



## ***Windows PowerShell Get-Help on Cmdlet 'Get-NetFirewallRule'***

***PS:\>Get-HELP Get-NetFirewallRule -Full***

### **NAME**

Get-NetFirewallRule

### **SYNOPSIS**

Retrieves firewall rules from the target computer.

### **SYNTAX**

```
Get-NetFirewallRule [-Action {NotConfigured | Allow | Block}] [-AsJob] [-CimSession <CimSession[]>] [-Description
<String[]>] [-Direction {Inbound | Outbound}]
[-DisplayGroup <String[]>] [-EdgeTraversalPolicy {Block | Allow | DeferToUser | DeferToApp}] [-Enabled {True | False}]
[-Group <String[]>] [-LocalOnlyMapping
<Boolean[]>] [-LooseSourceMapping <Boolean[]>] [-Owner <String[]>] [-PolicyStore <String>] [-PolicyStoreSource
<String[]>] [-PolicyStoreSourceType {None | Local |
GroupPolicy | Dynamic | Generated | Hardcoded}] [-PrimaryStatus {Unknown | OK | Inactive | Error}] [-Status <String[]>]
[-ThrottleLimit <Int32>] [-TracePolicyStore]
[<CommonParameters>]
```

```
Get-NetFirewallRule [-All] [-AsJob] [-CimSession <CimSession[]>] [-PolicyStore <String>] [-ThrottleLimit <Int32>]
[-TracePolicyStore] [<CommonParameters>]
```

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallAddressFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallApplicationFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallInterfaceFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallInterfaceTypeFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallPortFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallProfile <CimInstance> [-CimSession <CimSession[]>] [-PolicyStore  
<String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallSecurityFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

Get-NetFirewallRule [-AsJob] -AssociatedNetFirewallServiceFilter <CimInstance> [-CimSession <CimSession[]>]  
[-PolicyStore <String>] [-ThrottleLimit <Int32>]  
[-TracePolicyStore] [<CommonParameters>]

```
Get-NetFirewallRule [-AsJob] [-CimSession <CimSession[]>] -DisplayName <String[]> [-PolicyStore <String>]
[-ThrottleLimit <Int32>] [-TracePolicyStore]
[<CommonParameters>]
```

```
Get-NetFirewallRule [-Name] <String[]> [-AsJob] [-CimSession <CimSession[]>] [-PolicyStore <String>] [-ThrottleLimit
<Int32>] [-TracePolicyStore] [<CommonParameters>]
```

## DESCRIPTION

The Get-NetFirewallRule cmdlet returns the instances of firewall rules that match the search parameters from the user. See the New-NetFirewallRule cmdlet for more information.

This cmdlet returns one or more firewall rules by specifying the Name parameter (default), the DisplayName parameter, rule properties, or by associated filters or

objects. The queried rules can be placed into variables and piped to other cmdlets for further modifications or monitoring.

When running the cmdlet with the Get verb for any firewall, IPsec, or main mode rule, notice that the common conditions like addresses or ports do not appear. These

conditions are represented in separate objects called filters. The filter-to-rule relationship is always one-to-one and is managed automatically. If a query for rules

based on these parameters (ports, addresses, security, interfaces, and services) is performed, then the filter objects with the corresponding cmdlet with the Get verb

should to be retrieved. See the Get-NetFirewallAddressFilter, Get-NetFirewallApplicationFilter, Get-NetFirewallInterfaceFilter, Get-NetFirewallInterfaceTypeFilter,

Get-NetFirewallPortFilter, Get-NetFirewallProfile, Get-NetFirewallSecurityFilter, Get-NetFirewallServiceFilter, or Get-NetFirewallRule cmdlet for more information.

## PARAMETERS

-Action <Action[]>

Specifies that matching firewall rules of the indicated action are retrieved. This parameter specifies the action to take on traffic that matches this rule. The

acceptable values for this parameter are: Allow or Block.

- Allow: Network packets that match all of the criteria specified in this rule are permitted through the firewall. This is the default value. - Block: Network

packets that match all of the criteria specified in this rule are dropped by the firewall.

The default value is Allow. The OverrideBlockRules field changes an allow rule into an allow bypass rule.

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

#### -All [<SwitchParameter>]

Indicates that all of the firewall rules within the specified policy store are retrieved.

Required?	false
Position?	named
Default value	False
Accept pipeline input?	False
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

#### -AssociatedNetFirewallAddressFilter <CimInstance>

Gets the firewall rules that are associated with the given address filter to be retrieved. A NetFirewallAddressFilter object represents the address conditions

associated with a rule. See the Get-NetFirewallAddressFilter cmdlet for more information.

Required?	true
Position?	named
Default value	None
Accept pipeline input?	True (ByValue)
Accept wildcard characters?	false

-AssociatedNetFirewallApplicationFilter <CimInstance>

Gets the firewall rules that are associated with the given application filter to be retrieved. A NetFirewallApplicationFilter object represents the applications

associated with a rule. See the Get-NetFirewallApplicationFilter cmdlet for more information.

Required?	true
Position?	named
Default value	None
Accept pipeline input?	True (ByValue)
Accept wildcard characters?	false

-AssociatedNetFirewallInterfaceFilter <CimInstance>

Gets the firewall rules that are associated with the given interface filter to be retrieved. A NetFirewallInterfaceFilter object represents the interface

conditions associated with a rule. See the Get-NetFirewallInterfaceFilter cmdlet for more information.

Required?	true
Position?	named
Default value	None
Accept pipeline input?	True (ByValue)
Accept wildcard characters?	false

-AssociatedNetFirewallInterfaceTypeFilter <CimInstance>

Gets the firewall rules that are associated with the given interface type filter to be retrieved. A NetFirewallInterfaceTypeFilter object represents the

interface conditions associated with a rule. See the Get-NetFirewallInterfaceTypeFilter cmdlet for more information.

Required? true  
Position? named  
Default value None  
Accept pipeline input? True (ByValue)  
Accept wildcard characters? false

-AssociatedNetFirewallPortFilter <CimInstance>

Gets the firewall rules that are associated with the given port filter to be retrieved. A NetFirewallPortFilter object represents the port conditions associated with a rule. See the Get-NetFirewallPortFilter cmdlet for more information.

Required? true  
Position? named  
Default value None  
Accept pipeline input? True (ByValue)  
Accept wildcard characters? false

-AssociatedNetFirewallProfile <CimInstance>

Gets the firewall rules that are associated with the given firewall profile type to be retrieved. A NetFirewallProfile object represents the profile conditions associated with a rule. See the Get-NetFirewallProfile cmdlet for more information.

Required? true  
Position? named  
Default value None  
Accept pipeline input? True (ByValue)  
Accept wildcard characters? false

-AssociatedNetFirewallSecurityFilter <CimInstance>

Gets the firewall rules that are associated with the given security filter to be retrieved. A NetFirewallSecurityFilter object represents the security conditions

associated with a rule. See the Get-NetFirewallSecurityFilter cmdlet for more information. The security conditions include the Authentication , Encryption , LocalUser , RemoteUser , and RemoteMachine parameters.

Required?	true
Position?	named
Default value	None
Accept pipeline input?	True (ByValue)
Accept wildcard characters?	false

-AssociatedNetFirewallServiceFilter <CimInstance>

Gets the firewall rules that are associated with the given service filter to be retrieved. A NetFirewallServiceFilter object represents the profile conditions

associated with a rule. See the Get-NetFirewallServiceFilter cmdlet for more information.

Required?	true
Position?	named
Default value	None
Accept pipeline input?	True (ByValue)
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

#### -Description <String[]>

Specifies that matching firewall rules of the indicated description are retrieved. 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

#### -Direction <Direction[]>

Specifies that matching firewall rules of the indicated direction are retrieved. This parameter specifies which direction of traffic to match with this rule.

The acceptable values for this parameter are: Inbound or Outbound. The default value is Inbound.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

#### -DisplayGroup <String[]>

Specifies that only matching firewall rules of the indicated group association are retrieved. 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-NetFirewallRule cmdlet, 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 the Group parameter value with a

universal and world-ready indirect @FirewallAPI name. This parameter cannot be specified upon object creation using the New-NetFirewallRule cmdlet, but can be modified using dot-notation and the Set-NetFirewallRule cmdlet.

Required?	false
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 retrieved. Wildcard characters are accepted. Specifies the localized, user-facing

name of the firewall rule being retrieved. 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

-EdgeTraversalPolicy <EdgeTraversal[]>

Specifies that matching firewall rules of the indicated edge traversal policy are retrieved. This parameter specifies how this firewall rule will handle edge

traversal cases. Edge traversal allows the computer to accept unsolicited inbound packets that have passed through an edge device, such as a network address

translation (NAT) router or firewall. This option applies to inbound rules only. The acceptable values for this parameter are: Block, Allow, DeferToUser, or

DeferToApp.

- Block: Prevents applications from receiving unsolicited traffic from the Internet through a NAT edge device.

- Allow: Allows applications to receive unsolicited traffic directly from the Internet through a NAT edge device.

- DeferToUser: Allows the user to decide whether to allow unsolicited traffic from the Internet through a NAT edge device when an application requests it.

- DeferToApp: Allows each application to determine whether to allow unsolicited traffic from the Internet through a NAT edge device.

The default value is Block. The DeferToApp and DeferToUser options are only valid for computers running firstref\_client\_7, firstref\_server\_7, and Windows Server 2012.

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-Enabled <Enabled[]>

Specifies that matching firewall rules of the indicated state are retrieved. This parameter specifies that the rule object is administratively enabled or

administratively disabled. The acceptable values for this parameter are:

- True: Specifies the rule is currently enabled.

- False: Specifies the rule is currently disabled.

A disabled rule will not actively modify computer behavior, but the management construct still exists on the computer so it can be re-enabled.

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 retrieved. 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-NetFirewallRule cmdlet, 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 a 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-NetFirewallRule

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

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

#### `-LocalOnlyMapping <Boolean[]>`

Indicates that matching firewall rules of the indicated value are retrieved. This parameter specifies the firewall rules for local only mapping, which describes

whether a packet must pass through a local address on the way to the destination. Non-TCP traffic is session-less.

Windows Firewall authorizes traffic per

session, not per packet, for performance reasons. Generally, non-TCP sessions are inferred by checking the following fields: local address, remote address,

protocol, local port, and remote port. If this parameter is set to True, then the remote address and port will be ignored when inferring remote sessions.

Sessions will be grouped based on local address, protocol, and local port. This is similar to the LooseSourceMapping parameter, but performs better in cases

where the traffic does not need to be filtered by remote address. This could improve performance on heavy server workloads where UDP requests come from dynamic

client ports. For instance, Teredo relay servers.

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-LooseSourceMapping <Boolean[]>

Indicates that matching firewall rules of the indicated value are retrieved. This parameter specifies the firewall rules for loose source mapping, which

describes whether a packet can have a non-local source address when being forwarded to a destination. If this parameter is set to True, then the rule accepts

packets incoming from a host other than the one to which the packets were sent. This parameter applies only to UDP protocol traffic. The default value is False.

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-Name <String[]>

Specifies that only matching firewall rules of the indicated name are retrieved. Wildcard characters are accepted. This parameter acts just like a filename, 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 be overridden,

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

Required? true

Position? 0

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Owner <String[]>

Specifies that matching firewall rules of the indicated owner are retrieved. This parameter specifies the owner of the firewall rule, represented as an SDDL

string. All Windows Store applications that require network traffic create network isolation rules (normally through installing via the Store), where the user

that installed the application is the owner. This parameter specifies that only network packets that are authenticated as coming from or going to an owner

identified in the list of accounts (SID) match this rule.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PolicyStore <String>

Targets the policy store from which to retrieve the rules. 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 are stored in the registry under  
HKLM:\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\RestrictedServices\AppIso\FirewallR  
ules

and aren't accessible with Get-NetFirewallRule . 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 or with the New-NetFirewallRule cmdlet.

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-PolicyStoreSource <String[]>

Specifies that firewall rules matching the indicated policy store source are retrieved. This parameter contains a path to the policy store where the rule

originated if the object is retrieved from the ActiveStore with the TracePolicyStoreSource option set. This parameter value is automatically generated and should

not be modified. The monitoring output from this parameter is not completely compatible with the PolicyStore parameter. This parameter value cannot always be

passed into the PolicyStore parameter. Domain GPOs are one example in which this parameter contains only the GPO name, not the domain name.

Required?	false
Position?	named
Default value	None

Accept pipeline input? False

Accept wildcard characters? false

-PolicyStoreSourceType <PolicyStoreType[]>

Specifies that firewall rules that match the indicated policy store source type are retrieved. This parameter describes the type of policy store where the rule

originated if the object is retrieved from the ActiveStore with the TracePolicyStoreSource option set. This parameter value is automatically generated and should

not be modified. The acceptable values for this parameter are:

- Local: The object originates from the local store.

- GroupPolicy: The object originates from a GPO.

- Dynamic: The object originates from the local runtime state.

This policy store name is not valid for use in the cmdlets, but may appear when monitoring active policy. - Generated: The object was generated automatically.

This policy store name is not valid for use in the cmdlets, but may appear when monitoring active policy. - Hardcoded: The object was hard-coded. This policy store name is not valid for use in the cmdlets, but may appear when monitoring active policy.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PrimaryStatus <PrimaryStatus[]>

Specifies that firewall rules that match the indicated primary status are retrieved. This parameter specifies the overall status of the rule. - OK: Specifies

that the rule will work as specified.



- Degraded: Specifies that one or more parts of the rule will not be enforced.

- Error: Specifies that the computer is unable to use the rule at all.

Required?	false
Position?	named
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

#### -Status <String[]>

Specifies that firewall rules that match the indicated status are retrieved. This parameter describes the status message for the specified status code value. The

status code is a numerical value that indicates any syntax, parsing, or runtime errors in the rule or set. This parameter value should not be modified.

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 PowerShell 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

#### -TracePolicyStore [<SwitchParameter>]

Specifies that the name of the source GPO is queried and set to the PolicyStoreSource parameter value.

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\_NetAddressFilter

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.

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\MSFT\_NetApplicationFilter

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.

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\MSFT\_NetFirewallProfile

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.

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\MSFT\_NetInterfaceFilter

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.

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\MSFT\_NetInterfaceTypeFilter

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.

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\MSFT\_NetNetworkLayerSecurityFilter

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.

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.

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\MSFT\_NetServiceFilter

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\_NetFirewallRule

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.

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

```
PS C:\>Get-NetFirewallRule -PolicyStore ActiveStore
```

This example retrieves all of the firewall rules in the active store, which is a collection of all of the policy stores that apply to the computer. Running this

cmdlet without specifying the policy store retrieves the persistent store.

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

```
PS C:\>Get-NetFirewallProfile -Name Public | Get-NetFirewallRule
```

This example retrieves all of the firewall rules scoped to the public profile.

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

```
PS C:\>$nfServiceFilter = Get-NetFirewallRule -Group "@FirewallAPI.dll,-30502" | Get-NetFirewallServiceFilter -Service Any
```

```
PS C:\>Set-NetFirewallServiceFilter -Service Ssdpsrv -InputObject $nfServiceFilter
```

This cmdlet can be run using only the pipeline.

```
PS C:\>Get-NetFirewallRule -Group "@FirewallAPI.dll,-30502" | Get-NetFirewallServiceFilter -Service Any | Set-NetFirewallServiceFilter -Service Ssdpsrv
```

This example modifies the service associated with firewall rules in a specified group.

## RELATED LINKS

Online

Version:

[https://learn.microsoft.com/powershell/module/netsecurity/get-netfirewallrule?view=windowsserver2022-ps&wt.mc\\_id=ps-gethelp](https://learn.microsoft.com/powershell/module/netsecurity/get-netfirewallrule?view=windowsserver2022-ps&wt.mc_id=ps-gethelp)

Copy-NetFirewallRule

Get-NetFirewallAddressFilter

Get-NetFirewallApplicationFilter

Get-NetFirewallInterfaceFilter

Get-NetFirewallInterfaceTypeFilter

Get-NetFirewallPortFilter

Get-NetFirewallProfile

Get-NetFirewallRule

Get-NetFirewallSecurityFilter

Get-NetFirewallServiceFilter

New-NetFirewallRule

Open-NetGPO

Save-NetGPO

Set-NetFirewallRule

Set-NetFirewallServiceFilter