



Windows PowerShell Get-Help on Cmdlet 'Set-AzPolicyAttestation'

PS:\>Get-HELP Set-AzPolicyAttestation -Full

NAME

Set-AzPolicyAttestation

SYNOPSIS

Modifies a policy attestation.

SYNTAX

```
Set-AzPolicyAttestation [-AssessmentDate <System.Nullable`1[System.DateTime]>] [-Comment <System.String>]
[-ComplianceState <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Evidence
<Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestationEvidence[]>] [-ExpiresOn
<System.Nullable`1[System.DateTime]>] -InputObject
<Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestation> [-Metadata <System.String>] [-Owner
<System.String>] [-PolicyAssignmentId <System.String>]
[-PolicyDefinitionReferenceId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-AzPolicyAttestation [-AssessmentDate <System.Nullable`1[System.DateTime]>] [-Comment <System.String>]
[-ComplianceState <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Evidence
```

```

        <Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestationEvidence[]>    [-ExpiresOn
<System.Nullable`1[System.DateTime]>] [-Metadata <System.String>]
    -Name <System.String> [-Owner <System.String>] [-PolicyAssignmentId <System.String>] [-PolicyDefinitionReferenceId
<System.String>] [-ResourceGroupName
    <System.String>] [-Scope <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzPolicyAttestation [-AssessmentDate <System.Nullable`1[System.DateTime]>] [-Comment <System.String>]
[-ComplianceState <System.String>] [-DefaultProfile
    <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Evidence
        <Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestationEvidence[]>    [-ExpiresOn
<System.Nullable`1[System.DateTime]>] [-Metadata <System.String>]
    [-Owner <System.String>] [-PolicyAssignmentId <System.String>] [-PolicyDefinitionReferenceId <System.String>]
-ResourceId <System.String> [-Confirm] [-WhatIf]
    [<CommonParameters>]

```

DESCRIPTION

The Set-AzPolicyAttestation cmdlet modifies a policy attestation. Specify an attestation by Id or by Name and scope, or via piping.

> Note: >An existing policy attestation's `policyAssignmentId` or `policyDefinitionReferenceId` cannot be modified.

PARAMETERS

-AssessmentDate <System.Nullable`1[System.DateTime]>

The time the evidence of an attestation was assessed.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Comment <System.String>

Comments describing why this attestation was created.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-ComplianceState <System.String>

The Compliance State of the resource. E.g. 'Compliant', 'NonCompliant', 'Unknown'

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>

The credentials, account, tenant, and subscription used for communication with Azure.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Evidence <Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestationEvidence[]>

The evidence supporting the compliance state set in this attestation.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-ExpiresOn <System.Nullable`1[System.DateTime]>

The time the compliance state set in the attestation should expire.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-InputObject <Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestation>

The Attestation object.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-Metadata <System.String>

Additional metadata for the attestation.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Name <System.String>

Resource name.

Required? true
Position? named
Default value None
Accept pipeline input? True (ByPropertyName)
Accept wildcard characters? false

-Owner <System.String>

The person responsible for setting the state of the resource. This value is typically a Microsoft Entra object ID.

Required? false
Position? named
Default value None
Accept pipeline input? True (ByPropertyName)
Accept wildcard characters? false

-PolicyAssignmentId <System.String>

Policy assignment ID. E.g.

'/subscriptions/{subscriptionId}/providers/Microsoft.Authorization/policyAssignments/{assignmentName}'.

Required? false
Position? named
Default value None
Accept pipeline input? True (ByPropertyName)
Accept wildcard characters? false

-PolicyDefinitionReferenceId <System.String>

The policy definition reference ID of the individual definition. Required when the policy assignment assigns a policy set definition.

Required? false
Position? named
Default value None
Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-ResourceGroupName <System.String>

Resource group name.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-ResourceId <System.String>

Resource ID.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Scope <System.String>

Scope of the resource. E.g. '/subscriptions/{subscriptionId}/resourceGroups/{rgName}'.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

Required? false

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

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false
Position? named
Default value False
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.String

Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestation

System.Nullable`1[[System.DateTime, System.Private.CoreLib, Version=7.0.0.0, Culture=neutral,
PublicKeyToken=7cec85d7bea7798e]]

Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestationEvidence[]

System.Object

OUTPUTS

Microsoft.Azure.Commands.PolicyInsights.Models.Attestations.PSAttestation

NOTES

----- Example 1: Update an attestation by name -----

```
Set-AzContext -Subscription "d1acb22b-c876-44f7-b08e-3fcf9f6767f4"
```

```
# Update the existing attestation by resource name at subscription scope (default)
```

```
$comment = "Setting the state to non compliant"
```

```
$attestationName = "attestation-subscription"
```

```
$policyAssignmentId =
```

```
"/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/providers/Microsoft.Authorization/policyAssignments/PSAttestationSubAssignment"
```

```
Set-AzPolicyAttestation -PolicyAssignmentId $policyAssignmentId -Name $attestationName -ComplianceState "NonCompliant" -Comment $comment
```

```
Id : /subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/providers/microsoft.policyinsights/attestations/attestation-subscription
```

```
Name : attestation-subscription
```

```
Type : Microsoft.PolicyInsights/attestations
```

```
PolicyAssignmentId :
```

```
/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/providers/microsoft.authorization/policyassignme
```

nts/psattestationsubassignment

PolicyDefinitionReferenceId :

ComplianceState : NonCompliant

ExpiresOn :

Owner :

Comment : Setting the state to non compliant

Evidence :

ProvisioningState : Succeeded

LastComplianceStateChangeAt : 1/27/2023 4:00:04 PM

AssessmentDate :

Metadata :

SystemData :

The command here sets the compliance state and adds a comment to an existing attestation with name 'attestation-subscription' in the subscription with id

'd1acb22b-c876-44f7-b08e-3fcf9f6767f4'

----- Example 2: Update an attestation by ResourceId -----

Get an attestation

\$rgName = "ps-attestation-test-rg"

\$attestationName = "attestation-RG"

\$attestation = Get-AzPolicyAttestation -ResourceGroupName \$rgName -Name \$attestationName

Update the existing attestation by resource ID at RG

\$expiresOn = [System.DateTime]::UtcNow.AddYears(1)

Set-AzPolicyAttestation -Id \$attestation.Id -ExpiresOn \$expiresOn

Id

:

/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/resourcegroups/ps-attestation-test-rg/providers/

microsoft.policyinsights/attestations/attestation-rg

Name : attestation-rg

```

Type : Microsoft.PolicyInsights/attestations

PolicyAssignmentId :

/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/providers/microsoft.authorization/policyassignmentrgassignment

PolicyDefinitionReferenceId :

ComplianceState :

ExpiresOn : 1/27/2024 4:04:24 PM

Owner :

Comment :

Evidence :

ProvisioningState : Succeeded

LastComplianceStateChangeAt : 1/27/2023 4:04:11 PM

AssessmentDate :

Metadata :

SystemData :

```

The first command gets an existing attestation at the resource group 'ps-attestation-test-rg' with the name 'attestation-RG'.

The final command updates the expiry time of the policy attestation by the ResourceId property of the existing attestation.

----- Example 3: Update an attestation by input object -----

```

# Get an attestation

$attestationName = "attestation-resource"

$scope =

"/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/resourceGroups/ps-attestation-test-rg/providers/Microsoft.Network/networkSecurityGroups/pstests0"

$attestation = Get-AzPolicyAttestation -Name $attestationName -Scope $scope

# Update attestation by input object

$newOwner = "Test Owner 2"

```

\$attestation | Set-AzPolicyAttestation -Owner \$newOwner

```
Id :  
/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/resourcegroups/ps-attestation-test-rg/providers/  
microsoft.network/networksecuritygroups/pstests0/providers/microsoft.policyinsights/attestations/att  
estation-resource  
Name : attestation-resource  
Type : Microsoft.PolicyInsights/attestations  
PolicyAssignmentId :  
/subscriptions/d1acb22b-c876-44f7-b08e-3fcf9f6767f4/providers/microsoft.authorization/policyassignme  
nts/psattestationresourceassignment  
PolicyDefinitionReferenceId :  
ComplianceState : NonCompliant  
ExpiresOn :  
Owner : Test Owner 2  
Comment :  
Evidence :  
ProvisioningState : Succeeded  
LastComplianceStateChangeAt : 1/27/2023 2:38:17 AM  
AssessmentDate :  
Metadata :  
SystemData :
```

The first command gets an existing attestation with name 'attestation-resource' for the given resource using its resource id as the scope

The final command updates the owner of the policy attestation by using piping.

RELATED LINKS

Online Version: <https://learn.microsoft.com/powershell/module/az.policyinsights/set-azpolicyattestation>

New-AzPolicyAttestation

Remove-AzPolicyAttestation

Get-AzPolicyAttestation