -EndpointType {Endpoint1 | Endpoint2} [-GPOSession <String>]

-IPsecRuleName <String[]> [-IPv4Addresses <String[]>] [-IPv6Addresses <String[]>] [-PassThru] [-PolicyStore <String>] [-ThrottleLimit <Int32>] [-WhatIf]

[<CommonParameters>]

Update-NetIPsecRule -Action {Add | Delete} [-AsJob] [-CimSession <CimSession[]>] [-Confirm] -EndpointType {Endpoint1 | Endpoint2} [-IPv4Addresses <String[]>]

[-IPv6Addresses <String[]>] -InputObject <CimInstance[]> [-PassThru] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The Update-NetIPsecRule cmdlet adds or deletes IP addresses for an IPsec rule.

This cmdlet can get an IPsec rule to be updated using parameter values including IPsecRuleName (default), DisplayName , rule properties, or by associated NetFirewall

filters or NetIPsec objects. The resultant queried IP addresses of the rule are updated with an Add or Delete as specified by the Action parameter.

If the Get-DAPolicyChange cmdlet is run, then the Windows PowerShellr script (.ps1) that is generated contains instances of this cmdlet. If the Windows PowerShell

script (.ps1) is run, then the IPsec rules are updated in the appropriate policy stores. See the Get-DAPolicyChange cmdlet for more information.

PARAMETERS

-Action <ChangeAction>

Specifies that the specified addresses should be added or deleted for an IPsec rule. The acceptable values for this parameter are: Add or Delete.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-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

-EndpointType <EndpointType>

Specifies that the local or remote endpoint should be modified by adding or removing IP addresses. The acceptable values for this parameter are: Endpoint1 or

Endpoint2. Endpoint1 or Endpoint2 corresponds to the local address or remote address for the IPsec rule.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-GPOSession <String>

Specifies the network GPO from which to retrieve the rules to be updated. 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

-IPsecRuleName <String[]>

Specifies that only matching IPsec rules of the indicated name are updated. 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 with the same name in a higher-precedence GPO, then it

overrides other rules that exist. The default value is a randomly assigned value. When the defaults for main mode encryption need to overridden, specify the

customized parameters and set this parameter, making it the new default setting for encryption.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-IPv4Addresses <String[]>

Specifies the list of IPv4 addresses that are updated with this cmdlet. The list contains IPv4 addresses that are being added or removed from an IPsec rule.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-IPv6Addresses <String[]>

Specifies the list of IPv6 addresses that are updated with this cmdlet. The list contains IPv6 addresses that are being added or removed from an IPsec rule.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

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

-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 rules to be updated. 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 Server 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. The Set-NetIPsecRule 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-NetIPsecRule cmdlet or with the New-NetIPsecRule cmdlet.

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\MSFT_NetConSecRule[]

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

None

NOTES

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

PS C:\>$IPv4list = 2.1.1.1,2.1.1.2

PS C:\>$IPv6list = fefe:fefe::1,fefe:fefe::2

PS C:\>Update-NetIPsecRule -IPsecRuleName "IPsec Rule" -Action Add -IPv4addresses $IPv4list -IPv6addresses $IPv6list -EndpointType Endpoint2 -PolicyStore

domain.contoso.com/sample_gpo

This example adds a list of IP addresses to an IPsec rule in a specific GPO.

RELATED LINKS

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

Get-NetFirewallAddressFilter

Get-NetFirewallInterfaceFilter

Get-NetFirewallInterfaceTypeFilter

Get-NetFirewallPortFilter

Get-NetFirewallProfile

Get-NetIPsecPhase1AuthSet

Get-NetIPsecPhase2AuthSet

Get-NetIPsecQuickModeCryptoSet

New-NetIPsecQuickModeCryptoSet

New-NetIPsecRule

Open-NetGPO

Save-NetGPO

Set-NetIPsecRule

Get-DAPolicyChange