
**Full credit is given to all the above companies including the Operating System that this PDF file was generated!**

## *Windows PowerShell Get-Help on Cmdlet 'New-AzPolicyAssignment'*

***PS:\>Get-HELP New-AzPolicyAssignment -Full***

NAME

New-AzPolicyAssignment

SYNOPSIS

Creates or updates a policy assignment.

SYNTAX

New-AzPolicyAssignment -Name <String> [-Scope <String>] [-NotScope <String[]>] [-DisplayName <String>] [-Description <String>] [-Metadata <String>] [-EnforcementMode <String>] [-IdentityType <String>] [-IdentityId <String>] [-Location <String>] [-NonComplianceMessage <PSObject[]>] [-BackwardCompatible] [-DefaultProfile <PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy <Uri>] [-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]

New-AzPolicyAssignment -Name <String> [-Scope <String>] [-NotScope <String[]>] [-DisplayName <String>] [-Description <String>] [-Metadata <String>] [-EnforcementMode <String>] [-IdentityType <String>] [-IdentityId <String>] [-Location <String>] [-NonComplianceMessage <PSObject[]>] [-BackwardCompatible] [-PolicyDefinition

<PSObject>] -PolicyParameterObject <Hashtable> [-DefaultProfile <PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>]

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

New-AzPolicyAssignment -Name <String> [-Scope <String>] [-NotScope <String[]>] [-DisplayName <String>] [-Description <String>] [-Metadata <String>] [-EnforcementMode

<String>] [-IdentityType <String>] [-IdentityId <String>] [-Location <String>] [-NonComplianceMessage <PSObject[]>] [-BackwardCompatible] [-PolicyDefinition

<PSObject>] -PolicyParameter <String> [-DefaultProfile <PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy

<Uri>] [-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]

New-AzPolicyAssignment -Name <String> [-Scope <String>] [-NotScope <String[]>] [-DisplayName <String>] [-Description <String>] [-Metadata <String>] [-EnforcementMode

<String>] [-IdentityType <String>] [-IdentityId <String>] [-Location <String>] [-NonComplianceMessage <PSObject[]>] [-BackwardCompatible] -PolicyDefinition <PSObject>

[-DefaultProfile <PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>] [-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy <Uri>] [-ProxyCredential

<PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]

DESCRIPTION

The **New-AzPolicyAssignment** cmdlet creates or updates a policy assignment with the given scope and name.

Policy assignments apply to all resources contained within their scope.

For example, when you assign a policy at resource group scope, that policy applies to all resources in the group.

PARAMETERS

-Name <String>

The name of the policy assignment.

Required? true

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-Scope <String>

The scope of the policy assignment.

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

Required? false

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-NotScope <String[]>

The policy's excluded scopes.

Required? false

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

-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

-EnforcementMode <String>

The policy assignment enforcement mode.

Possible values are Default and DoNotEnforce.

Required? false

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-IdentityType <String>

The identity type.

This is the only required field when adding a system or user assigned identity to a resource.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-IdentityId <String>

The user identity associated with the policy.

The user identity dictionary key references will be ARM resource ids in the form:

'/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.ManagedIdentity/userAssignedIdentities/{identityName}'.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Location <String>

The location of the policy assignment.

Only required when utilizing managed identity.

Required? false

Position? named

Default value

Accept pipeline input? true (ByPropertyName)

Accept wildcard characters? false

-NonComplianceMessage <PSObject[]>

The messages that describe why a resource is non-compliant with the policy.

To construct, see NOTES section for NONCOMPLIANCEMESSAGE properties and create a hash table.

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

-PolicyDefinition <PSObject>

Accept policy definition or policy set definition object

Required? false

Position? named

Default value

Accept pipeline input? true (ByValue)

Accept wildcard characters? false

-PolicyParameter <String>

The parameter values for the assigned policy rule.

The keys are the parameter names.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-PolicyParameterObject <Hashtable>

The parameter values for the assigned policy rule.

The keys are the parameter names.

Required? true

Position? named

Default value

Accept pipeline input? false

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).

INPUTS

System.Management.Automation.PSObject

System.Management.Automation.PSObject[]

System.String

System.String[]

OUTPUTS

Microsoft.Azure.PowerShell.Cmdlets.Policy.Models.IPolicyAssignment

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

PS C:\>$Subscription = Get-AzSubscription -SubscriptionName 'Subscription01'

$Policy = Get-AzPolicyDefinition -Name 'VirtualMachinePolicy'

New-AzPolicyAssignment -Name 'VirtualMachinePolicyAssignment' -PolicyDefinition $Policy -Scope "/subscriptions/$($Subscription.Id)"

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

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

```
$Policy = Get-AzPolicyDefinition -Name 'VirtualMachinePolicy'
        New-AzPolicyAssignment -Name 'VirtualMachinePolicyAssignment' -PolicyDefinition $Policy -Scope
$ResourceGroup.ResourceId
```

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

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

$Policy = Get-AzPolicyDefinition -BuiltIn | Where-Object {$_.DisplayName -eq 'Allowed locations'}
$Locations = Get-AzLocation | Where-Object displayname -like '*east*'
$AllowedLocations = @{'listOfAllowedLocations'=($Locations.location)}
        New-AzPolicyAssignment -Name 'RestrictLocationPolicyAssignment' -PolicyDefinition $Policy -Scope
$ResourceGroup.ResourceId -PolicyParameterObject $AllowedLocations
```

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

```
PS C:\>{

"listOfAllowedLocations":  {
    "value": [
      "westus",
      "westeurope",
      "japanwest"
    ]
  }
}
```

```
$ResourceGroup = Get-AzResourceGroup -Name 'ResourceGroup11'
$Policy = Get-AzPolicyDefinition -BuiltIn | Where-Object {$_.DisplayName -eq 'Allowed locations'}
New-AzPolicyAssignment -Name 'RestrictLocationPolicyAssignment' -PolicyDefinition $Policy -Scope
$ResourceGroup.ResourceId -PolicyParameter .\AllowedLocations.json
```

```
---------- EXAMPLE 5 ----------
PS C:\>$ResourceGroup = Get-AzResourceGroup -Name 'ResourceGroup11'
$Policy = Get-AzPolicyDefinition -Name 'VirtualMachinePolicy'
New-AzPolicyAssignment -Name 'VirtualMachinePolicyAssignment' -PolicyDefinition $Policy -Scope
$ResourceGroup.ResourceId -Location 'eastus' -IdentityType
'SystemAssigned'
```

```
---------- EXAMPLE 6 ----------
PS C:\>$ResourceGroup = Get-AzResourceGroup -Name 'ResourceGroup11'
$Policy = Get-AzPolicyDefinition -Name 'VirtualMachinePolicy'
$UserAssignedIdentity = Get-AzUserAssignedIdentity -ResourceGroupName 'ResourceGroup1' -Name
'UserAssignedIdentity1'
New-AzPolicyAssignment -Name 'VirtualMachinePolicyAssignment' -PolicyDefinition $Policy -Scope
$ResourceGroup.ResourceId -Location 'eastus' -IdentityType
'UserAssigned' -IdentityId $UserAssignedIdentity.Id
```

-------------------------- EXAMPLE 7 --------------------------

PS C:\>$Subscription = Get-AzSubscription -SubscriptionName 'Subscription01'

$Policy = Get-AzPolicyDefinition -Name 'VirtualMachinePolicy'

New-AzPolicyAssignment -Name 'VirtualMachinePolicyAssignment' -PolicyDefinition $Policy -Scope "/subscriptions/$($Subscription.Id)" -EnforcementMode DoNotEnforce

-------------------------- EXAMPLE 8 --------------------------

PS C:\>$PolicySet = Get-AzPolicySetDefinition -Name 'VirtualMachinePolicySet'

$NonComplianceMessages = @(@{Message="Only DsV2 SKUs are allowed."; PolicyDefinitionReferenceId="DefRef1"}, @{Message="Virtual machines must follow cost management

best practices."})

New-AzPolicyAssignment -Name 'VirtualMachinePolicyAssignment' -PolicySetDefinition $PolicySet -NonComplianceMessage $NonComplianceMessages

-------------------------- EXAMPLE 9 --------------------------

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

$Policy = Get-AzPolicyDefinition -BuiltIn | Where-Object {$_.Properties.DisplayName -eq 'Allowed locations'}

$Locations = Get-AzLocation | Where-Object displayname -like '*east*'

$AllowedLocations = @{'listOfAllowedLocations'=($Locations.location)}

New-AzPolicyAssignment -Name 'RestrictLocationPolicyAssignment' -PolicyDefinition $Policy -Scope

$ResourceGroup.ResourceId -PolicyParameterObject $AllowedLocations

---------- EXAMPLE 10 ----------

PS C:\>{

"listOfAllowedLocations":  {
    "value": [
      "westus",
      "westeurope",
      "japanwest"
    ]
  }
}

$ResourceGroup = Get-AzResourceGroup -Name 'ResourceGroup11'

$Policy = Get-AzPolicyDefinition -BuiltIn | Where-Object {$_.Properties.DisplayName -eq 'Allowed locations'}

New-AzPolicyAssignment -Name 'RestrictLocationPolicyAssignment' -PolicyDefinition $Policy -Scope $ResourceGroup.ResourceId -PolicyParameter .\AllowedLocations.json

RELATED LINKS

https://learn.microsoft.com/powershell/module/az.resources/new-azpolicyassignment