PowerShell Get-Help on command 'Get-NetFirewallApplicationFilter'

PS C:\Users\wahid> Get-Help Get-NetFirewallApplicationFilter

```
NAME
    Get-NetFirewallApplicationFilter


SYNOPSIS
    Retrieves application filter objects from the target computer.



SYNTAX
    Get-NetFirewallApplicationFilter [-All] [-AsJob] [-CimSession <CimSession[]>]
    [-GPOSession <String>] [-PolicyStore <String>] [-ThrottleLimit <Int32>]
    [<CommonParameters>]

    Get-NetFirewallApplicationFilter [-AsJob] -AssociatedNetFirewallRule
    <CimInstance> [-CimSession <CimSession[]>] [-GPOSession <String>]
    [-PolicyStore <String>] [-ThrottleLimit <Int32>] [<CommonParameters>]

    Get-NetFirewallApplicationFilter [-AsJob] [-CimSession <CimSession[]>]
    [-GPOSession <String>] [-Package <String[]>] [-PolicyStore <String>] [-Program
    <String[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]
```

DESCRIPTION

The Get-NetFirewallApplicationFilter cmdlet returns application filter objects associated with the input rules.

Application filter objects represent the applications associated with firewall rules. The Program and Package parameters of a single rule are represented in a separate NetFirewallApplicationFilter object. The filter to rule relationship is always one-to-one and is managed automatically. Rule parameters associated with filters can only be queried using filter objects.

This cmdlet displays the programs associated with firewall rules. This allows for rule querying based on the application fields using the Program or Package parameters; this cmdlet returns filter objects that may be further queried with the Where-Object (https://go.microsoft.com/fwlink/?LinkID=113423)cmdlet. The resultant filters are passed into the Get-NetFirewallRule cmdlet to return the rules queried by address.

To modify the application conditions, two methods can be used starting with the application filters returned by this cmdlet and optional additional querying.

The application filter objects can be piped into the Get-NetFirewallRule cmdlet, which returns the rule objects associated with the filters. These rules are then piped into the Set-NetFirewallRule cmdlet where the application properties can be configured.

Alternatively, piping the address filter objects directly into the Set-NetFirewallAddressFilter cmdlet allows the Program and Package parameters of the rules to be specified.

PARAMETERS

-All [<SwitchParameter>]

Indicates that all of the application filters within the specified policy store are retrieved.

-AsJob [<SwitchParameter>]

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

-AssociatedNetFirewallRule <CimInstance>

Gets the application filter object associated with the specified firewall rule to be retrieved. This parameter represents a firewall rule, which defines how traffic is filtered by the firewall. See the New-NetFirewallRule cmdlet for more information.

-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.

-GPOSession <String>

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

-Package <String[]>

Specifies the Windows Store application to which the firewall rule applies. This parameter is specified as a security identifier (SID). Querying for rules with this parameter can only be performed using filter objects.

-PolicyStore <String>

Specifies the policy store from which to retrieve the rules to be retrieved. 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 hostnamehostname`.

---- 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.

-Program <String[]>

Specifies the path and file name of the program for which the rule allows traffic. This is specified as the full path to an application file.
Querying for rules with this parameter can only be performed using filter objects.

-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.

<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 --------------------------

This example gets all of the firewall rules configured to a specified program.

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

This cmdlet shows the same information in a dynamically-sized, formatted table.

```
PS C:\>Get-NetFirewallApplicationFilter -PolicyStore ActiveStore | Format-Table
```

This example retrieves the applications associated with all of the rules in the active store.

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

```
PS C:\>Get-NetFirewallRule -DisplayName "Contoso Messenger" |
Get-NetFirewallApplicationFilter
```

This example gets the application configurations associated with a particular firewall rule.

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

```
PS C:\>Get-NetFirewallRule -DisplayName "Contoso Messenger" |
Get-NetFirewallApplicationFilter | Set-NetFirewallApplicationFilter -Program
%SystemRoot%\System32\messenger.exe
```

An alternate method for performing the same action.

```
PS C:\>Set-NetFirewallRule -DisplayName "Contoso Messenger" -Program
%SystemRoot%\System32\messenger.exe
```

This example changes the application path associated with a particular firewall rule.

-------------------------- EXAMPLE 4 --------------------------

```
PS C:\>$NewPackageSDDL = "S-1-15-2-4292807980-2381230043-3108820062-1451069988-
2614848061-670482394-695399705"




PS C:\>Get-NetFirewallRule -Group Socialite | Get-NetFirewallApplicationFilter
| Set-NetFirewallAddressFilter -Package $NewPackageSDDL
```

This example modifies the package associated with all of the related firewall rules for the Windows Store applications.

REMARKS

To see the examples, type: "get-help Get-NetFirewallApplicationFilter -examples".

For more information, type: "get-help Get-NetFirewallApplicationFilter -detailed".

For technical information, type: "get-help Get-NetFirewallApplicationFilter -full".

For online help, type: "get-help Get-NetFirewallApplicationFilter -online"