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

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

NAME

Set-NetFirewallPortFilter

SYNOPSIS

Modifies port filter objects, thereby modifying the protocol and port conditions using the Protocol, LocalPort, RemotePort, IcmpType, and DynamicTransport parameters

of the firewall or IPsec rules.

SYNTAX

Set-NetFirewallPortFilter [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-DynamicTarget {Any | ProximityApps | ProximitySharing | WifiDirectPrinting |

WifiDirectDisplay | WifiDirectDevices}] [-GPOSession <String>] [-IcmpType <String[]>] [-LocalPort <String[]>] [-PassThru] [-PolicyStore <String>] [-Protocol <String>]

[-RemotePort <String[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

Set-NetFirewallPortFilter [-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-DynamicTarget {Any | ProximityApps | ProximitySharing | WifiDirectPrinting |

WifiDirectDisplay | WifiDirectDevices}] [-IcmpType <String[]>] -InputObject <CimInstance[]> [-LocalPort <String[]>] [-PassThru] [-Protocol <String>] [-RemotePort

<String[]>] [-ThrottleLimit <Int32>] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The Set-NetFirewallPortFilter cmdlet modifies the protocol and port conditions using the Protocol , LocalPort , RemotePort , IcmpType , and DynamicTransport

parameters associated with the input firewall or IPsec rules.

See the Get-NetFirewallPortFilter cmdlet for more information about the interface type filters.

To modify the port and protocol conditions, two methods can be used starting with the port filters returned by the Get-NetFirewallPortFilter cmdlet and optional

additional querying. - The network firewall port filter objects are piped into the Get-NetFirewallRule or Get-NetIPsecRule cmdlet. The Get-NetFirewallRule or

Get-NetIPsecRule cmdlet returns the rules associated with the filters and pipes the rules into the Set-NetFirewallRule or Set-NetIPsecRule cmdlet which configures the

interface properties. - Alternatively, the network firewall port filter objects are piped directly to this cmdlet, which modifies the Protocol , LocalPort ,

RemotePort , IcmpType , and DynamicTransport parameters values of the rules.

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

-DynamicTarget <DynamicTransport>

Specifies a dynamic transport. The cmdlet sets the protocol and port conditions for the input rules that have the dynamic transport that you specify.The acceptable values for this parameter are:

- Any

- ProximityApps

- ProximitySharing

- WifiDirectPrinting

- WifiDirectDisplay

- WifiDirectDevices

The default value is Any.

Some types of dynamic transports, such as proximity sharing, abstract the network layer details. This means that you cannot use standard network layer conditions,

such as protocols and ports, to identify the dynamic transports.

Required? false

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

-IcmpType <String[]>

Specifies the Internet Control Message Protocol (ICMP) type codes. The key encoding is specified by running the Set-NetFirewallSetting cmdlet with the KeyEncoding

parameter. The acceptable values for this parameter are:

- ICMP type code: 0 through 255

- ICMP type code pairs: 3:4

- Keyword: Any.

A rule can be queried for this condition, or modified by using the security filter object. See the Get-NetFirewallPortFilter cmdlet for more information.

Required? false
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

-LocalPort <String[]>

Specifies that network packets with matching IP port numbers match this rule. This parameter value is the first end point of an IPsec rule. The acceptable value

is a port, range, or keyword and depends on the protocol. If the Protocol parameter value is TCP or UDP, then the acceptable values for this parameter are: -

Port range: 0 through 65535.

- Port number: 80.

- Keyword: Any.

If the Protocol parameter value is ICMPv4 or ICMPv6, then the acceptable values for this parameter are: - An ICMP type, code pair: 0, 8.

- Type and code: 0 through 255.

- Keyword: Any.

If the Protocol parameter is not specified, then the acceptable values for this parameter are: Any, RPC, RPC-EPMap, or IPHTTPS. IPHTTPS is only supported on

Windows Server 2012. Querying for rules with this parameter can only be performed using filter objects. See the Get-NetFirewallPortFilter cmdlet for more

information.

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 rules 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. The Set-NetFirewallRule 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-NetFirewallRule cmdlet or with the New-NetFirewallRule cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Protocol <String>

Specifies that network packets with matching IP addresses match this rule. This parameter specifies the protocol for an IPsec rule. The acceptable values for

this parameter are:

- Protocols by number: 0 to 255.

- Protocols by name: TCP, UDP, ICMPv4, or ICMPv6.

If a port number is identified by using port1 or port2, then this parameter must be set to TCP or UDP. The values ICMPv4 and ICMPv6 create a rule that exempts

ICMP network traffic from the IPsec requirements of another rule.

The default value is Any. : Querying for rules with this parameter can only be performed using filter objects.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-RemotePort <String[]>

Specifies that network packets with matching IP port numbers match this rule. This parameter value is the second end point of an IPsec rule. The acceptable value

is a port, range, or keyword and depends on the protocol. If the protocol is TCP or UDP, then the acceptable values for this parameter are:

- Port range: 0 through 65535

- Port number: 80

- Keyword: Any

If the protocol is ICMPv4 or ICMPv6, then the acceptable values for this parameter are: - An ICMP type, code pair: 0, 8

- Type and code: 0 through 255

- Keyword: Any.

If a protocol is not specified, then the acceptable values for this parameter are: Any, RPC, RPC-EPMap, or IPHTTPS. IPHTTPS is only supported on Windows Server

2012. Querying for rules with this parameter can only be performed using filter objects. See the Get-NetFirewallPortFilter cmdlet for more information.

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_NetProtocolPortFilter[]

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\MSFT_NetProtocolPortFilter[]

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:\>$nfPortFilter = Get-FirewallRule -DisplayName "Play To streaming server" | Get-NetFirewallPortFilter

PS C:\>Set-NetFirewallPortFilter -LocalPort 10246 -InputObject $nfPortFilter

This cmdlet can be run using only the pipeline.

PS C:\>Get-FirewallRule -DisplayName "Play To streaming server" | Get-NetFirewallPortFilter | Set-NetFirewallPortFilter -LocalPort 10246

This cmdlet can be run without the pipeline.

PS C:\>Set-NetFirewallRule -DisplayName "Play To streaming server" -LocalPort 10246

This example modifies the LocalPort parameter value of the specified firewall rule.

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

PS C:\>$nfPortFilter = Get-NetFirewallPortFilter

PS C:\>$nfPortFilter10246 = Where-Object -FilterScript { $_.LocalPort -Eq "10246" } -InputObject $nfPortFilter

PS C:\>Set-NetFirewallPortFilter -LocalPort Any -InputObject $nfPortFilter10246

This cmdlet can be run using only the pipeline.

PS C:\>Get-NetFirewallPortFilter | Where-Object -FilterScript { $_.LocalPort -Eq "10246" } | Set-NetFirewallPortFilter -LocalPort Any

This example modifies all of the rules associated with a specific port.

------------------------- EXAMPLE 3 -------------------------

PS C:\>$nfPortFilter = Get-NetFirewallRule -DisplayGroup "File and Printer Sharing" | Get-NetFirewallPortFilter

PS C:\>$nfPortFilter137 = Where-Object -FilterScript { $_.RemotePort -Eq "137" } -InputObject $nfPortFilter

PS C:\>Set-NetFirewallPortFilter -LocalPort Any -InputObject $nfPortFilter137

This cmdlet can be run using only the pipeline.

PS C:\>Get-NetFirewallRule -DisplayGroup "File and Printer Sharing" | Get-NetFirewallPortFilter | Where-Object -FilterScript { $_.RemotePort -Eq "137" } |

Set-NetFirewallPortFilter -LocalPort Any

This example modifies the interface type associated with all of the firewall rules in a specified group.

RELATED LINKS

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

Where-Object https://go.microsoft.com/fwlink/p/?LinkId=113423

Copy-NetIPsecRule

Get-NetFirewallPortFilter

Get-NetFirewallRule

Get-NetIPsecRule

New-NetFirewallRule

New-NetIPsecRule

Set-NetFirewallRule

Set-NetFirewallSetting

Set-NetIPsecRule