Windows PowerShell Get-Help on Cmdlet 'Get-DAPolicyChange'

PS:\>Get-HELP Get-DAPolicyChange -Full

NAME

Get-DAPolicyChange

SYNOPSIS

Gets a list of IP addresses that need to be added and deleted to an IPsec rule based on the differences detected between the IP addresses for the existing rule and

the IP addresses derived from the input parameters, and creates a Windows PowerShellr script (.ps1) that updates the IPsec rule in the appropriate policy stores.

SYNTAX

Get-DAPolicyChange [[-Servers] <String[]>] [[-Domains] <String[]>] [-DisplayName] <String> [[-PolicyStore] <String>] [-PSLocation] <String> [-EndpointType] <String>

[[-DnsServers] <String[]>] [<CommonParameters>]

DESCRIPTION

The Get-DAPolicyChange cmdlet returns the detected differences between the IP addresses (remote and local addresses) of an existing IPsec rule, and the IP addresses

derived by the input parameters. This cmdlet also creates a Windows PowerShellr script (`.ps1`) that updates the IPsec

rule end points with the retrieved IP

addresses. The created script contains instances of the Update-NetIPsecRule cmdlet, that adds or deletes IP addresses to or from IPsec rules.

This cmdlet is used to keep the IPsec policies for client and server refreshed in DirectAccess (DA) deployments in a double tunnel model. The DA first tunnel policy

is defined by IP addresses that are derived from domain names and servers. A list of IP addresses is retrieved based on the derived values from the Domains or Servers

parameter. This cmdlet outputs DeltaCollection objects that contain the following: the actual list of address changes detected, whether to add or delete the change in

IP addresses, and a list of fully qualified domain names (FQDNs) that did not resolve. If there are multiple rules that match the same name, then this cmdlet fails

with an error.

Running the output script for this cmdlet (located at PSLocation) resolves the IP addresses for the DA first tunnel and updates the Group Policy Objects (GPOs)

appropriately. The DNS server specified in the DnsServers parameter will be used to resolve the domain name and server names.

By generating a Windows PowerShell script, this cmdlet allows administrators to have greater control over policy synchronization. The Sync-NetIPsecRule cmdlet also

detects the IP address changes, but immediately updates the rules instead of returning the deviations and a `.ps1` script.

PARAMETERS

-DisplayName <String>

Specifies the display name to match the differences in IP addresses of the IPsec rule.

Required? true

Position? 2

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-DnsServers <String[]>

Specifies a list of DNS server IP addresses that will be used for name resolution used to determine IP address differences. This parameter accepts one or more DNS

server IP addresses. If this parameter is not specified, then this cmdlet uses the default DNS servers.

Required? false

Position? 6

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Domains <String[]>

Specifies the domains from which the deltas in IP addresses are derived. The list is specified by an array of fully qualified domain names (FQDN).

Required? false

Position? 1

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EndpointType <String>

Specifies that the local or remote endpoint should be modified by adding or removing the IP address differences. The acceptable values for this parameter are:

Endpoint1 or Endpoint2. Endpoint1 corresponds to the local address and Endpoint2 corresponds to the remote address for any IPsec rule.

Required? true

Position? 5

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PSLocation <String>

Specifies the path for the newly created Windows PowerShell script (`.ps1`) file. This parameter supports standard Windows PowerShell path syntax. This parameter

must contain a rooted path, such as `C:\users\User1\WPS_Script.ps1`.

Required? true

Position? 4

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-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 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.

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

Required? false

Position? 3

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Servers <String[]>

Specifies a list of server IP addresses that will be used to derive IP address differences.

Required? false

Position? 0

Default value None

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

None

OUTPUTS

Microsoft.Management.Infrastructure.CimInstance#root\StandardCimv2\DeltaCollection[]

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.

NOTES

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

PS C:\>Get-DAPolicyChange -DisplayName "TunnelPolicy1" -EndpointType Endpoint1 -PSLocation "C:\Update.ps1" -Servers "server1.corp.contoso.com",

"server2.corp.contoso.com", "server3.corp.contoso.com"

IPsec Rule name : TunnelPolicy1

Action : Add

IPv6addresses : 2001:4829:3243::100:1

: 2001:4829:3243::100:1

GPO : contoso\DAClientPolicy

IPsec Rule name : TunnelPolicy1

Action : Delete

IPv6addresses : 2001:4829:3243::100:3

: 2001:4829:3243::100:4

GPO : contoso\DAClientPolicy

FQDN's that did not resolve into IP address:

server1.corp.contoso.com

server3.corp.contoso.com

This example gets the list of IP addresses that need to be added and deleted to an IPsec rule based on the differences detected between the existing rule IP addresses

and the IP addresses derived from the input parameters and returns a .ps1 file that updates the local end point for the rule.

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

PS C:\>$serverPolicyStore = domain.contoso.com/server_GPO

PS C:\>$serverRuleDisplayName = "Any-Traffic-Win8DA-Rule"

PS C:\>$domains = "corp.contoso.com", "corp.contoso2.com"

PS C:\>$servers = "server2.corp.contoso.com"

PS C:\>$primaryDns64 = 1.2.2.1

PS C:\>Get-DAPolicyChange -PolicyStore $serverPolicyStore -DisplayName $serverRuleDisplayName -EndpointType Endpoint1 -Domains $domains -Servers $servers -DNSServers
$primaryDns64 -AddressType IPv6 -PSLocation C:\Users\Administrator\Documents\PSscripts\dapolicychange.ps1

This example gets the list of IP addresses that need to be added and deleted to an IPsec rule based on the differences detected between the existing rule IP addresses
and the IP addresses derived from the input parameters and returns a .ps1 file that updates the end points.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/netsecurity/get-dapolicychange?view=windowsserver2022-ps&wt.mc_id=ps-gethelp

Sync-NetIPsecRule

Update-NetIPsecRule