



python



PowerShell

FPDF Library
PDF generator

Full credit is given to the above companies including the OS that this PDF file was generated!

PowerShell Get-Help on command 'Get-NetIPsecMainModeCryptoSet'

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

NAME

Get-NetIPsecMainModeCryptoSet

SYNOPSIS

Gets main mode cryptographic sets from the target computer.

SYNTAX

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

```
Get-NetIPsecMainModeCryptoSet [-AsJob] -AssociatedNetIPsecMainModeRule
<CimInstance> [-CimSession <CimSession[]>] [-GPOSession <String>]
[-PolicyStore <String>] [-ThrottleLimit <Int32>] [-TracePolicyStore]
[<CommonParameters>]
```

```
Get-NetIPsecMainModeCryptoSet [-AsJob] [-CimSession <CimSession[]>]
[-Description <String[]>] [-DisplayGroup <String[]&;] [-ForceDiffieHellman
<Boolean[]>] [-GPOSession <String>] [-Group <String[]>] [-MaxMinutes
<UInt32[]>] [-MaxSessions <UInt32[]>] [-PolicyStore <String>]
```

```
[-PolicyStoreSource <String[]>] [-PolicyStoreSourceType {None | Local |
GroupPolicy | Dynamic | Generated | Hardcoded}] [-PrimaryStatus {Unknown | OK
| Inactive | Error}] [-Status <String[]>] [-ThrottleLimit <Int32>]
[-TracePolicyStore] [<CommonParameters>]
```

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

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

DESCRIPTION

The `Get-NetIPsecMainModeCryptoSet` cmdlet returns the instances of cryptographic sets that match the search parameters from the user. See the `New-NetIPsecMainModeCryptoSet` cmdlet for more information.

This cmdlet returns main mode cryptographic sets 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 into other cmdlets for further modifications or monitoring.

PARAMETERS

`-All` [<SwitchParameter>]

Indicates that all of the main mode cryptographic sets 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.

-AssociatedNetIPsecMainModeRule <CimInstance>

Gets the main mode cryptographic sets that are associated, via the pipeline, with the input main mode rule to be retrieved. This parameter represents a main mode rule, which alters the behavior of main mode authentications. Main mode negotiation establishes a secure channel between two computers by determining a set of cryptographic protection suites, exchanging keying material to establish a shared secret key, and authenticating computer and user identities. See the `Get-NetIPsecMainModeRule` 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.

-Description <String[]>

Specifies that matching main mode cryptographic sets of the indicated description are retrieved. Wildcard characters are accepted. This parameter provides information about the main mode cryptographic sets. This parameter specifies a localized, user-facing description of the object.

-DisplayGroup <String[]>

Specifies that only matching main mode cryptographic sets 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-NetIPsecMainModeCryptoSet` 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 with a universal and world-ready indirect @FirewallAPI name. This parameter cannot be specified upon object creation using the New-NetIPsecMainModeCryptoSet cmdlet, but can be modified using dot notation and the Set-NetIPsecMainModeCryptoSet cmdlet.

-DisplayName <String[]>

Specifies that only matching main mode cryptographic sets of the indicated display name are retrieved. Wildcard characters are accepted. This parameter specifies the localized, user-facing name of a single main mode cryptographic sets. When creating a set 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 resilient scripts, the Name parameter should be used instead, where the default value is a randomly assigned value. This parameter value cannot be All.

-ForceDiffieHellman <Boolean[]>

Indicates that matching main mode cryptographic sets of the indicated value are retrieved. If this parameter is set to True, then IPsec uses Diffie-Hellman exchanges to protect the main mode key exchange when AuthIP is used. AuthIP is specified by KeyModule. This provides stronger security for the key exchange. The default value is False.

-GPOSession <String>

Specifies the network GPO from which to retrieve the sets to be retrieved. This parameter is used in the same way as the PolicyStore parameter. When modifying Group Policy Objects (GPOs) in Windows PowerShell, 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 main mode cryptographic sets 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 organizes rules by influence and allows batch rule modifications. Using the Set-NetIPsecMainModeCryptoSet 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 this parameter with a universal and world-ready indirect @FirewallAPI name. The DisplayGroup parameter cannot be specified upon object creation using the New-NetIPsecMainModeCryptoSet cmdlet, but can be modified using dot notation and the Set-NetIPsecMainModeCryptoSet cmdlet.

-MaxMinutes <UInt32[]>

Specifies that matching main mode cryptographic sets of the indicated maximum lifetime, in minutes, are retrieved. This parameter specifies the number of minutes established for a main mode security association before it expires and must be renegotiated. The acceptable values for this parameter are: 0 through 2879.

- A non-zero value specifies the desired minute lifetime.

- NotConfigured: Valid only when configuring a GPO.

Removes the setting from the GPO, which results in the policy not changing the value on the computer when the policy is applied.

The default value is 4800 minutes (eight hours). When managing a GPO, the default setting is NotConfigured.

-MaxSessions <UInt32[]>

Specifies that matching main mode cryptographic sets of the indicated maximum lifetime, in sessions, are retrieved. This parameter specifies the number of sessions established for a main mode security association before it expires and must be renegotiated. The acceptable values for this parameter are: 0 to 2147483647.

- A value of zero (0) specifies that there should be no maximum session lifetime.

- A non-zero value specifies the desired session number.

- NotConfigured: Valid only when configuring a GPO.

Removes the setting from the GPO, which results in the policy not changing the value on the computer when the policy is applied.

The default value is 0 sessions. When managing a GPO, the default setting is NotConfigured.

-Name <String[]>

Specifies that only matching main mode cryptographic sets of the indicated name are retrieved. 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 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, making this parameter the new default setting for encryption.

-PolicyStore <String>

Specifies the policy store from which to retrieve the sets 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 system 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.

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 the New-NetIPsecMainModeCryptoSet cmdlet.

-PolicyStoreSource <String[]>

Specifies that the main mode cryptographic sets that match 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.

-PolicyStoreSourceType <PolicyStoreType[]>

Specifies that the main mode cryptographic sets 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.

-PrimaryStatus <PrimaryStatus[]>

Specifies that the main mode cryptographic sets that match the indicated primary status are retrieved. This parameter describes 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.

status information.

-Status <String[]>

Specifies that the main mode cryptographic sets 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. This parameter value should not be modified.

-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.

-TracePolicyStore [<SwitchParameter>]

Indicates that the main mode cryptographic sets that match the indicated policy store are retrieved. This parameter specifies that the name of the source GPO is queried and set to the PolicyStoreSource parameter value.

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see [about_CommonParameters \(https://go.microsoft.com/fwlink/?LinkID=113216\)](https://go.microsoft.com/fwlink/?LinkID=113216).

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

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

This example gets all of the main mode cryptographic sets in the currently

active policy, which is the sum of all policy stores that apply to the computer. Running this cmdlet without specifying the policy store retrieves the persistent store.

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

```
PS C:\>Get-NetIPsecMainModeCryptoSet -ForceDiffieHellman $false
```

This example gets all of the main mode cryptographic sets that do not use the Diffie-Hellman exchange to protect the main mode key exchange.

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

```
PS C:\>$proposal1 = New-NetIPsecMainModeCryptoProposal -KeyExchange DH1
```

```
PS C:\>$proposal2 = New-NetIPsecMainModeCryptoProposal -KeyExchange DH14
```

```
PS C:\>$cryptoset1 = (New-NetIPsecMainModeCryptoSet -DisplayName  
MainModeCryptoSet -Proposal $proposal1.Name, $proposal2.Name)
```

```
PS C:\>$mainModeRule = New-NetIPsecMainModeRule -DisplayName MainModeRule  
-MainModeCryptoSet $cryptoset1
```

```
PS C:\>$mainModeCryptoSet = ($MainModeRule | Get-NetIPsecQuickModeCryptoSet)
```

```
PS C:\>$mainModeCryptoSet.Proposal[1] = DH19
```

```
PS C:\>$mainModeCryptoSet | Set-NetIPsecMainModeCryptoSet
```

The following cmdlets shows an alternative method to the previous cmdlets. The main mode rule setup is the same.

```
PS C:\>$mainModeRule = New-NetIPsecMainModeRule -DisplayName MainModeRule  
-MainModeCryptoSet (New-NetIPsecMainModeCryptoSet -DisplayName  
MainModeCryptoSet -Proposal (New-NetIPsecMainModeCryptoProposal -KeyExchange  
DH1),(New-NetIPsecMainModeCryptoProposal -KeyExchange DH14)).Name
```

```
PS C:\>$mainModeCryptoSet = ($mainModeRule | Get-NetIPsecQuickModeCryptoSet)
```

```
PS C:\>$mainModeCryptoSet | Set-NetIPsecMainModeCryptoSet -Proposal  
(New-NetIPsecMainModeCryptoProposal -KeyExchange DH1),  
(New-NetIPsecMainModeCryptoProposal -KeyExchange DH19)
```

This example shows how to replace a key exchange option of a main mode cryptographic proposal to an existing main mode cryptographic set, given the associated main mode rule. The key exchange will be changed for the second specified cryptographic proposal.

REMARKS

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

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

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

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