PowerShell Get-Help on command 'New-NetIPsecQuickModeCryptoSet'

PS C:\Users\wahid> Get-Help New-NetIPsecQuickModeCryptoSet

NAME

    New-NetIPsecQuickModeCryptoSet

SYNOPSIS

    Creates a quick mode cryptographic set that contains suites of cryptographic

    protocols to offer in IPsec quick mode negotiations with other computers.

SYNTAX

    New-NetIPsecQuickModeCryptoSet [-AsJob] [-CimSession <CimSession[]>]

    [-Confirm] [-Default] [-Description <String>] -DisplayName <String>

    [-GPOSession <String>] [-Group <String>] [-Name <String>]

    [-PerfectForwardSecrecyGroup {None | DH1 | DH2 | DH14 | DH19 | DH20 | DH24 |

    SameAsMainMode}] [-PolicyStore <String>] -Proposal <CimInstance[]>

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

DESCRIPTION

    The New-NetIPsecQuickModeCryptoSet cmdlet creates a cryptographic set to be

    used to negotiate a quick mode security association (SA).

A quick mode cryptographic set is an ordered list of cryptographic proposals. Each proposal in the set specifies a set of IPsec protocols such as authentication header (AH) or encapsulating security payload (ESP), hashing algorithms, and algorithms for encryption, if requested. A proposal is created by running the New-NetIPsecQuickModeCryptoProposal cmdlet. The quick mode exchange will use the first proposal that the peers have in common.

The default quick mode cryptographic set is used with all IPsec rules. An IPsec rule is specified by the using the Default parameter at creation time. Additional quick mode cryptographic sets can be used with IPsec rules for fully customized quick mode negotiations.

PARAMETERS

  -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.

  -Default [<SwitchParameter>]

    Specifies the customized parameters for overriding the defaults for main mode encryption, making it the new default setting for encryption. For the default QuickModeCryptoSet object, the default Name parameter value is {E5A5D32A-4BCE-4e4d-B07F-4AB1BA7E5FE2}. To retrieve default settings,

query by using the default Name parameter value. To specify a different default cryptographic set, delete the current default set and use the Rename-NetIPsecQuickModeCryptoSet cmdlet to specify the default set with {E5A5D32A-4BCE-4e4d-B07F-4AB1BA7E5FE2}.

-Description <String>

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

-DisplayName <String>

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

-GPOSession <String>

Targets the network Group Policy Object (GPO) from which to retrieve the rules to be created. 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 firewall rules of the indicated group
association are created. 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.

-Name <String>

Specifies that only matching main mode cryptographic sets of the indicated
name are created. 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 the same name in a higher-precedence
GPO, then it overrides other rules that exist. The default value is a
randomly assigned value. When you want to override the defaults for main
mode encryption, specify the customized parameters and set this parameter
value, making this parameter the new default setting for encryption.

-PerfectForwardSecrecyGroup <DiffieHellmanGroup>

Specifies the Diffie-Hellman group for session key perfect forward secrecy. The acceptable values for this parameter are: None, DH1, DH2, DH14, DH19, DH20, DH24, or SameAsMainMode

-PolicyStore <String>

Targets the policy store from which to retrieve the sets to be created. 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-NetIPsecMainModeCryptoSet 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-NetIPsecMainModeCryptoSet cmdlet or with this cmdlet.

-Proposal <CimInstance[]>
    Associates the specified cryptographic proposal to the corresponding cryptographic set to be used in main mode negotiations. Separate multiple entries with a comma.

-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.

-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:\>$qMProposal = New-NetIPsecQuickModeCryptoProposal -Encapsulation ESP
-ESPHash SHA1 -Encryption DES3

PS C:\>$qMCryptoSet = New-NetIPsecQuickModeCryptoSet -DisplayName
"esp:sha1-des3" -Proposal $qMProposal

PS C:\>New-NetIPSecRule -DisplayName "Tunnel from HQ to Dallas Branch" -Mode
Tunnel -LocalAddress 192.168.0.0/16 -RemoteAddress 192.157.0.0/16
-LocalTunnelEndpoint 1.1.1.1 -RemoteTunnelEndpoint 2.2.2.2 -InboundSecurity
Require -OutboundSecurity Require -QuickModeCryptoSet $qMCryptoSet.Name

This example creates an IPsec tunnel that routes traffic from a private

network at 192.168.0.0/16 through an interface on the local computer at

1.1.1.1 attached to a public network to a second computer through a public

interface at 2.2.2.2 to another private network at 192.157.0.0/16. All traffic

through the tunnel is integrity checked using ESP and SHA1, and encrypted

using ESP and DES3.

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

This cmdlet illustrates how to include both AH and ESP protocols in a single suite.

```
PS C:\>$aHandESPQM = New-NetIPsecQuickModeCryptoProposal -Encapsulation AH,ESP
-AHHash SHA1 -ESPHash SHA1 -Encryption DES3
```

This cmdlet illustrates how to specify the use of the AH protocol only.

```
PS C:\>$aHQM = New-NetipsecQuickModeCryptoProposal -Encapsulation AH -AHHash
SHA1 -ESPHash None -Encryption None
```

This cmdlet illustrates how to specify the use of the ESP protocol only, and uses the None keyword to specify not to include an encryption option.

```
PS C:\>$eSPQM = New-NetIPsecQuickModeCryptoProposal -Encapsulation ESP
-ESPHash SHA1 -Encryption None
```

This cmdlet illustrates how to use the None keyword to specify that ESP is used with an encryption protocol, but with no integrity protocol. This cmdlet also illustrates how to set a custom SA timeout using both time and data amount values.

```
PS C:\>$eSPnoAHQM = New-NetIPsecQuickModeCryptoProposal -Encapsulation ESP
-ESPHash None -Encryption AES256 -MaxKiloBytes 50000 -MaxMinutes 30
```

```
PS C:\>$qMCryptoSet = New-NetIPsecQuickModeCryptoSet -DisplayName "Custom
Quick Mode" -Proposal $aHandESPQM, $aHQM, $eSPQM, $eSPnoAHQM
```

```
PS C:\>New-NetIPsecRule -DisplayName "Domain Isolation Rule" -InboundSecurity
```

"Require Request" -OutboundSecurity Request -QuickModeCryptoSet

$qMCryptoSet.Name

This example creates a domain isolation rule, but uses a custom quick mode

proposal that includes multiple quick mode suites, separated by commas.

REMARKS

To see the examples, type: "get-help New-NetIPsecQuickModeCryptoSet -examples".

For more information, type: "get-help New-NetIPsecQuickModeCryptoSet

-detailed".

For technical information, type: "get-help New-NetIPsecQuickModeCryptoSet

-full".

For online help, type: "get-help New-NetIPsecQuickModeCryptoSet -online"