



Windows PowerShell Get-Help on Cmdlet 'Update-AzPolicyExemption'

PS:\>Get-HELP Update-AzPolicyExemption -Full

NAME

Update-AzPolicyExemption

SYNOPSIS

This operation updates a policy exemption with the given scope and name.

SYNTAX

```
Update-AzPolicyExemption -Name <String> [-Scope <String>] [-ExemptionCategory <String>]
[-PolicyDefinitionReferenceId <String[]>] [-DisplayName <String>]
[-Description <String>] [-ExpiresOn <Nullable`1>] [-ClearExpiration] [-Metadata <String>] [-BackwardCompatible]
[-AssignmentScopeValidation <String>] [-DefaultProfile
<PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy
<Uri>] [-ProxyCredential <PSCredential>]
[-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]
```

```
Update-AzPolicyExemption [-ExemptionCategory <String>] [-PolicyDefinitionReferenceId <String[]>] -Id <String>
[-DisplayName <String>] [-Description <String>]
[-ExpiresOn <Nullable`1>] [-ClearExpiration] [-Metadata <String>] [-BackwardCompatible] [-AssignmentScopeValidation
<String>] [-DefaultProfile <PSObject>] [-Break]
```

[-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy <Uri>] [-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]

Update-AzPolicyExemption [-ExemptionCategory <String>] [-PolicyDefinitionReferenceId <String[]>] [-DisplayName <String>] [-Description <String>] [-ExpiresOn <Nullable`1>] [-ClearExpiration] [-Metadata <String>] [-BackwardCompatible] [-AssignmentScopeValidation <String>] [-InputObject <IPolicyExemption>] [-DefaultProfile <PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy <Uri>] [-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]

DESCRIPTION

This operation updates a policy exemption with the given scope and name.

PARAMETERS

-Name <String>
The name of the policy exemption.

Required? true
Position? named
Default value
Accept pipeline input? true (ByPropertyName)
Accept wildcard characters? false

-Scope <String>
The scope of the policy exemption.

Valid scopes are: management group (format: '/providers/Microsoft.Management/managementGroups/{managementGroup}'), subscription (format: '/subscriptions/{subscriptionId}'), resource group (format: '/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}'), or resource (format: *Page 2/12*

'/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/{resourceProviderNamespace}/{parentResourcePath}/{resourceType}/{resourceName}'

Required? false
Position? named
Default value
Accept pipeline input? true (ByPropertyName)
Accept wildcard characters? false

-ExemptionCategory <String>

The policy exemption category

Required? false
Position? named
Default value
Accept pipeline input? true (ByPropertyName)
Accept wildcard characters? false

-PolicyDefinitionReferenceId <String[]>

The policy definition reference ID list when the associated policy assignment is for a policy set (initiative).

Required? false
Position? named
Default value
Accept pipeline input? true (ByPropertyName)
Accept wildcard characters? false

-Id <String>

The ID of the policy assignment to delete.

Use the format '{scope}/providers/Microsoft.Authorization/policyAssignments/{policyAssignmentName}'.

Required? true

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-DisplayName <String>

The display name of the policy assignment.

Required? false

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-Description <String>

This message will be part of response in case of policy violation.

Required? false

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-ExpiresOn <Nullable`1>

The expiration date and time (in UTC ISO 8601 format yyyy-MM-ddTHH:mm:ssZ) of the policy exemption.

Required? false

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-ClearExpiration [<SwitchParameter>]

Indicates whether to clear the expiration date and time of the policy exemption.

Required?	false
Position?	named
Default value	False
Accept pipeline input?	true (ByPropertyName)
Accept wildcard characters?	false

-Metadata <String>

The policy assignment metadata.

Metadata is an open ended object and is typically a collection of key value pairs.

Required?	false
Position?	named
Default value	
Accept pipeline input?	true (ByPropertyName)
Accept wildcard characters?	false

-BackwardCompatible [<SwitchParameter>]

Causes cmdlet to return artifacts using legacy format placing policy-specific properties in a property bag object.

Required?	false
Position?	named
Default value	False
Accept pipeline input?	false
Accept wildcard characters?	false

-AssignmentScopeValidation <String>

The option whether validate the exemption is at or under the assignment scope.

Required?	false
Position?	named
Default value	

Accept pipeline input? false

Accept wildcard characters? false

-InputObject <IPolicyExemption>

Required? true

Position? named

Default value

Accept pipeline input? true (ByValue, ByPropertyName)

Accept wildcard characters? false

-DefaultProfile <PSObject>

The DefaultProfile parameter is not functional.

Use the SubscriptionId parameter when available if executing the cmdlet against a different subscription.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Break [<SwitchParameter>]

Wait for .NET debugger to attach

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-HttpPipelineAppend <SendAsyncStep[]>

SendAsync Pipeline Steps to be appended to the front of the pipeline

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-HttpPipelinePrepend <SendAsyncStep[]>

SendAsync Pipeline Steps to be prepended to the front of the pipeline

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-Proxy <Uri>

The URI for the proxy server to use

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-ProxyCredential <PSCredential>

Credentials for a proxy server to use for the remote call

Required? false
Position? named
Default value
Accept pipeline input? false
Accept wildcard characters? false

-ProxyUseDefaultCredentials [<SwitchParameter>]

Use the default credentials for the proxy

Required? false

Position? named

Default value False

Accept pipeline input? false

Accept wildcard characters? false

-WhatIf [<SwitchParameter>]

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Confirm [<SwitchParameter>]

Required? false

Position? named

Default value

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 \(https://go.microsoft.com/fwlink/?LinkID=113216\)](https://go.microsoft.com/fwlink/?LinkID=113216).

INPUTS

System.Management.Automation.SwitchParameter

System.Nullable`1[[System.DateTime, System.Private.CoreLib, Version=8.0.0.0, Culture=neutral, PublicKeyToken=7cec85d7bea7798e]]

System.String

System.String[]

OUTPUTS

Microsoft.Azure.PowerShell.Cmdlets.Policy.Models.IPolicyExemption

NOTES

COMPLEX PARAMETER PROPERTIES

To create the parameters described below, construct a hash table containing the appropriate properties. For information on hash tables, run Get-Help

about_Hash_Tables.

INPUTOBJECT <IPolicyExemption>:

ExemptionCategory <String>: The policy exemption category. Possible values are Waiver and Mitigated.

PolicyAssignmentId <String>: The ID of the policy assignment that is being exempted.

[AssignmentScopeValidation <String>]: The option whether validate the exemption is at or under the assignment scope.

[Description <String>]: The description of the policy exemption.

[DisplayName <String>]: The display name of the policy exemption.

[ExpiresOn <DateTime?>]: The expiration date and time (in UTC ISO 8601 format yyyy-MM-ddTHH:mm:ssZ) of the policy exemption.

[Metadata <IPolicyExemptionPropertiesMetadata>]: The policy exemption metadata. Metadata is an open ended object and is typically a collection of key value pairs.

[(Any) <Object>]: This indicates any property can be added to this object.

[PolicyDefinitionReferenceId <List<String>>]: The policy definition reference ID list when the associated policy assignment is an assignment of a policy set definition.

[ResourceSelector <List<IResourceSelector>>]: The resource selector list to filter policies by resource properties.

[Name <String>]: The name of the resource selector.

[Selector <List<ISelector>>]: The list of the selector expressions.

[In <List<String>>]: The list of values to filter in.

[Kind <String>]: The selector kind.

[NotIn <List<String>>]: The list of values to filter out.

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

```
PS C:\>$ResourceGroup = Get-AzResourceGroup -Name 'ResourceGroup11'
```

```
$PolicyExemption = Get-AzPolicyExemption -Name 'PolicyExemption07' -Scope $ResourceGroup.ResourceId  
Update-AzPolicyExemption -Id $PolicyExemption.ResourceId -DisplayName 'Exempt VM creation limit'
```

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

```
PS C:\>$NextMonth = (Get-Date).AddMonths(1)
```

```
$PolicyExemption = Get-AzPolicyExemption -Name 'PolicyExemption07'  
Update-AzPolicyExemption -Id $PolicyExemption.ResourceId -ExpiresOn $NextMonth
```

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

```
PS C:\>$PolicyExemption = Get-AzPolicyExemption -Name 'PolicyExemption07'
```

```
Update-AzPolicyExemption -Id $PolicyExemption.ResourceId -ClearExpiration
```

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

```
PS C:\>$PolicyExemption = Get-AzPolicyExemption -Name 'PolicyExemption07'
```

```
Update-AzPolicyExemption -Id $PolicyExemption.ResourceId -ExemptionCategory Mitigated
```

----- EXAMPLE 5 -----

```
PS C:\>$PolicyExemption = Get-AzPolicyExemption -Name 'PolicyExemption07'
```

```
Set-AzPolicyExemption -Id $PolicyExemption.ResourceId -ClearExpiration
```

RELATED LINKS

<https://learn.microsoft.com/powershell/module/az.resources/update-azpolicyexemption>

