PowerShell Get-Help on command 'Set-NetIPsecRule'

PS C:\Users\wahid> Get-Help Set-NetIPsecRule

NAME

Set-NetIPsecRule

SYNOPSIS

Modifies existing IPsec rules.

SYNTAX

```
    Set-NetIPsecRule [-AllowSetKey <Boolean>] [-AllowWatchKey <Boolean>] [-AsJob]
    [-CimSession <CimSession[]>] [-Confirm] [-Description <String>] -DisplayGroup
    <String[]> [-Enabled {True | False}] [-EncryptedTunnelBypass <Boolean>]
    [-ForwardPathLifetime <UInt32>] [-GPOSession <String>] [-InboundSecurity {None
    | Request | Require}] [-InterfaceAlias <WildcardPattern[]>] [-InterfaceType
    {Any | Wired | Wireless | RemoteAccess}] [-KeyModule {Default | IKEv1 | AuthIP
    | IKEv2}] [-LocalAddress <String[]>] [-LocalPort <String[]>]
    [-LocalTunnelEndpoint <String[]>] [-Machine <String>] [-Mode {None | Tunnel |
    Transport}] [-NewDisplayName <String>] [-OutboundSecurity {None | Request |
    Require}] [-PassThru] [-Phase1AuthSet <String>] [-Phase2AuthSet <String>]
    [-Platform <String[]>] [-PolicyStore <String>] [-Profile {Any | Domain |
    Private | Public | NotApplicable}] [-Protocol <String>] [-QuickModeCryptoSet
    <String>] [-RemoteAddress <String[]>] [-RemotePort <String[]>]
```

```
[-RemoteTunnelEndpoint <String[]>] [-RemoteTunnelHostname <String>]
[-RequireAuthorization <Boolean>] [-ThrottleLimit <Int32>] [-User <String>]
[-WhatIf] [<CommonParameters>]

Set-NetIPsecRule [-AllowSetKey <Boolean>] [-AllowWatchKey <Boolean>] [-AsJob]
[-CimSession <CimSession[]>] [-Confirm] [-Description <String>] -DisplayName
<String[]> [-Enabled {True | False}] [-EncryptedTunnelBypass <Boolean>]
[-ForwardPathLifetime <UInt32>] [-GPOSession <String>] [-InboundSecurity {None
| Request | Require}] [-InterfaceAlias <WildcardPattern[]>] [-InterfaceType
{Any | Wired | Wireless | RemoteAccess}] [-KeyModule {Default | IKEv1 | AuthIP
| IKEv2}] [-LocalAddress <String[]>] [-LocalPort <String[]>]
[-LocalTunnelEndpoint <String[]>] [-Machine <String>] [-Mode {None | Tunnel |
Transport}] [-NewDisplayName <String>] [-OutboundSecurity {None | Request |
Require}] [-PassThru] [-Phase1AuthSet <String>] [-Phase2AuthSet <String>]
[-Platform <String[]>] [-PolicyStore <String>] [-Profile {Any | Domain |
Private | Public | NotApplicable}] [-Protocol <String>] [-QuickModeCryptoSet
<String>] [-RemoteAddress <String[]>] [-RemotePort <String[]>]
[-RemoteTunnelEndpoint <String[]>] [-RemoteTunnelHostname <String>]
[-RequireAuthorization <Boolean>] [-ThrottleLimit <Int32>] [-User <String>]
[-WhatIf] [<CommonParameters>]

Set-NetIPsecRule [-IPsecRuleName] <String[]> [-AllowSetKey <Boolean>]
[-AllowWatchKey <Boolean>] [-AsJob] [-CimSession <CimSession[]>] [-Confirm]
[-Description <String>] [-Enabled {True | False}] [-EncryptedTunnelBypass
<Boolean>] [-ForwardPathLifetime <UInt32>] [-GPOSession <String>]
[-InboundSecurity {None | Request | Require}] [-InterfaceAlias
<WildcardPattern[]>] [-InterfaceType {Any | Wired | Wireless | RemoteAccess}]
[-KeyModule {Default | IKEv1 | AuthIP | IKEv2}] [-LocalAddress <String[]>]
[-LocalPort <String[]>] [-LocalTunnelEndpoint <String[]>] [-Machine <String>]
[-Mode {None | Tunnel | Transport}] [-NewDisplayName <String>]
[-OutboundSecurity {None | Request | Require}] [-PassThru] [-Phase1AuthSet
<String>] [-Phase2AuthSet <String>] [-Platform <String[]>] [-PolicyStore
<String>] [-Profile {Any | Domain | Private | Public | NotApplicable}]
```

```
[-Protocol <String>] [-QuickModeCryptoSet <String>] [-RemoteAddress
<String[]>] [-RemotePort <String[]>] [-RemoteTunnelEndpoint <String[]>]
[-RemoteTunnelHostname <String>] [-RequireAuthorization <Boolean>]
[-ThrottleLimit <Int32>] [-User <String>] [-WhatIf] [<CommonParameters>]

Set-NetIPsecRule [-AllowSetKey <Boolean>] [-AllowWatchKey <Boolean>] [-AsJob]
[-CimSession <CimSession[]>] [-Confirm] [-Description <String>] [-Enabled
{True | False}] [-EncryptedTunnelBypass <Boolean>] [-ForwardPathLifetime
<UInt32>] [-GPOSession <String>] -Group <String[]> [-InboundSecurity {None |
Request | Require}] [-InterfaceAlias <WildcardPattern[]>] [-InterfaceType {Any
| Wired | Wireless | RemoteAccess}] [-KeyModule {Default | IKEv1 | AuthIP |
IKEv2}] [-LocalAddress <String[]>] [-LocalPort <String[]>]
[-LocalTunnelEndpoint <String[]>] [-Machine <String>] [-Mode {None | Tunnel |
Transport}] [-NewDisplayName <String>] [-OutboundSecurity {None | Request |
Require}] [-PassThru] [-Phase1AuthSet <String>] [-Phase2AuthSet <String>]
[-Platform <String[]>] [-PolicyStore <String>] [-Profile {Any | Domain |
Private | Public | NotApplicable}] [-Protocol <String>] [-QuickModeCryptoSet
<String>] [-RemoteAddress <String[]>] [-RemotePort <String[]>]
[-RemoteTunnelEndpoint <String[]>] [-RemoteTunnelHostname <String>]
[-RequireAuthorization <Boolean>] [-ThrottleLimit <Int32>] [-User <String>]
[-WhatIf] [<CommonParameters>]

Set-NetIPsecRule [-AllowSetKey <Boolean>] [-AllowWatchKey <Boolean>] [-AsJob]
[-CimSession <CimSession[]>] [-Confirm] [-Description <String>] [-Enabled
{True | False}] [-EncryptedTunnelBypass <Boolean>] [-ForwardPathLifetime
<UInt32>] [-InboundSecurity {None | Request | Require}] -InputObject
<CimInstance[]> [-InterfaceAlias <WildcardPattern[]>] [-InterfaceType {Any |
Wired | Wireless | RemoteAccess}] [-KeyModule {Default | IKEv1 | AuthIP |
IKEv2}] [-LocalAddress <String[]>] [-LocalPort <String[]>]
[-LocalTunnelEndpoint <String[]>] [-Machine <String>] [-Mode {None | Tunnel |
Transport}] [-NewDisplayName <String>] [-OutboundSecurity {None | Request |
Require}] [-PassThru] [-Phase1AuthSet <String>] [-Phase2AuthSet <String>]
[-Platform <String[]>] [-Profile {Any | Domain | Private | Public |
```

NotApplicable}] [-Protocol <String>] [-QuickModeCryptoSet <String>] [-RemoteAddress <String[]>] [-RemotePort <String[]>] [-RemoteTunnelEndpoint <String[]>] [-RemoteTunnelHostname <String>] [-RequireAuthorization <Boolean>] [-ThrottleLimit <Int32>] [-User <String>] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The Set-NetIPsecRule cmdlet modifies existing IPsec rules. This cmdlet gets one or more IPsec rules to be modified with the IPsecRuleName parameter (default), the DisplayName parameter, or by group association using the DisplayGroup or Group parameter. The rules cannot be queried by property in this cmdlet. The Get-NetIPsecRule cmdlet returns the queried rules and pipes the rules into this cmdlet. The remaining parameters specify the properties of the rule to be modified. When the DisplayGroup or Group parameter is specified, then 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 rule to a new GPO, copy the existing rule by running the Copy-NetIPsecRule cmdlet with the NewPolicyStore parameter, then remove the old rule by running the Remove-NetIPsecRule cmdlet.

This cmdlet modifies one or more authentication or cryptographic configurations to use the default settings including the NetIPsecPhase1AuthSet, NetIPsecPhase2AuthSet, or NetIPsecQuickModeCryptoSet object with the Default flag enabled must be done by using dot-notation

PARAMETERS

-AllowSetKey <Boolean>

Indicates that matching IPsec rules of the indicated value are modified. This parameter specifies that the IPsec rule allows trusted intermediaries to override keying material. If this parameter is set to True, then the trusted intermediaries are allowed to manipulate the cryptographic keying material used with an IPsec security association (SA). It is possible that when this parameter is True at both ends, the computer will perform arbitration through SA negotiation so that one end sets the key while the other end watches the key. See the AllowWatchKey parameter for more information. The default value is False. This parameter is only supported on Windows Serverr 2012.

-AllowWatchKey <Boolean>

Indicates that matching IPsec rules of the indicated value are modified. This parameter specifies that the IPsec rule allows trusted intermediaries to provide notification of changes in keying material. If this parameter is set to True, then the trusted intermediaries are allowed to retrieve the cryptographic keying material associated with an IPsec security association (SA), and to subscribe for notification of changes. The default value is False. This parameter is only supported on Windows Server 2012.

-AsJob [<SwitchParameter>]

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

-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.

-Confirm [<SwitchParameter>]

Prompts you for confirmation before running the cmdlet.

-Description <String>

Specifies that matching IPsec rules of the indicated description are modified. Wildcard characters are accepted. This parameter provides information about the IPsec rule. This parameter specifies a localized, user-facing description of the object.

-DisplayGroup <String[]>

Specifies that only matching firewall rules 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-NetIPsecRule 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-NetIPsecRule cmdlet, but can be modified using dot-notation and the Set-NetIPsecRule cmdlet.

-DisplayName <String[]>

Specifies that only matching IPsec rules of the indicated display name are modified. Wildcard characters are accepted. This parameter specifies the localized, user-facing name of the IPsec 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 IPsecRuleName parameter should be used instead, where the default value is a randomly assigned value. This parameter cannot be set to All.

-Enabled <Enabled>

Specifies that matching IPsec rules of the indicated state are modified. 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.

-EncryptedTunnelBypass <Boolean>

Indicates that matching IPsec rules of the specified value are modified. This parameter specifies the encapsulation state for network traffic sent to a tunnel end point that is already IPsec protected. If this parameter is set to True, then the network traffic sent to a tunnel end point that is already IPsec protected does not have to be encapsulated again. This option can improve network performance in the case where network traffic that is already end-to-end protected by other IPsec rules. The default value is False. This parameter is only supported on firstref_server_7 and Windows Server 2012.

-ForwardPathLifetime <UInt32>

Specifies that matching IPsec rules of the specified path lifetime value are modified. This parameter specifies the session key lifetime for an IPsec rule, in minutes. The acceptable values for this parameter are: 78 through 172799. The default value is 0 minutes. This parameter is only supported on Windows Server 2012. When managing a GPO, the default setting is NotConfigured. This parameter is case sensitive and NotConfigured can only be specified using dot-notation.

-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.

-Group <String[]>

Specifies that only matching IPsec 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.

-IPsecRuleName <String[]>

Specifies that only matching IPsec rules 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 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.

-InboundSecurity <SecurityPolicy>

Specifies that matching IPsec rules of the indicated security policy are modified. This parameter determines the degree of enforcement for security on inbound traffic. The acceptable values for this parameter are:

- None: No authentication is requested or required for connections that match the rule. It specifies that the local computer does not attempt authentication for any network connections that match this rule. This option is typically used to grant IPsec exemptions for network connections that do not need to be protected by IPsec, but would otherwise match other rules that could cause the connection to be dropped. - Request: Authentication is requested for connections that match the rule. The local computer attempts to authenticate any inbound network connections that match this rule, but allows the connection if the authentication attempt is no successful. - Require: Authentication is required for connections that match the rule. If the authentication is not successful, then the inbound network traffic is discarded. The default value is None. When the OutboundSecurity parameter is also specified, the following configurations are valid: InboundSecurity \ OutboundSecurity = None\None, Request\None, Request\Request, Require\Request, or Require\Require.

-InputObject <CimInstance[]>

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

-InterfaceAlias <WildcardPattern[]>

Specifies the alias of the interface that applies to the traffic.

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

-InterfaceType <InterfaceType>

Specifies that only network connections made through the indicated interface types are subject to the requirements of this rule. This parameter specifies different authentication requirements for each of the three main network types. The acceptable values for this parameter are: Any, Wired, Wireless, or RemoteAccess. The default value is Any. Querying for rules with this parameter can only be performed using filter objects. See the Get-NetFirewallInterfaceTypeFilter cmdlet for more information.

-KeyModule <KeyModule>

Specifies that matching IPsec rules of the indicated key module are modified. This parameter specifies which keying modules to negotiate. The acceptable values for this parameter are: Default, AuthIP, IKEv1, or IKEv2.

- Default: Equivalent to both IKEv1 and AuthIP. Required in order for the rule to be applied to computers running Windows versions prior to nextref_server_7. ---- There are authentication and cryptographic methods that are only compatible with certain keying modules. This is a very advanced setting intended only for specific interoperability scenarios. Overriding this parameter value may result in traffic being sent in plain-text if the authorization and cryptographic settings are not supported by the keying modules there. - AuthIP: Supported with phase 2 authentication.

- IKEv1: Supported with pre-shared key (PSK), Certificates, and Kerberos.

- IKEv2: Not supported with Kerberos, PSK, or NTLM.

Windows versions prior to Windows Server 2012 only support the Default configuration.

-LocalAddress <String[]>

Specifies that network packets with matching IP addresses match this rule. This parameter value is the first end point of an IPsec rule and specifies the computers that are subject to the requirements of this rule. This parameter value is an IPv4 or IPv6 address, host name, subnet, range, or the following keyword: Any. The acceptable formats for this parameter are: - Single IPv4 Address: 1.2.3.4

- Single IPv6 Address: fe80::1

- IPv4 Subnet (by network bit count): 1.2.3.4/24

- IPv6 Subnet (by network bit count): fe80::1/48

- IPv4 Subnet (by network mask): 1.2.3.4/255.255.255.0

- IPv4 Range: 1.2.3.4 through 1.2.3.7

- IPv6 Range: fe80::1 through fe80::9

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

-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.

-LocalTunnelEndpoint <String[]>

Specifies the IP address of the computer or gateway device that sends traffic from computers that match the LocalAddress parameter value to computers that match the RemoteAddress parameter value. The traffic is being secured from this IP address to the device identified in the RemoteTunnelEndpoint parameter. This parameter value must use the same type of IP address as the RemoteTunnelEndpoint parameter, which is either IPv4 or IPv6. This parameter is required and valid only for tunnel mode rules. Address keywords are not supported. On firstref_client_7, nextref_server_7, and Windows Server 2012, this value can also be Any. When applied to a client computer, this option supports connection via a tunnel to a remote gateway or host regardless of the IP address or address type of the local computer.

-Machine <String>

Specifies that matching IPsec rules of the indicated computer accounts are modified. This parameter specifies that only network packets that are authenticated as incoming from or outgoing to a computer identified in the list of computer accounts (SID) match this rule. This parameter value is specified as an SDDL string.

-Mode <IPsecMode>

Specifies that matching IPsec rules of the indicated mode are modified. This parameter specifies the type of IPsec mode connection that the IPsec rule defines. The acceptable values for this parameter are: None, Transport, or Tunnel. The default value is Transport.

-NewDisplayName <String>

Specifies the new display name for an IPsec rule.

-OutboundSecurity <SecurityPolicy>

Specifies that matching IPsec rules of the indicated security policy are modified. This parameter determines the degree of enforcement for security on outbound traffic. The acceptable values for this parameter are:

- None: No authentication is requested or required for connections that match the rule. It specifies that the local computer does not attempt authentication for any network connections that match this rule. This option is typically used to grant IPsec exemptions for network connections that do not need to be protected by IPsec, but would otherwise match other rules that could cause the connection to be dropped. - Request: Authentication is requested for connections that match the rule. The local computer attempts to authenticate any outbound network connections that match this rule, but allows the connection if the authentication attempt fails. - Require: Authentication is required for connections that match the rule. If the authentication is not successful, then the outbound network traffic is discarded.

The default value is None. When the InboundSecurity parameter is also specified, the following configurations are valid: InboundSecurity / OutboundSecurity = None\None, Request\None, Request\Request, Require\Request, or Require\Require.

-PassThru [<SwitchParameter>]

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

-Phase1AuthSet <String>

Gets the IPsec rules that are associated with the given phase 1 authentication set to be modified. A NetIPsecPhase1AuthSet object represents the phase 1 authorization set conditions associated with an IPsec or main mode rule. This parameter sets the methods for main mode negotiation by describing the proposals for computer authentication. See the Get-NetIPsecPhase1AuthSet cmdlet for more information. Alternatively, this parameter can be used for the same purpose, but does not allow the authentication set to be piped into the cmdlet and the set must be specified with the IPsecRuleName parameter.

-Phase2AuthSet <String>

Gets the IPsec rules that are associated with the given phase 2 authentication set to be copied. A NetIPsecPhase2AuthSet object represents the phase 2 authorization set conditions associated with an IPsec or main mode rule. This parameter sets the methods for main mode negotiation by describing the proposals for computer authentication. See the Get-NetIPsecPhase2AuthSet cmdlet for more information. Alternatively, this parameter can be used for the same purpose, but does not allow the authentication set to be piped into the cmdlet and the set must be specified with the IPsecRuleName parameter.

-Platform <String[]>

Specifies which version of Windows the associated rule applies. The acceptable format for this parameter is a number in the Major.Minor format. The version number of 6.0 corresponds to Vista (firstref_vista), 6.1 corresponds to Win7 (Windowsr 7 or nextref_server_7), and 6.2 corresponds to Win8 (Windowsr 8 or Windows Server 2012). If + is not specified, then only that version is associated. If + is specified, then that version and later are associated. Querying for rules with this parameter with the Get-NetIPsecRule cmdlet cannot be performed.

-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 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. 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 this Copy-NetIPsecRule cmdlet or with the New-NetIPsecRule cmdlet.

-Profile <Profile>

Specifies one or more profiles to which the rule is assigned. The rule is active on the local computer only when the specified profile is currently active. This relationship is many-to-many and can be indirectly modified by the user, by changing the Profiles field on instances of firewall rules. Only one profile is applied at a time. The acceptable values for this parameter are: Any, Domain, Private, Public, or NotApplicable. The default is Any. Separate multiple entries with a comma and do not include any spaces. Use the keyword Any to configure the profile as Private, Public, Domain in the ConfigurableServiceStore.

-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 through 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. See the Get-NetFirewallPortFilter cmdlet for more information.

-QuickModeCryptoSet <String>

Specifies that matching IPsec rules of the specified quick mode cryptographic set are modified. This parameter specifies the quick mode cryptographic set to be associated with the IPsec rule. A NetIPsecMainModeCryptoSet object represents quick mode cryptographic conditions associated with an IPsec rule. This parameter sets the methods for quick mode negotiation by describing the proposals for encryption. See the New-NetIPsecQuickModeCryptoSet cmdlet for more information.

-RemoteAddress <String[]>

Specifies that network packets with matching IP addresses match this rule. This parameter value is the second end point of an IPsec rule and specifies the computers that are subject to the requirements of this rule. This parameter value is an IPv4 or IPv6 address, host name, subnet, range, or the following keyword: Any. The acceptable formats for this parameter are: - Single IPv4 Address: 1.2.3.4

- Single IPv6 Address: fe80::1

- IPv4 Subnet (by network bit count): 1.2.3.4/24

- IPv6 Subnet (by network bit count): fe80::1/48

- IPv4 Subnet (by network mask): 1.2.3.4/255.255.255.0

- IPv4 Range: 1.2.3.4 through 1.2.3.7

- IPv6 Range: fe80::1 through fe80::9

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

-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.

-RemoteTunnelEndpoint <String[]>

Specifies the IP address of the computer or gateway device that secures traffic from computers that match the LocalAddress parameter value to computers that match the RemoteAddress parameter value. The traffic is being secured to this IP address to the device identified in the LocalTunnelEndpoint parameter. This parameter value must use the same type of IP address as the LocalTunnelEndpoint parameter, which is either IPv4 or IPv6. Address keywords are not supported. On Windowsr 7, nextref_server_7, and Windows Server 2012, this parameter value can also be Any. When applied to a client computer, this option supports connection via a tunnel to a remote gateway or host regardless of the IP address or address type of the local computer.

-RemoteTunnelHostname <String>

Specifies that matching IPsec rules of the specified second end point tunnel host name are modified. Specifies a fully qualified DNS name that resolves to a list of remote tunnel end points. This parameter is only supported on Windows Server 2012. This parameter can only be used with multiple remote tunnel end points. Specifying this parameter prevents a non-asymmetric tunnel mode IPsec rule from being created. Rule creation will fail when a single remote tunnel end point and this parameter are specified, or when remote tunnel end point is Any and this parameter is specified.

-RequireAuthorization <Boolean>

Indicates that matching IPsec rules of the specified value are modified. Specifies the given value for an IPsec rule. If this parameter is set to True, then enforcement of authorization is allowed for end points. This parameter is only supported on nextref_server_7 and Windows Server 2012.

-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.

-User <String>

Specifies that matching IPsec rules of the indicated user accounts are modified. This parameter specifies that only network packets that are authenticated as incoming from or outgoing to a user identified in the list of user accounts match this rule. This parameter value is specified as an SDDL string.

-WhatIf [<SwitchParameter>]

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

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

```
PS C:\>$kerbComputer = New-NetIPsecAuthProposal -Kerberos -Machine
PS C:\>$Phase1AuthSet = New-NetIPsecPhase1AuthSet -DisplayName "Computer Kerb
Auth" -Proposal $kerbComputer




PS C:\>Set-NetIPsecRule -DisplayName "SecureNet Rule" -Phase1AuthSet
$Phase1AuthSet.Name
```

This example replaces the proposals of an existing IPsec rule.

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

```
PS C:\>Set-NetIPsecRule -DisplayGroup "DA Client" -Enabled True




PS C:\>Enable-NetIPsecRule -DisplayGroup "DA Client"
```

This example shows two ways to enables all of the IPsec rules in a predefined group.

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

```
PS C:\>Set-NetIPsecRule -DisplayName "Tunnel Mode - (DA Client)"
-NewDisplayName "Tunnel Mode - Americas (DA Client)"
```

This example changes the display name of an IPsec rule

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

```
PS C:\>$rule = Get-NetIPsecRule -DisplayName "IPsec Rule"
```

```
    PS C:\>$rule.QuickModeCryptoSet = "Default"




    PS C:\>Set-NetIPsecRule -InputObject $rule


    This example modifies an IPsec rule to use the default encryption method for
    main mode if a custom main mode has been previously set to the rule.
REMARKS
    To see the examples, type: "get-help Set-NetIPsecRule -examples".
    For more information, type: "get-help Set-NetIPsecRule -detailed".
    For technical information, type: "get-help Set-NetIPsecRule -full".
    For online help, type: "get-help Set-NetIPsecRule -online"
```