



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

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

NAME

Update-AzSentinelAutomationRule

SYNOPSIS

Creates or updates the automation rule.

SYNTAX

```
Update-AzSentinelAutomationRule -Id <String> -ResourceGroupName <String> [-SubscriptionId <String>]
-WorkspaceName <String> [-Action <IAutomationRuleAction[]>]
[-DisplayName <String>] [-Order <Int32>] [-TriggeringLogicCondition <IAutomationRuleCondition[]>]
[-TriggeringLogicExpirationTimeUtc <DateTime>]
[-TriggeringLogicsEnabled] [-DefaultProfile <PSObject>] [-Break] [-HttpPipelineAppend <SendAsyncStep[]>]
[-HttpPipelinePrepend <SendAsyncStep[]>] [-Proxy <Uri>]
[-ProxyCredential <PSCredential>] [-ProxyUseDefaultCredentials] [-WhatIf] [-Confirm] [<CommonParameters>]
```

```
Update-AzSentinelAutomationRule -InputObject <ISecurityInsightsIdentity> [-Action <IAutomationRuleAction[]>]
[-DisplayName <String>] [-Order <Int32>]
[-TriggeringLogicCondition <IAutomationRuleCondition[]>] [-TriggeringLogicExpirationTimeUtc <DateTime>]
[-TriggeringLogicsEnabled] [-DefaultProfile <PSObject>]
```

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

DESCRIPTION

Creates or updates the automation rule.

PARAMETERS

-Id <String>

Automation rule ID

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-ResourceGroupName <String>

The name of the resource group.

The name is case insensitive.

Required? true

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-SubscriptionId <String>

The ID of the target subscription.

Required? false

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

-WorkspaceName <String>

The name of the workspace.

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

-InputObject <ISecurityInsightsIdentity>

Identity Parameter

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

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

-Action <IAutomationRuleAction[]>

The actions to execute when the automation rule is triggered

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

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

-DisplayName <String>

The display name of the automation rule

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-Order <Int32>

The order of execution of the automation rule

Required? false

Position? named

Default value 0

Accept pipeline input? false

Accept wildcard characters? false

-TriggeringLogicCondition <IAutomationRuleCondition[]>

The conditions to evaluate to determine if the automation rule should be triggered on a given object

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-TriggeringLogicExpirationTimeUtc <DateTime>

Determines when the automation rule should automatically expire and be disabled.

Required? false

Position? named

Default value

Accept pipeline input? false

Accept wildcard characters? false

-TriggeringLogicIsEnabled [<SwitchParameter>]

Determines whether the automation rule is enabled or disabled.

Required? false

Position? named

Default value False

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

Microsoft.Azure.PowerShell.Cmdlets.SecurityInsights.Models.ISecurityInsightsIdentity

OUTPUTS

Microsoft.Azure.PowerShell.Cmdlets.SecurityInsights.Models.Api20210901Preview.IAutomationRule

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.

ACTION <IAutomationRuleAction[]>: The actions to execute when the automation rule is triggered

ActionType <AutomationRuleActionType>: The type of the automation rule action

Order <Int32>: The order of execution of the automation rule action

INPUTOBJECT <ISecurityInsightsIdentity>: Identity Parameter

[ActionId <String>]: Action ID

[AlertRuleTemplateId <String>]: Alert rule template ID

[AutomationRuleId <String>]: Automation rule ID

[BookmarkId <String>]: Bookmark ID

[ConsentId <String>]: consent ID

[DataConnectorId <String>]: Connector ID

[EntityId <String>]: entity ID

[EntityQueryId <String>]: entity query ID

[EntityQueryTemplateId <String>]: entity query template ID

[Id <String>]: Resource identity path

[IncidentCommentId <String>]: Incident comment ID

[IncidentId <String>]: Incident ID

[MetadataName <String>]: The Metadata name.

[Name <String>]: Threat intelligence indicator name field.

[RelationName <String>]: Relation Name

[ResourceGroupName <String>]: The name of the resource group. The name is case insensitive.

[RuleId <String>]: Alert rule ID

[SentinelOnboardingStateName <String>]: The Sentinel onboarding state name. Supports - default

[SettingsName <String>]: The setting name. Supports - Anomalies, EyesOn, EntityAnalytics, Ueba

[SourceControlId <String>]: Source control Id

[SubscriptionId <String>]: The ID of the target subscription.

[WorkspaceName <String>]: The name of the workspace.

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

```
PS C:\>$LogicAppResourceId = Get-AzLogicApp -ResourceGroupName "myResourceGroup" -Name
"Reset-AADPassword"

$automationRuleAction =
[Microsoft.Azure.PowerShell.Cmdlets.SecurityInsights.Models.Api20210901Preview.AutomationRuleRunPlaybookAction]::n
ew()

$automationRuleAction.Order = 1

$automationRuleAction.ActionType = "RunPlaybook"

$automationRuleAction.ActionConfigurationLogicAppResourceId = ($LogicAppResourceId.Id)

$automationRuleAction.ActionConfigurationTenantId = (Get-AzContext).Tenant.Id

Update-AzSentinelAutomationRule -ResourceGroupName "myResourceGroup" -WorkspaceName "myWorkspaceName"
-Id ((New-Guid).Guid) -Action $automationRuleAction -DisplayName
"Run Playbook to reset AAD password" -Order 2 -TriggeringLogicIsEnabled
```

