

---

# AWS Systems Manager Automation runbook reference

## User Guide



## **AWS Systems Manager Automation runbook reference: User Guide**

Copyright © 2022 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

## Table of Contents

Automation runbook reference .....	1
View runbook content .....	2
API Gateway .....	2
AWSConfigRemediation-DeleteAPIGatewayStage .....	3
AWSConfigRemediation-EnableAPIGatewayTracing .....	4
AWSConfigRemediation-UpdateAPIGatewayMethodCaching .....	5
AWS CloudFormation .....	6
AWS-DeleteCloudFormationStack .....	6
AWS-RunCfnLint .....	7
AWSSupport-TroubleshootCFNCustomResource .....	8
AWS-UpdateCloudFormationStack .....	9
CloudFront .....	10
AWSConfigRemediation-EnableCloudFrontDefaultRootObject .....	11
AWSConfigRemediation-EnableCloudFrontAccessLogs .....	12
AWSConfigRemediation-EnableCloudFrontOriginAccessIdentity .....	13
AWSConfigRemediation-EnableCloudFrontOriginFailover .....	14
AWSConfigRemediation-EnableCloudFrontViewerPolicyHTTPS .....	15
CloudTrail .....	16
AWSConfigRemediation-CreateCloudTrailMultiRegionTrail .....	17
AWS-EnableCloudTrail .....	18
AWSConfigRemediation-EnableCloudTrailEncryptionWithKMS .....	19
AWSConfigRemediation-EnableCloudTrailLogFileValidation .....	20
CloudWatch .....	21
AWS-ConfigureCloudWatchOnEC2Instance .....	21
CodeBuild .....	22
AWSConfigRemediation-ConfigureCodeBuildProjectWithKMSSCMK .....	22
AWSConfigRemediation-DeleteAccessKeysFromCodeBuildProject .....	24
AWS CodeDeploy .....	25
AWSSupport-TroubleshootCodeDeploy .....	25
AWS Config .....	26
AWSSupport-SetupConfig .....	26
AWS Directory Service .....	28
AWS-CreateDSManagementInstance .....	28
AWSSupport-TroubleshootDirectoryTrust .....	31
DynamoDB .....	34
AWS-CreateDynamoDBBackup .....	34
AWS-DeleteDynamoDbBackup .....	35
AWSConfigRemediation-DeleteDynamoDbTable .....	35
AWS-DeleteDynamoDbTableBackups .....	36
AWSConfigRemediation-EnableEncryptionOnDynamoDBTable .....	37
AWSConfigRemediation-EnablePITRForDynamoDbTable .....	38
Amazon EBS .....	39
AWS-AttachEBSVolume .....	40
AWSSupport-CalculateEBSPerformanceMetrics .....	41
AWS-CopySnapshot .....	42
AWS-CreateSnapshot .....	43
AWS-DeleteEbsVolumeSnapshots .....	44
AWS-DeleteSnapshot .....	45
AWSConfigRemediation-DeleteUnusedEBSVolume .....	46
AWS-DetachEBSVolume .....	47
AWSConfigRemediation-EnableEbsEncryptionByDefault .....	48
AWSSupport-ModifyEBSSnapshotPermission .....	48
AWSConfigRemediation-ModifyEBSVolumeType .....	50
Amazon EC2 .....	51

AWS-ASGEnterStandby .....	52
AWS-ASGExitStandby .....	53
AWS-CreateImage .....	54
AWS-DeleteImage .....	55
AWSEC2-CloneInstanceAndUpgradeSQLServer .....	56
AWSEC2-CloneInstanceAndUpgradeWindows .....	58
AWSEC2-ConfigureSTIG .....	61
AWSEC2-PatchLoadBalancerInstance .....	69
AWSEC2-SQLServerDBRestore .....	70
AWSSupport-ActivateWindowsWithAmazonLicense .....	73
AWSPremiumSupport-ChangeInstanceTypeIntelToAMD .....	75
AWSSupport-CheckXenToNitroMigrationRequirements .....	79
AWSSupport-ConfigureEC2Metadata .....	81
AWSSupport-CopyEC2Instance .....	83
AWSSupport-ExecuteEC2Rescue .....	87
AWSSupport-ListEC2Resources .....	89
AWSSupport-ManageRDPSettings .....	90
AWSSupport-ManageWindowsService .....	92
AWSSupport-MigrateEC2ClassicToVPC .....	93
AWSSupport-MigrateXenToNitroLinux .....	98
AWSSupport-ResetAccess .....	105
AWSSupport-RestoreEC2InstanceFromSnapshot .....	107
AWSSupport-SendLogBundleToS3Bucket .....	110
AWSSupport-StartEC2RescueWorkflow .....	111
AWSPremiumSupport-TroubleshootEC2DiskUsage .....	118
AWSSupport-TroubleshootRDP .....	121
AWSSupport-TroubleshootSSH .....	125
AWSSupport-TroubleshootSUSERegistration .....	127
AWSSupport-UpgradeWindowsAWSDrivers .....	128
Amazon ECS .....	131
AWSSupport-CollectECSInstanceLogs .....	131
AWS-InstallECSContainerAgent .....	133
AWSSupport-TroubleshootECSContainerInstance .....	134
AWS-UpdateECSContainerAgent .....	135
Amazon EFS .....	136
AWSSupport-CheckAndMountEFS .....	137
Amazon EKS .....	139
AWSSupport-CollectEKSInstanceLogs .....	139
AWS-DeleteEKSCluster .....	140
AWSPremiumSupport-TroubleshootEKSCluster .....	142
AWSSupport-TroubleshootEKSEWorkerNode .....	145
AWS-UpdateEKSMangedNodegroupVersion .....	147
Elastic Beanstalk .....	148
AWSSupport-CollectElasticBeanstalkLogs .....	148
AWSConfigRemediation-EnableElasticBeanstalkEnvironmentLogStreaming .....	150
AWSConfigRemediation-EnableBeanstalkEnvironmentNotifications .....	151
Elastic Load Balancing .....	153
AWSConfigRemediation-DropInvalidHeadersForALB .....	153
AWSConfigRemediation-EnableCLBCrossZoneLoadBalancing .....	154
AWSConfigRemediation-EnableELBDeletionProtection .....	155
AWSConfigRemediation-EnableLoggingForALBAndCLB .....	156
AWSSupport-TroubleshootCLBConnectivity .....	157
AWSConfigRemediation-EnableNLBCrossZoneLoadBalancing .....	159
Amazon EMR .....	160
AWSSupport-AnalyzeEMRLogs .....	160
Amazon OpenSearch Service .....	164
AWSConfigRemediation-DeleteOpenSearchDomain .....	164

AWSConfigRemediation-EnforceHTTPSOnOpenSearchDomain .....	165
AWSConfigRemediation-UpdateOpenSearchDomainSecurityGroups .....	166
EventBridge .....	167
AWS-AddOpsItemDedupStringToEventBridgeRule .....	168
AWS-DisableEventBridgeRule .....	169
GuardDuty .....	170
AWSConfigRemediation-CreateGuardDutyDetector .....	170
IAM .....	171
AWS-AttachIAMToInstance .....	171
AWSConfigRemediation-DeleteIAMRole .....	173
AWSConfigRemediation-DeleteIAMUser .....	174
AWSConfigRemediation-DeleteUnusedIAMGroup .....	175
AWSConfigRemediation-DeleteUnusedIAMPolicy .....	176
AWSConfigRemediation-DetachIAMPolicy .....	177
AWSConfigRemediation-EnableAccountAccessAnalyzer .....	178
AWSSupport-GrantPermissionsToIAMUser .....	179
AWSConfigRemediation-RemoveUserPolicies .....	183
AWSConfigRemediation-ReplaceIAMInlinePolicy .....	184
AWSConfigRemediation-RevokeUnusedIAMUserCredentials .....	185
AWSConfigRemediation-SetIAMPASSWORDPolicy .....	187
AWS KMS .....	189
AWSConfigRemediation-CancelKeyDeletion .....	189
AWSConfigRemediation-EnableKeyRotation .....	190
Lambda .....	191
AWSConfigRemediation-ConfigureLambdaFunctionXRayTracing .....	191
AWSConfigRemediation-DeleteLambdaFunction .....	192
AWSConfigRemediation-EncryptLambdaEnvironmentVariablesWithCMK .....	193
AWSConfigRemediation-MoveLambdaToVPC .....	194
AWSSupport-RemediateLambdaS3Event .....	196
AWSSupport-TroubleshootLambdaInternetAccess .....	197
AWSSupport-TroubleshootLambdaS3Event .....	200
Amazon RDS .....	201
AWS-CreateRdsSnapshot .....	202
AWSConfigRemediation-DeleteRDSCluster .....	203
AWSConfigRemediation-DeleteRDSClusterSnapshot .....	204
AWSConfigRemediation-DeleteRDSInstance .....	205
AWSConfigRemediation-DeleteRDSInstanceSnapshot .....	206
AWSConfigRemediation-DisablePublicAccessToRDSInstance .....	207
AWSConfigRemediation-EnableCopyTagsToSnapshotOnRDSCluster .....	208
AWSConfigRemediation-EnableCopyTagsToSnapshotOnRDSDBInstance .....	209
AWSConfigRemediation-EnableEnhancedMonitoringOnRDSInstance .....	211
AWSConfigRemediation-EnableMinorVersionUpgradeOnRDS .....	212
AWSConfigRemediation-EnableMultiAZOnRDSInstance .....	213
AWSConfigRemediation-EnablePerformanceInsightsOnRDSInstance .....	214
AWSConfigRemediation-EnableRDSClusterDeletionProtection .....	216
AWSConfigRemediation-EnableRDSInstanceBackup .....	217
AWSConfigRemediation-EnableRDSInstanceDeletionProtection .....	218
AWSConfigRemediation-ModifyRDSInstancePortNumber .....	220
AWSSupport-ModifyRDSSnapshotPermission .....	221
AWSPremiumSupport-PostgreSQLWorkloadReview .....	222
AWS-RebootRdsInstance .....	233
AWSSupport-ShareRDSSnapshot .....	234
AWS-StartRdsInstance .....	236
AWSSupport-TroubleshootConnectivityToRDS .....	237
Amazon Redshift .....	238
AWSConfigRemediation-DeleteRedshiftCluster .....	239
AWSConfigRemediation-DisablePublicAccessToRedshiftCluster .....	240

AWSConfigRemediation-EnableRedshiftClusterAuditLogging .....	241
AWSConfigRemediation-EnableRedshiftClusterAutomatedSnapshot .....	242
AWSConfigRemediation-EnableRedshiftClusterEncryption .....	243
AWSConfigRemediation-EnableRedshiftClusterEnhancedVPCRouting .....	244
AWSConfigRemediation-EnforceSSLOnlyConnectionsToRedshiftCluster .....	245
AWSConfigRemediation-ModifyRedshiftClusterMaintenanceSettings .....	246
AWSConfigRemediation-ModifyRedshiftClusterNodeType .....	248
Amazon S3 .....	249
AWS-ConfigureS3BucketLogging .....	249
AWS-ConfigureS3BucketVersioning .....	251
AWSConfigRemediation-ConfigureS3BucketPublicAccessBlock .....	252
AWSConfigRemediation-ConfigureS3PublicAccessBlock .....	253
AWS-DisableS3BucketPublicReadWrite .....	255
AWS-EnableS3BucketEncryption .....	255
AWSConfigRemediation-RemovePrincipalStarFromS3BucketPolicy .....	256
AWSConfigRemediation-RestrictBucketSSLRequestsOnly .....	257
AWSSupport-TroubleshootS3PublicRead .....	258
Secrets Manager .....	262
AWSConfigRemediation-DeleteSecret .....	262
AWSConfigRemediation-RotateSecret .....	263
Security Hub .....	264
AWSConfigRemediation-EnableSecurityHub .....	265
Amazon SNS .....	266
AWSConfigRemediation-EncryptSNSTopic .....	266
AWS-PublishSNSNotification .....	267
Systems Manager .....	268
AWS-BulkEditOpsItems .....	268
AWS-BulkResolveOpsItems .....	270
AWS-CreateManagedLinuxInstance .....	272
AWS-CreateManagedWindowsInstance .....	273
AWSConfigRemediation-EnableCWLoggingForSessionManager .....	275
AWS-ExportOpsDataToS3 .....	276
AWS-ExportPatchReportToS3 .....	277
AWS-SetupInventory .....	279
AWS-SetupManagedInstance .....	281
AWS-SetupManagedRoleOnEC2Instance .....	282
AWSSupport-TroubleshootManagedInstance .....	283
Third-party .....	285
AWS-CreateJiraIssue .....	285
AWS-CreateServiceNowIncident .....	287
AWS-RunPacker .....	288
Amazon VPC .....	289
AWSSupport-ConfigureDNSQueryLogging .....	290
AWSSupport-ConfigureTrafficMirroring .....	292
AWSSupport-ConnectivityTroubleshooter .....	294
AWSConfigRemediation-DeleteEgressOnlyInternetGateway .....	296
AWSConfigRemediation-DeleteUnusedENI .....	297
AWSConfigRemediation-DeleteUnusedSecurityGroup .....	298
AWSConfigRemediation-DeleteUnusedVPCNetworkACL .....	299
AWSConfigRemediation-DeleteVPCFlowLog .....	300
AWSConfigRemediation-DetachAndDeleteInternetGateway .....	301
AWSConfigRemediation-DetachAndDeleteVirtualPrivateGateway .....	302
AWS-DisablePublicAccessForSecurityGroup .....	304
AWSConfigRemediation-DisableSubnetAutoAssignPublicIP .....	305
AWSSupport-EnableVPCFlowLogs .....	306
AWSConfigRemediation-EnableVPCFlowLogsToCloudWatch .....	308
AWSConfigRemediation-EnableVPCFlowLogsToS3Bucket .....	310

AWS-ReleaseElasticIP .....	311
AWSConfigRemediation-RemoveUnrestrictedSourceIngressRules .....	312
AWSConfigRemediation-RemoveVPCDefaultSecurityGroupRules .....	313
AWSSupport-SetupIPMonitoringFromVPC .....	314
AWSSupport-TerminateIPMonitoringFromVPC .....	321
AWS WAF .....	323
AWSConfigRemediation-EnableWAFClassicLogging .....	323
AWSConfigRemediation-EnableWAFClassicRegionalLogging .....	325
AWSConfigRemediation-EnableWAFV2Logging .....	326
Amazon WorkSpaces .....	327
AWSSupport-RecoverWorkSpace .....	327
X-Ray .....	330
AWSConfigRemediation-UpdateXRayKMSKey .....	330

# Systems Manager Automation runbook reference

To help you get started quickly, AWS Systems Manager provides predefined runbooks. These runbooks are maintained by Amazon Web Services, AWS Support, and AWS Config. The runbook reference describes each of the predefined runbooks provided by Systems Manager, AWS Support, and AWS Config.

## Important

If you run an automation workflow that invokes other services by using an AWS Identity and Access Management (IAM) service role, be aware that the service role must be configured with permission to invoke those services. This requirement applies to all AWS Automation runbooks (AWS-\* runbooks) such as the `AWS-ConfigureS3BucketLogging`, `AWS-CreateDynamoDBBackup`, and `AWS-RestartEC2Instance` runbooks, to name a few. This requirement also applies to any custom Automation runbooks you create that invoke other AWS services by using actions that call other services. For example, if you use the `aws:executeAwsApi`, `aws:createStack`, or `aws:copyImage` actions, then you must configure the service role with permission to invoke those services. You can enable permissions to other AWS services by adding an IAM inline policy to the role. For more information, see [Add an Automation inline policy to invoke other AWS services](#).

This reference includes topics that describe each of the Systems Manager runbooks that are owned by AWS, AWS Support, and AWS Config. Runbooks are organized by the relevant AWS service. Each page provides an explanation of the required and optional parameters you can specify when using the runbook. Each page also lists the steps in the runbook and the output of the automation, if any.

This section does *not* include a separate page for runbooks that require approval such as the `AWS-CreateManagedLinuxInstanceWithApproval` or `AWS-StopEC2InstanceWithApproval` runbook. Any runbook name that includes *WithApproval*, means the runbook includes the `aws:approve` action. This action temporarily pauses an automation until designated principals either approve or reject the action. After the required number of approvals is reached, the automation resumes.

For information about running automations, see [Running a simple automation](#). For information about running automations on multiple targets, see [Running automations that use targets and rate controls](#).

## Topics

- [View runbook content \(p. 2\)](#)
- [API Gateway \(p. 2\)](#)
- [AWS CloudFormation \(p. 6\)](#)
- [CloudFront \(p. 10\)](#)
- [CloudTrail \(p. 16\)](#)
- [CloudWatch \(p. 21\)](#)
- [CodeBuild \(p. 22\)](#)
- [AWS CodeDeploy \(p. 25\)](#)
- [AWS Config \(p. 26\)](#)
- [AWS Directory Service \(p. 28\)](#)
- [DynamoDB \(p. 34\)](#)
- [Amazon EBS \(p. 39\)](#)
- [Amazon EC2 \(p. 51\)](#)



- [Amazon ECS \(p. 131\)](#)
- [Amazon EFS \(p. 136\)](#)
- [Amazon EKS \(p. 139\)](#)
- [Elastic Beanstalk \(p. 148\)](#)
- [Elastic Load Balancing \(p. 153\)](#)
- [Amazon EMR \(p. 160\)](#)
- [Amazon OpenSearch Service \(p. 164\)](#)
- [EventBridge \(p. 167\)](#)
- [GuardDuty \(p. 170\)](#)
- [IAM \(p. 171\)](#)
- [AWS KMS \(p. 189\)](#)
- [Lambda \(p. 191\)](#)
- [Amazon RDS \(p. 201\)](#)
- [Amazon Redshift \(p. 238\)](#)
- [Amazon S3 \(p. 249\)](#)
- [Secrets Manager \(p. 262\)](#)
- [Security Hub \(p. 264\)](#)
- [Amazon SNS \(p. 266\)](#)
- [Systems Manager \(p. 268\)](#)
- [Third-party \(p. 285\)](#)
- [Amazon VPC \(p. 289\)](#)
- [AWS WAF \(p. 323\)](#)
- [Amazon WorkSpaces \(p. 327\)](#)
- [X-Ray \(p. 330\)](#)

## View runbook content

You can view the content for runbooks in the Systems Manager console.

### To view runbook content

1. Open the AWS Systems Manager console at <https://console.aws.amazon.com/systems-manager/>.
2. In the navigation pane, choose **Documents**.

-or-

If the AWS Systems Manager home page opens first, choose the menu icon (≡) to open the navigation pane, and then choose **Documents** in the navigation pane.

3. In the **Categories** section, choose **Automation documents**.
4. Choose a runbook, and then choose **View details**.
5. Choose the **Content** tab.

## API Gateway

AWS Systems Manager Automation provides predefined runbooks for Amazon API Gateway. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

## Topics

- [AWSConfigRemediation-DeleteAPIGatewayStage \(p. 3\)](#)
- [AWSConfigRemediation-EnableAPIGatewayTracing \(p. 4\)](#)
- [AWSConfigRemediation-UpdateAPIGatewayMethodCaching \(p. 5\)](#)

# AWSConfigRemediation-DeleteAPIGatewayStage

## Description

The `AWSConfigRemediation-DeleteAPIGatewayStage` runbook deletes an Amazon API Gateway (API Gateway) stage. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `StageArn`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the API Gateway stage you want to delete.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `apigateway:GET`
- `apigateway:DELETE`

## Document Steps

- `aws:executeScript` - Deletes the API Gateway stage specified in the `StageArn` parameter.

# AWSConfigRemediation- EnableAPIGatewayTracing

## Description

The `AWSConfigRemediation-EnableAPIGatewayTracing` runbook enables tracing on an Amazon API Gateway (API Gateway) stage. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\/\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `StageArn`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the API Gateway stage you want to enable tracing on.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:GetAutomationExecution`
- `ssm:StartAutomationExecution`
- `config:GetResourceConfigHistory`
- `apigateway:GET`
- `apigateway:PATCH`

## Document Steps

- `aws:executeScript` - Enables tracing on the API Gateway stage specified in the `StageArn` parameter.

## AWSConfigRemediation- UpdateAPIGatewayMethodCaching

### Description

The `AWSConfigRemediation-UpdateAPIGatewayMethodCaching` runbook updates the cache method setting for an Amazon API Gateway stage resource.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=,./-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `CachingAuthorizedMethods`

Type: StringList

Description: (Required) The methods authorized to have caching enabled. The list must be some combination of `DELETE`, `GET`, `HEAD`, `OPTIONS`, `PATCH`, `POST`, and `PUT`. Caching is enabled for selected methods and disabled for non-selected methods. Caching is enabled for all methods if `ANY` is selected and is disabled for all methods if `NONE` is selected.

- `StageArn`

Type: String

Description: (Required) The API Gateway stage ARN for the `REST` API.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

- `apigateway:PATCH`
- `apigateway:GET`

#### Document Steps

- `aws:executeScript` - Accepts the stage resource ID as input, updates the cache method setting for an API Gateway stage using the `UpdateStage` API action, and verifies the update.

## AWS CloudFormation

AWS Systems Manager Automation provides predefined runbooks for AWS CloudFormation. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

#### Topics

- [AWS-DeleteCloudFormationStack \(p. 6\)](#)
- [AWS-RunCfnLint \(p. 7\)](#)
- [AWSSupport-TroubleshootCFNCustomResource \(p. 8\)](#)
- [AWS-UpdateCloudFormationStack \(p. 9\)](#)

## AWS-DeleteCloudFormationStack

#### Description

Delete an AWS CloudFormation stack.

[Run this Automation \(console\)](#)

#### Document type

Automation

#### Owner

Amazon

#### Platforms

Linux, macOS, Windows

#### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `StackNameOrId`

Type: String

Description: (Required) Name or Unique ID of the CloudFormation stack to be deleted

## AWS-RunCfnLint

### Description

This runbook uses an [AWS CloudFormation Linter](#) ( `cfn-python-lint` ) to validate YAML and JSON templates against the AWS CloudFormation resource specification. The `AWS-RunCfnLint` runbook performs additional checks, such as ensuring that valid values have been entered for resource properties. If validation is not successful, the `RunCfnLintAgainstTemplate` step fails and the linter tool's output is provided in an error message. This runbook is using `cfn-lint v0.24.4`.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ConfigureRuleFlag

Type: String

Description: (Optional) Configuration options for a rule to pass to the `--configure-rule` parameter.

Example: `E2001:strict=false,E3012:strict=false`.

- FormatFlag

Type: String

Description: (Optional) Value to pass to the `--format` parameter to specify the output format.

Valid values: `Default` | `quiet` | `parseable` | `json`

Default: `Default`

- IgnoreChecksFlag

Type: String

Description: (Optional) IDs of rules to pass to the `--ignore-checks` parameter. These rules are not checked.

Example: E1001,E1003,W7001

- IncludeChecksFlag

Type: String

Description: (Optional) IDs of rules to pass to the `--include-checks` parameter. These rules are checked.

Example: E1001,E1003,W7001

- InfoFlag

Type: String

Description: (Optional) Option for the `--info` parameter. Include the option to enable additional logging information about the template processing.

Default: false

- TemplateFileName

Type: String

Description: The name, or key, of the template file in the S3 bucket.

- TemplateS3BucketName

Type: String

Description: The name of the S3 bucket containing the packer template.

- RegionsFlag

Type: String

Description: (Optional) Values to pass to the for `--regions` parameter to test the template against specified AWS Regions.

Example: us-east-1,us-west-1

### Document Steps

RunCfnLintAgainstTemplate – Runs the `cfn-python-lint` tool against the specified AWS CloudFormation template.

### Outputs

RunCfnLintAgainstTemplate.output – The stdout from the `cfn-python-lint` tool.

## AWSSupport-TroubleshootCFNCustomResource

### Description

The `AWSSupport-TroubleshootCFNCustomResource` runbook helps diagnose why an AWS CloudFormation stack failed in creating, updating, or deleting a custom resource. The runbook checks the service token used for the custom resource and the error message that was returned. After reviewing the details for the custom resource, the runbook output provides an explanation of the stack behavior and troubleshooting steps for the custom resource.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- StackName

Type: String

Description: (Required) The name of the AWS CloudFormation stack where the custom resource failed.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- cloudformation:DescribeStacks
- cloudformation:DescribeStackEvents
- cloudformation:ListStackResources
- ec2:DescribeRouteTables
- ec2:DescribeNatGateways
- ec2:DescribeSecurityGroups
- ec2:DescribeVpcs
- ec2:DescribeVpcEndpoints
- ec2:DescribeSubnets
- logs:FilterLogEvents

### Document Steps

- validateCloudFormationStack - Verifies that the AWS CloudFormation stack exists in the same AWS account and AWS Region.
- checkCustomResource - Analyzes the AWS CloudFormation stack, checks the failed custom resource, and outputs information about how to troubleshoot the failed custom resource.

## AWS-UpdateCloudFormationStack

### Description



Update an AWS CloudFormation stack by using an AWS CloudFormation template stored in an Amazon S3 bucket.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- LambdaAssumeRole

Type: String

Description: (Required) The ARN of the role assumed by Lambda

- StackNameOrId

Type: String

Description: (Required) Name or Unique ID of the AWS CloudFormation stack to be updated

- TemplateUrl

Type: String

Description: (Required) S3 bucket location that contains the updated CloudFormation template (e.g. `https://s3.amazonaws.com/doc-example-bucket/updated.template`)

## CloudFront

AWS Systems Manager Automation provides predefined runbooks for Amazon CloudFront. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

**Topics**

- [AWSConfigRemediation-EnableCloudFrontDefaultRootObject \(p. 11\)](#)
- [AWSConfigRemediation-EnableCloudFrontAccessLogs \(p. 12\)](#)
- [AWSConfigRemediation-EnableCloudFrontOriginAccessIdentity \(p. 13\)](#)
- [AWSConfigRemediation-EnableCloudFrontOriginFailover \(p. 14\)](#)
- [AWSConfigRemediation-EnableCloudFrontViewerPolicyHTTPS \(p. 15\)](#)

# AWSConfigRemediation- EnableCloudFrontDefaultRootObject

## Description

The `AWSConfigRemediation-EnableCloudFrontDefaultRootObject` runbook configures the default root object for the Amazon CloudFront (CloudFront) distribution that you specify.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- CloudFrontDistributionId

Type: String

Description: (Required) The ID of the CloudFront distribution that you want to configure the default root object for.

- DefaultRootObject

Type: String

Description: (Required) The object that you want CloudFront to return when a viewer request points to your root URL.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudfront:GetDistributionConfig`
- `cloudfront:UpdateDistribution`

## Document Steps

- `aws:executeScript` - Configures the default root object for the CloudFront distribution that you specify in the `CloudFrontDistributionId` parameter.

## AWSConfigRemediation- EnableCloudFrontAccessLogs

### Description

The `AWSConfigRemediation-EnableCloudFrontAccessLogs` runbook enables access logging for the Amazon CloudFront (CloudFront) distribution you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `BucketName`

Type: String

Description: (Required) The name of the Amazon Simple Storage Service (Amazon S3) bucket you want to store access logs in. Buckets in the `af-south-1`, `ap-east-1`, `eu-south-1`, and `me-south-1` AWS Region are not supported.

- `CloudFrontId`

Type: String

Description: (Required) The ID of the CloudFront distribution you want to enable access logging on.

- `IncludeCookies`

Type: Boolean

Valid values: `true` | `false`

Description: (Optional) Set this parameter to `true`, if you want cookies to be included in the access logs.

- `Prefix`

Type: String

Description: (Optional) An optional string that you want CloudFront to prefix to the access log filenames for your distribution, for example, `myprefix/`.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudfront:GetDistribution`
- `cloudfront:GetDistributionConfig`
- `cloudfront:UpdateDistribution`
- `s3:GetBucketLocation`
- `s3:GetBucketAcl`
- `s3:PutBucketAcl`

### Document Steps

- `aws:executeScript` - Enables access logging for the CloudFront distribution you specify in the `CloudFrontDistributionId` parameter.

## AWSConfigRemediation- EnableCloudFrontOriginAccessIdentity

### Description

The `AWSConfigRemediation-EnableCloudFrontOriginAccessIdentity` runbook enables origin access identity for the Amazon CloudFront (CloudFront) distribution you specify. This automation assigns the same CloudFront Origin Access Identity for all Origins of the Amazon Simple Storage Service (Amazon S3) Origin type without origin access identity for the CloudFront distribution you specify. This automation does not grant read permission to the origin access identity for CloudFront to access objects in your Amazon S3 bucket. You must update your Amazon S3 bucket permissions to allow access.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- CloudFrontDistributionId

Type: String

Description: (Required) The ID of the CloudFront distribution you want to enable origin failover on.

- OriginAccessIdentityId

Type: String

Description: (Required) The ID of the CloudFront origin access identity to associate with the origin.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudfront:GetDistributionConfig`
- `cloudfront:UpdateDistribution`

### Document Steps

- `aws:executeScript` - Enables origin access identity for the CloudFront distribution you specify in the `CloudFrontDistributionId` parameter, and verifies the origin access identity was assigned.

## AWSConfigRemediation- EnableCloudFrontOriginFailover

### Description

The `AWSConfigRemediation-EnableCloudFrontOriginFailover` runbook enables origin failover for the Amazon CloudFront (CloudFront) distribution you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- CloudFrontDistributionId

Type: String

Description: (Required) The ID of the CloudFront distribution you want to enable origin failover on.

- OriginGroupId

Type: String

Description: (Required) The ID of the origin group.

- PrimaryOriginId

Type: String

Description: (Required) The ID of the primary origin in the origin group.

- SecondaryOriginId

Type: String

Description: (Required) The ID of the secondary origin in the origin group.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudfront:GetDistributionConfig`
- `cloudfront:UpdateDistribution`

## Document Steps

- `aws:executeScript` - Enables origin failover for the CloudFront distribution you specify in the `CloudFrontDistributionId` parameter, and verifies that failover has been enabled.

# AWSConfigRemediation- EnableCloudFrontViewerPolicyHTTPS

## Description

The `AWSConfigRemediation-EnableCloudFrontViewerPolicyHTTPS` runbook enables the viewer protocol policy for the Amazon CloudFront (CloudFront) distribution you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- CloudFrontDistributionId

Type: String

Description: (Required) The ID of the CloudFront distribution you want to enable the viewer protocol policy on.

- ViewerProtocolPolicy

Type: String

Valid values: `https-only, redirect-to-https`

Description: (Required) The protocol that viewers can use to access the files in the origin.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudfront:GetDistributionConfig`
- `cloudfront:UpdateDistribution`

### Document Steps

- `aws:executeScript` - Enables the viewer protocol policy for the CloudFront distribution you specify in the `CloudFrontDistributionId` parameter, and verifies the policy was assigned.

## CloudTrail

AWS Systems Manager Automation provides predefined runbooks for AWS CloudTrail. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-CreateCloudTrailMultiRegionTrail \(p. 17\)](#)
- [AWS-EnableCloudTrail \(p. 18\)](#)
- [AWSConfigRemediation-EnableCloudTrailEncryptionWithKMS \(p. 19\)](#)
- [AWSConfigRemediation-EnableCloudTrailLogFileValidation \(p. 20\)](#)

## AWSConfigRemediation-CreateCloudTrailMultiRegionTrail

### Description

The `AWSConfigRemediation-CreateCloudTrailMultiRegionTrail` runbook creates an AWS CloudTrail (CloudTrail) trail that delivers log files from multiple AWS Regions to the Amazon Simple Storage Service (Amazon S3) bucket of your choice.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `BucketName`

Type: String

Description: (Required) The name of the Amazon S3 bucket you want to upload logs to.

- `KeyPrefix`

Type: String

Description: (Optional) The Amazon S3 key prefix that comes after the name of the bucket you designated for log file delivery.

- `TrailName`

Type: String

Description: (Required) The name of the CloudTrail trail to be created.



### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudtrail:CreateTrail`
- `cloudtrail:StartLogging`
- `cloudtrail:GetTrail`
- `s3:PutObject`
- `s3:GetBucketAcl`
- `s3:PutBucketLogging`
- `s3:ListBucket`

### Document Steps

- `aws:executeAwsApi` - Accepts the trail name and the Amazon S3 bucket name as input and creates a CloudTrail trail.
- `aws:executeAwsApi` - Enables logging on the created trail and starts log delivery to the Amazon S3 bucket you specified.
- `aws:assertAwsResourceProperty` - Verifies that the CloudTrail trail has been created.

## AWS-EnableCloudTrail

### Description

Create an AWS CloudTrail trail and configure logging to an S3 bucket.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `S3BucketName`

Type: String

Description: (Required) Name of the S3 bucket designated for publishing log files.

**Note**

The S3 bucket must exist and the bucket policy must grant CloudTrail permission to write to it. For information, see [Amazon S3 Bucket Policy for CloudTrail](#) .

- TrailName

Type: String

Description: (Required) The name of the new trail.

## AWSConfigRemediation- EnableCloudTrailEncryptionWithKMS

### Description

The AWSConfigRemediation-EnableCloudTrailEncryptionWithKMS runbook encrypts an AWS CloudTrail (CloudTrail) trail using the AWS Key Management Service (AWS KMS) customer managed key you specify. This runbook should only be used as a baseline to ensure that your CloudTrail trails are encrypted according to minimum recommended security best practices. We recommend encrypting multiple trails with different KMS keys. CloudTrail digest files are not encrypted. If you have previously set the `EnableLogFileValidation` parameter to `true` for the trail, see the "Use server-side encryption with AWS KMS managed keys" section of the [CloudTrail Preventative Security Best Practices](#) topic in the *AWS CloudTrail User Guide* for more information.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\d{4}`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- KMSKeyId

Type: String

Description: (Required) The ARN, key ID, or the key alias of the of the customer managed key you want to use to encrypt the trail you specify in the `TrailName` parameter.

- TrailName

Type: String

Description: (Required) The ARN or name of the trail you want to update to be encrypted.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudtrail:GetTrail`
- `cloudtrail:UpdateTrail`

### Document Steps

- `aws:executeAwsApi` - Enables encryption on the trail you specify in the `TrailName` parameter.
- `aws:executeAwsApi` - Gathers the ARN for the customer managed key you specify in the `KMSKeyId` parameter.
- `aws:assertAwsResourceProperty` - Verifies that encryption has been enabled on the CloudTrail trail.

## AWSConfigRemediation- EnableCloudTrailLogFileValidation

### Description

The `AWSConfigRemediation-EnableCloudTrailLogFileValidation` runbook enables log file validation for your AWS CloudTrail trail.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- TrailName

Type: String

Description: (Required) The name or Amazon Resource Name (ARN) of the trail you want to enable log validation for.

#### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `cloudtrail:DescribeTrails`
- `cloudtrail:UpdateTrail`

#### Document Steps

- `aws:executeAwsApi` - Enables log validation for the AWS CloudTrail trail you specify in the `TrailName` parameter.
- `aws:assertAwsResourceProperty` - Verifies log validation is enabled for your trail.

## CloudWatch

AWS Systems Manager Automation provides predefined runbooks for Amazon CloudWatch. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

#### Topics

- [AWS-ConfigureCloudWatchOnEC2Instance \(p. 21\)](#)

## AWS-ConfigureCloudWatchOnEC2Instance

#### Description

Enable or disable Amazon CloudWatch detailed monitoring on managed instances.

[Run this Automation \(console\)](#)

#### Document type

Automation

#### Owner

Amazon

#### Platforms

Linux, macOS, Windows

#### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the Amazon EC2 instance on which you want to enable CloudWatch monitoring.

- properties

Type: String

Description: (Optional) This parameter is not supported. It is listed here for backwards compatibility.

- status

Valid values: Enabled | Disabled

Description: (Optional) Specifies whether to enable or disable CloudWatch.

Default: Enabled

### Document Steps

configureCloudWatch - Configures CloudWatch on the Amazon EC2 instance with the given status.

### Outputs

This automation has no output.

## CodeBuild

AWS Systems Manager Automation provides predefined runbooks for AWS CodeBuild. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-ConfigureCodeBuildProjectWithKMSCMK \(p. 22\)](#)
- [AWSConfigRemediation-DeleteAccessKeysFromCodeBuildProject \(p. 24\)](#)

## AWSConfigRemediation-ConfigureCodeBuildProjectWithKMSCMK

### Description

The AWSConfigRemediation-ConfigureCodeBuildProjectWithKMSCMK runbook encrypts an AWS CodeBuild (CodeBuild) project's build artifacts using the AWS Key Management Service (AWS KMS) customer managed key you specify. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- KMSKeyId

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the AWS KMS customer managed key you want to use to encrypt the CodeBuild project you specify in the `ProjectId` parameter.

- ProjectId

Type: String

Description: (Required) The ID of the CodeBuild project whose build artifacts you want to encrypt.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `codebuild:BatchGetProjects`
- `codebuild:UpdateProject`
- `config:GetResourceConfigHistory`

## Document Steps

- `aws:executeAwsApi` - Gathers the CodeBuild project name from the project ID.
- `aws:executeAwsApi` - Enables encryption on the CodeBuild project you specify in the `ProjectId` parameter.
- `aws:assertAwsResourceProperty` - Verifies that encryption has been enabled on the CodeBuild project.

## Outputs

`UpdateLambdaConfig.UpdateFunctionConfigurationResponse` - Response from the `UpdateFunctionConfiguration` API call.

# AWSConfigRemediation-DeleteAccessKeysFromCodeBuildProject

## Description

The `AWSConfigRemediation-DeleteAccessKeysFromCodeBuildProject` runbook deletes the `AWS_ACCESS_KEY_ID` and `AWS_SECRET_ACCESS_KEY` environment variables from the AWS CodeBuild (CodeBuild) project you specify. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@/-\]+`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- ResourceId

Type: String

Description: (Required) The ID of the CodeBuild project whose access key environment variables you want to delete.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `codebuild:BatchGetProjects`
- `codebuild:UpdateProject`

## Document Steps

- `aws:executeScript` - Deletes the access key environment variables for the CodeBuild project specified in the `ResourceId` parameter.

# AWS CodeDeploy

AWS Systems Manager Automation provides predefined runbooks for AWS CodeDeploy. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

## Topics

- [AWSSupport-TroubleshootCodeDeploy \(p. 25\)](#)

## AWSSupport-TroubleshootCodeDeploy

### Description

The `AWSSupport-TroubleshootCodeDeploy` runbook helps diagnose why an AWS CodeDeploy deployment failed on an Amazon Elastic Compute Cloud (Amazon EC2) instance. The runbook outputs steps to help you resolve the issue or troubleshoot further. Best practices for CodeDeploy are also provided to help you avoid similar issues in the future.

This runbook can help you to resolve the following issues:

- The CodeDeploy agent is not installed or not running on the Amazon EC2 instance
- The Amazon EC2 instance does not have an AWS Identity and Access Management (IAM) instance profile attached
- The IAM instance profile attached to the Amazon EC2 instance does not have the required Amazon Simple Storage Service (Amazon S3) permissions
- A revision stored in Amazon S3 is missing, or the Amazon S3 bucket used is in an AWS Region that is different than the Amazon EC2 instance
- Application specification (AppSpec) file issues
- "File already exists at location" errors
- Failed CodeDeploy managed lifecycle event hooks
- Failed customer managed lifecycle event hooks
- Scale-in events during the deployment

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your



behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `DeploymentId`

Type: String

Description: (Required) The ID of the deployment which failed.

- `InstanceId`

Type: String

Description: (Required) The ID of the Amazon EC2 instance where the deployment failed.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `codedeploy:GetDeployment`
- `codedeploy:GetDeploymentTarget`
- `ec2:DescribeInstances`

### Document Steps

- `aws:executeAwsApi` - Verifies the values provided for the `DeploymentId` and `InstanceId` parameters.
- `aws:executeScript` - Collects information from the Amazon EC2 instance such as the state of the instance and IAM instance profile details.
- `aws:executeScript` - Reviews the specified deployment, and returns an analysis regarding why the deployment failed.

## AWS Config

AWS Systems Manager Automation provides predefined runbooks for AWS Config. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSSupport-SetupConfig \(p. 26\)](#)

## AWSSupport-SetupConfig

### Description

The `AWSSupport-SetupConfig` runbook creates an AWS Identity and Access Management (IAM) service-linked role, a configuration recorder powered by AWS Config, and a delivery channel with an Amazon Simple Storage Service (Amazon S3) bucket where AWS Config sends configuration snapshots and configuration history files. If you specify values for the `AggregatorAccountId` and `AggregatorAccountRegion` parameters, the runbook also creates authorizations for data aggregation to collect AWS Config configuration and compliance data from multiple AWS accounts and multiple AWS Regions. To learn more about aggregating data from multiple accounts and Regions, see [Multi-Account Multi-Region Data Aggregation](#) in the *AWS Config Developer Guide* .

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- AggregatorAccountId

Type: String

Description: (Optional) The ID of the AWS account where an aggregator will be added to aggregate AWS Config configuration and compliance data from multiple accounts and AWS Regions. This account is also used by the aggregator to authorize the source accounts.

- AggregatorAccountRegion

Type: String

Description: (Optional) The Region where an aggregator will be added to aggregate AWS Config configuration and compliance data from multiple accounts and Regions.

- IncludeGlobalResourcesRegion

Type: String

Default: us-east-1

Description: (Required) To avoid recording global resource data in each Region, specify one Region to record global resource data from.

- Partition

Type: String

Default: aws

Description: (Required) The partition you want to collect AWS Config configuration and compliance data from.

- S3BucketName

Type: String

Default: aws-config-delivery-channel

Description: (Optional) The name you want to apply to the Amazon S3 bucket created for the delivery channel. The account ID is appended to the end of the name.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:DescribeConfigurationRecorders`
- `config:DescribeDeliveryChannels`
- `config:PutAggregationAuthorization`
- `config:PutConfigurationRecorder`
- `config:PutDeliveryChannel`
- `config:StartConfigurationRecorder`
- `iam:CreateServiceLinkedRole`
- `iam:PassRole`
- `s3:CreateBucket`
- `s3:ListAllMyBuckets`
- `s3:PutBucketPolicy`

### Document Steps

- `aws:executeScript` - Creates a service-linked IAM role for AWS Config if one does not already exist.
- `aws:executeScript` - Creates a configuration recorder if one does not already exist.
- `aws:executeScript` - Creates an Amazon S3 bucket to be used by the delivery channel if one does not already exist.
- `aws:executeScript` - Creates a delivery channel using the resources created by the runbook.
- `aws:executeAwsApi` - Starts the configuration recorder.
- `aws:executeScript` - If you specified values for the `AggregatorAccountId` and `AggregatorAccountRegion` parameters, authorizations for multi-account and multi-Region data aggregation are configured.

## AWS Directory Service

AWS Systems Manager Automation provides predefined runbooks for AWS Directory Service. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWS-CreateDSManagementInstance \(p. 28\)](#)
- [AWSSupport-TroubleshootDirectoryTrust \(p. 31\)](#)

## AWS-CreateDSManagementInstance

### Description

The `AWS-CreateDSManagementInstance` runbook creates an Amazon Elastic Compute Cloud (Amazon EC2) Windows instance that you can use to manage your AWS Directory Service directory. The management instance can't be used to manage AD Connector directories.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- AmiID

Type: String

Default: `{{ ssm:/aws/service/ami-windows-latest/Windows_Server-2019-English-Full-Base }}`

Description: (Required) The ID of the Amazon Machine Image (AMI) you want to use to launch the management instance.

- DirectoryId

Type: String

Description: (Required) The ID of the AWS Directory Service directory you want to manage. The instance is joined to the directory you specify.

- IamInstanceProfileName

Type: String

Description: (Required) The name you specify is applied to the IAM instance profile that is created by the automation and attached to the management instance.

- InstanceType

Type: String

Default: `t3.medium`

Allowed values:

- t2.nano
- t2.micro
- t2.small
- t2.medium
- t2.large
- t2.xlarge
- t2.2xlarge
- t3.nano

- t3.micro
- t3.small
- t3.medium
- t3.large
- t3.xlarge
- t3.2xlarge

Description: (Required) The type of instance you want to launch.

- KeyPairName

Type: String

Description: (Optional) The key pair to use when creating the instance. If you do not specify a value, no key pair is associated with the instance.

- RemoteAccessCidr

Type: String

Description: (Required) The CIDR block you want to allow RDP traffic (port 3389) from. The CIDR block you specify is applied to an inbound rule that's added to the security group created by the automation.

- SecurityGroupName

Type: String

Description: (Required) The name you specify is applied to the security group that is created by the automation and associated with the management instance.

- Tags

Type: MapList

Description: (Optional) A key-value pair you want to apply to the resources created by the automation.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ds:DescribeDirectories`
- `ec2:AuthorizeSecurityGroupIngress`
- `ec2:CreateSecurityGroup`
- `ec2:CreateTags`
- `ec2>DeleteSecurityGroup`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceStatus`
- `ec2:DescribeKeyPairs`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeVpcs`
- `ec2:RunInstances`
- `ec2:TerminateInstances`
- `iam:AddRoleToInstanceProfile`
- `iam:AttachRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:CreateRole`

- iam:DeleteInstanceProfile
- iam:DeleteRole
- iam:DetachRolePolicy
- iam:GetInstanceProfile
- iam:GetRole
- iam>ListAttachedRolePolicies
- iam>ListInstanceProfiles
- iam>ListInstanceProfilesForRole
- iam:PassRole
- iam:RemoveRoleFromInstanceProfile
- iam:TagInstanceProfile
- iam:TagRole
- ssm:CreateDocument
- ssm>DeleteDocument
- ssm:DescribeInstanceInformation
- ssm:GetAutomationExecution
- ssm:GetParameters
- ssm>ListCommandInvocations
- ssm>ListCommands
- ssm>ListDocuments
- ssm:SendCommand
- ssm:StartAutomationExecution

### Document Steps

- aws:executeAwsApi - Gathers details about the directory you specify in the DirectoryId parameter.
- aws:executeAwsApi - Gets the CIDR block of the virtual private cloud (VPC) where the directory was launched.
- aws:executeAwsApi - Creates a security group using the value you specify in the SecurityGroupName parameter.
- aws:executeAwsApi - Creates an inbound rule for the newly created security group that allows RDP traffic from the CIDR you specify in the RemoteAccessCidr parameter.
- aws:executeAwsApi - Creates an IAM role and instance profile using the value you specify in the IamInstanceProfileName parameter.
- aws:executeAwsApi - Launches an Amazon EC2 instance based on the values you specify in the runbook parameters.
- aws:executeAwsApi - Creates an AWS Systems Manager document to join the newly launched instance to your directory.
- aws:runCommand - Joins the new instance to your directory.
- aws:runCommand - Installs remote server administration tools on the new instance.

## AWSSupport-TroubleshootDirectoryTrust

### Description

The AWSSupport-TroubleshootDirectoryTrust runbook diagnoses trust creation issues between an AWS Managed Microsoft AD and a Microsoft Active Directory. The automation ensures the directory

type supports trusts, and then checks the associated security group rules, network access control lists (network ACLs), and route tables for potential connectivity issues.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- DirectoryId

Type: String

Allowed pattern: `^d-[a-z0-9]{10}$`

Description: (Required) The ID of the AWS Managed Microsoft AD to troubleshoot.

- RemoteDomainCidrs

Type: StringList

Allowed pattern: `^(([0-9]{1-9}[0-9]{1-9}[0-9]{1-9}){2}[0-4][0-9]{25}[0-5])\.){3}([0-9]{1-9}[0-9]{1-9}[0-9]{1-9}){2}[0-4][0-9]{25}[0-5])\.(3[0-2][1-2][0-9][1-9])$`

Description: (Required) The CIDR(s) of the remote domain you are attempting to establish a trust relationship with. You can add multiple CIDRs using comma-separated values. For example, 172.31.48.0/20, 192.168.1.10/32.

- RemoteDomainName

Type: String

Description: (Required) The fully qualified domain name of the remote domain you are establishing a trust relationship with.

- RequiredTrafficACL

Type: String

Description: (Required) The default port requirements for AWS Managed Microsoft AD. In most cases, you should not modify the default value.

Default: `{"inbound":{"tcp":[[53,53],[88,88],[135,135],[389,389],[445,445],[464,464],[636,636],[1024,65535]],"udp":[[53,53],[88,88],[123,123],[138,138],[389,389],[445,445],[464,464]],"icmp":[[-1,-1]]},"outbound":{"-1":[[0,65535]]}}`

- RequiredTrafficSG

Type: String

Description: (Required) The default port requirements for AWS Managed Microsoft AD. In most cases, you should not modify the default value.

Default: {"inbound":{"tcp":[[53,53],[88,88],[135,135],[389,389],[445,445],[464,464],[636,636],[1024,65535]],"udp":[[53,53],[88,88],[123,123],[138,138],[389,389],[445,445],[464,464]],"icmp":[[-1,-1]]},"outbound":{"-1":[[0,65535]]}}

- TrustId

Type: String

Description: (Optional) The ID of the trust relationship to troubleshoot.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ds:DescribeConditionalForwarders
- ds:DescribeDirectories
- ds:DescribeTrusts
- ds:ListIpRoutes
- ec2:DescribeNetworkAcls
- ec2:DescribeSecurityGroups
- ec2:DescribeSubnets

### Document Steps

- aws:assertAwsResourceProperty - Confirms the directory type is AWS Managed Microsoft AD.
- aws:executeAwsApi - Gets information about the AWS Managed Microsoft AD.
- aws:branch - Branches automation if a value is provided for the TrustId input parameter.
- aws:executeAwsApi - Gets information about the trust relationship.
- aws:executeAwsApi - Gets the conditional forwarder DNS IP addresses for the RemoteDomainName .
- aws:executeAwsApi - Gets information about IP routes that have been added to the AWS Managed Microsoft AD.
- aws:executeAwsApi - Gets the CIDRs of the AWS Managed Microsoft AD subnets.
- aws:executeAwsApi - Gets information about the security groups associated with the AWS Managed Microsoft AD.
- aws:executeAwsApi - Gets information about the network ACLs associated with the AWS Managed Microsoft AD.
- aws:executeScript - Confirms the RemoteDomainCidrs are valid values. Confirms that the AWS Managed Microsoft AD has conditional forwarders for the RemoteDomainCidrs , and that the requisite IP routes have been added to the AWS Managed Microsoft AD if the RemoteDomainCidrs are non-RFC 1918 IP addresses.
- aws:executeScript - Evaluates security group rules.
- aws:executeScript - Evaluates network ACLs.

### Outputs



`evalDirectorySecurityGroup.output` - Results from evaluating whether the security group rules associated with the AWS Managed Microsoft AD allow the requisite traffic for trust creation.

`evalAclEntries.output` - Results from evaluating whether the network ACLs associated with the AWS Managed Microsoft AD allow the requisite traffic for trust creation.

`evaluateRemoteDomainCidr.output` - Results from evaluating whether the `RemoteDomainCidrs` are valid values. Confirms that the AWS Managed Microsoft AD has conditional forwarders for the `RemoteDomainCidrs`, and that the requisite IP routes have been added to the AWS Managed Microsoft AD if the `RemoteDomainCidrs` are non-RFC 1918 IP addresses.

## DynamoDB

AWS Systems Manager Automation provides predefined runbooks for Amazon DynamoDB. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWS-CreateDynamoDBBackup \(p. 34\)](#)
- [AWS-DeleteDynamoDbBackup \(p. 35\)](#)
- [AWSConfigRemediation-DeleteDynamoDbTable \(p. 35\)](#)
- [AWS-DeleteDynamoDbTableBackups \(p. 36\)](#)
- [AWSConfigRemediation-EnableEncryptionOnDynamoDBTable \(p. 37\)](#)
- [AWSConfigRemediation-EnablePITRForDynamoDbTable \(p. 38\)](#)

## AWS-CreateDynamoDBBackup

### Description

Create a backup of an Amazon DynamoDB table.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `BackupName`

Type: String

Description: (Required) Name of the backup to create.

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

- TableName

Type: String

Description: (Required) Name of the DynamoDB table.

## AWS-DeleteDynamoDbBackup

### Description

Delete the backup of an Amazon DynamoDB table.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- BackupArn

Type: String

Description: (Required) ARN of the DynamoDB table backup to delete.

## AWSConfigRemediation-DeleteDynamoDbTable

### Description

The `AWSConfigRemediation-DeleteDynamoDbTable` runbook deletes the Amazon DynamoDB (DynamoDB) table you specify.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Databases

**Parameters**

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- TableName

Type: String

Description: (Required) The name of the DynamoDB table you want to delete.

**Required IAM permissions**

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- dynamodb>DeleteTable
- dynamodb:DescribeTable

**Document Steps**

- aws:executeScript - Deletes the DynamoDB table specified in the TableName parameter.
- aws:executeScript - Verifies the DynamoDB table has been deleted.

## AWS-DeleteDynamoDbTableBackups

**Description**

Delete DynamoDB table backups based on retention days or count.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Databases

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

- RetentionCount

Type: String

Default: 10

Description: (Optional) The number of backups to retain for the table. If more than the specified number of backup exist, the oldest backups beyond that number are deleted. Either RetentionCount or RetentionDays can be used, not both.

- RetentionDays

Type: String

Description: (Optional) The number of days to retain backups for the table. Backups older than the specified number of days are deleted. Either RetentionCount or RetentionDays can be used, not both.

- TableName

Type: String

Description: (Required) Name of the DynamoDB table.

## AWSConfigRemediation- EnableEncryptionOnDynamoDBTable

**Description**

The AWSConfigRemediation-EnableEncryptionOnDynamoDBTable runbook encrypts an Amazon DynamoDB (DynamoDB) table using the AWS Key Management Service (AWS KMS) customer managed key you specify for the `KMSKeyId` parameter.

[Run this Automation \(console\)](#)

**Document type**

## Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- KMSKeyId

Type: String

Description: (Required) The ARN of the customer managed key you want to use to encrypt the DynamoDB table you specify in the `TableName` parameter.

- TableName

Type: String

Description: (Required) The name of the DynamoDB table you want to encrypt.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `dynamodb:DescribeTable`
- `dynamodb:UpdateTable`

### Document Steps

- `aws:executeAwsApi` - Encrypts the DynamoDB table you specify in the `TableName` parameter.
- `aws:waitForAwsResourceProperty` - Verifies the `Enabled` property for the DynamoDB table's `SSSEncryptionSpecification` is set to `true`.
- `aws:assertAwsResourceProperty` - Verifies the DynamoDB table is encrypted with the customer managed key specified in the `KMSKeyId` parameter.

## AWSConfigRemediation- EnablePITRForDynamoDbTable

### Description

The `AWSConfigRemediation-EnablePITRForDynamoDbTable` runbook enables point-in-time recovery (PITR) on the Amazon DynamoDB table you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- TableName

Type: String

Description: (Required) The name of the DynamoDB table to enable point-in-time recovery on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `dynamodb:DescribeContinuousBackups`
- `dynamodb:UpdateContinuousBackups`

### Document Steps

- `aws:executeAwsApi` - Enables point-in-time recovery on the DynamoDB table you specify in the `TableName` parameter.
- `aws:assertAwsResourceProperty` - Confirms point-in-time recovery is enabled on the DynamoDB table.

## Amazon EBS

AWS Systems Manager Automation provides predefined runbooks for Amazon Elastic Block Store. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWS-AttachEBSVolume](#) (p. 40)
- [AWSSupport-CalculateEBSPerformanceMetrics](#) (p. 41)
- [AWS-CopySnapshot](#) (p. 42)
- [AWS-CreateSnapshot](#) (p. 43)
- [AWS-DeleteEbsVolumeSnapshots](#) (p. 44)
- [AWS-DeleteSnapshot](#) (p. 45)
- [AWSConfigRemediation-DeleteUnusedEBSVolume](#) (p. 46)
- [AWS-DetachEBSVolume](#) (p. 47)
- [AWSConfigRemediation-EnableEbsEncryptionByDefault](#) (p. 48)
- [AWSSupport-ModifyEBSSnapshotPermission](#) (p. 48)
- [AWSConfigRemediation-ModifyEBSVolumeType](#) (p. 50)

## AWS-AttachEBSVolume

### Description

Attach an Amazon Elastic Block Store (Amazon EBS) volume to an Amazon Elastic Compute Cloud (Amazon EC2) instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Device

Type: String

Description: (Required) The device name (for example, /dev/sdh or xvdh ).

- InstanceId

Type: String

Description: (Required) The ID of the instance where you want to attach the volume.

- VolumeId

Type: String

Description: (Required) The ID of the Amazon EBS volume. The volume and instance must be in the same Availability Zone.

## AWSSupport- CalculateEBSPerformanceMetrics

### Description

The `AWSSupport-CalculateEBSPerformanceMetrics` runbook helps diagnose Amazon Elastic Block Store (Amazon EBS) performance issues by calculating metrics published to Amazon CloudWatch and creating CloudWatch dashboards that display the average I/O size and total aggregate throughput and IOPS of individual Amazon EBS volumes, or the total aggregate throughput and IOPS across all Amazon EBS volumes attached to an Amazon Elastic Compute Cloud (Amazon EC2) instance by utilizing CloudWatch metric math. The runbook outputs the link to the newly created CloudWatch dashboard that displays the relevant calculated CloudWatch metrics.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- EndTime

Type: String

Description: (Required) The end of the time range to use for each widget on the CloudWatch dashboard when the dashboard loads. For the value of this parameter, specify an absolute time in the ISO 8601 format. To learn more about creating ISO 8601 formatted strings, see [Creating formatted date and time strings for Systems Manager](#).

- ResourceId

Type: String

Description: (Required) The ID of the Amazon EBS volume or Amazon EC2 instance whose attached Amazon EBS volumes performance metrics you want to calculate and view.



- `StartTime`

Type: String

Description: (Required) The start of the time range to use for each widget on the CloudWatch dashboard. For the value of this parameter, specify an absolute time in the ISO 8601 format.

- `Period`

Type: String

Description: (Required) The default period, in seconds, for all metrics in this widget. The period is the length of time represented by one data point on the graph.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeVolumes`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceTypes`
- `cloudwatch:PutDashboard`
- `cloudwatch:ListMetrics`

### Document Steps

- `aws:executeScript` - Validates the values provided for `StartTime` and `EndTime` parameters.
- `aws:branch` - Checks whether the value provided in the parameter `ResourceId` is an Amazon EBS volume ID, or an Amazon EC2 instance ID.
- `aws:executeScript` - If an Amazon EBS volume ID is provided for the parameter `ResourceId`, creates a CloudWatch dashboard with calculated aggregate throughput, aggregate IOPS, and average IO size for the Amazon EBS volume.
- `aws:executeScript` - If an Amazon EC2 instance ID is provided for the parameter `ResourceId`, creates a CloudWatch dashboard with calculated aggregate throughput and aggregate IOPS for the Amazon EC2 instance and the calculated aggregate throughput, aggregate IOPS, and average IO size for each of the Amazon EBS volumes attached to the Amazon EC2 instance.

## AWS-CopySnapshot

### Description

Copies a point-in-time snapshot of an Amazon Elastic Block Store (Amazon EBS) volume. You can copy the snapshot within the same AWS Region or from one Region to another. Copies of encrypted Amazon EBS snapshots remain encrypted. Copies of unencrypted snapshots remain unencrypted. To copy an encrypted snapshot that was shared from another account, you must have permissions for the KMS key used to encrypt the snapshot. Snapshots created by copying another snapshot have an arbitrary volume ID that should not be used for any purpose.

[Run this Automation \(console\)](#)

### Document type

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Description

Type: String

Description: (Optional) A description for the Amazon EBS snapshot.

- SnapshotId

Type: String

Description: (Required) The ID of the Amazon EBS snapshot to copy.

- SourceRegion

Type: String

Description: (Required) The Region where the source snapshot currently exists.

**Document Steps**

copySnapshot - Copies a snapshot of an Amazon EBS volume.

**Outputs**

copySnapshot.SnapshotId - The ID of the new snapshot.

## AWS-CreateSnapshot

**Description**

Create a snapshot of an Amazon EBS volume.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

#### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Description

Type: String

Description: (Optional) A description for the snapshot

- Volumeld

Type: String

Description: (Required) The ID of the volume.

## AWS-DeleteEbsVolumeSnapshots

### Description

Delete a snapshot of an Amazon Elastic Block Store (Amazon EBS) volume.

#### Note

The AWS Lambda function that runs during this operation has a maximum run time (timeout) of 60 seconds. If you have a large number of Amazon EBS volume snapshots to delete, the operation might fail with an error message.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

- RetentionCount

Type: String

Default: 10

Description: (Optional) Number of snapshots to keep for the volume. Either RetentionCount or RetentionDays should be mentioned, not both.

- RetentionDays

Type: String

Description: (Optional) Number of days to keep snapshots for the volume. Either RetentionCount or RetentionDays should be mentioned, not both

- Volumeld

Type: String

Description: (Required) The volume identifier to delete snapshots for.

## AWS-DeleteSnapshot

### Description

Delete a snapshot of an Amazon EBS volume.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- SnapshotId

Type: String

Description: (Required) The ID of the EBS snapshot.

## AWSConfigRemediation-DeleteUnusedEBSVolume

### Description

The `AWSConfigRemediation-DeleteUnusedEBSVolume` runbook deletes an unused Amazon Elastic Block Store (Amazon EBS) volume.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- CreateSnapshot

Type: Boolean

Description: (Optional) If set to `true`, the automation creates a snapshot of the Amazon EBS volume before it is deleted.

- Volumeld

Type: String

Description: (Required) The ID of the Amazon EBS volume that you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:CreateSnapshot`
- `ec2>DeleteVolume`

- `ec2:DescribeSnapshots`
- `ec2:DescribeVolumes`

### Document Steps

- `aws:executeScript` - Verifies the Amazon EBS volume you specify in the `VolumeId` parameter is not in use, and creates a snapshot depending on the value you choose for the `CreateSnapshot` parameter.
- `aws:branch` - Branches based on the value you chose for the `CreateSnapshot` parameter.
- `aws:waitForAwsResourceProperty` - Waits for the snapshot to complete.
- `aws:executeAwsApi` - Deletes the snapshot if the snapshot creation failed.
- `aws:executeAwsApi` - Deletes the Amazon EBS volume you specify in the `VolumeId` parameter.
- `aws:executeScript` - Verifies the Amazon EBS volume has been deleted.

## AWS-DetachEBSVolume

### Description

Detach an Amazon EBS volume from an Amazon Elastic Compute Cloud (Amazon EC2) instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `LambdaAssumeRole`

Type: String

Description: (Optional) The ARN of the role assumed by Lambda

- `VolumeId`

Type: String

Description: (Required) The ID of the EBS volume. The volume and instance must be within the same Availability Zone

## AWSConfigRemediation- EnableEbsEncryptionByDefault

### Description

The AWSConfigRemediation-EnableEbsEncryptionByDefault runbook enables encryption on all new Amazon Elastic Block Store (Amazon EBS) volumes in the AWS account and AWS Region where you run the automation. Volumes that were created before you run the automation are not encrypted.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=, @/ -]+\]\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ec2:EnableEbsEncryptionByDefault
- ssm:StartAutomationExecution
- ssm:GetAutomationExecution

### Document Steps

- aws:executeAwsApi - Enables the default Amazon EBS encryption setting in the current account and Region.
- aws:assertAwsResourceProperty - Verifies that the default Amazon EBS encryption setting has been enabled.

## AWSsupport-ModifyEBSSnapshotPermission

### Description

The AWSsupport-ModifyEBSSnapshotPermission runbook helps you to modify permissions for multiple Amazon Elastic Block Store (Amazon EBS) snapshots. Using this runbook, you can make

snapshots `Public` or `Private` and share them with other AWS accounts. Snapshots encrypted with a default KMS key can't be shared with other accounts using this runbook.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- AccountIds

Type: StringList

Default: none

Description: (Optional) The IDs of the accounts you want to share snapshots with. This parameter is required if you enter `No` for the value of the `Private` parameter.

- AccountPermissionOperation

Type: String

Valid values: `add` | `remove`

Default: none

Description: (Optional) The type of operation to perform.

- Private

Type: String

Valid values: `Yes` | `No`

Description: (Required) Enter `No` for the value if you want to share snapshots with specific accounts.

- SnapshotIds

Type: StringList

Description: (Required) The IDs of Amazon EBS snapshots whose permission you want to modify.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.



- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeSnapshots`
- `ec2:ModifySnapshotAttribute`

### Document Steps

1. `aws:executeScript` - Verifies the IDs of the snapshots provided in the `SnapshotIds` parameter. After verifying the IDs, the script checks for encrypted snapshots and outputs a list if any are found.
2. `aws:branch` - Branches the automation based on the value you enter for the `Private` parameter.
3. `aws:executeScript` - Modifies permissions of the snapshots specified to share it with the accounts specified.
4. `aws:executeScript` - Modifies permissions of the snapshots to change them from `Public` to `Private`.

### Outputs

`ValidateSnapshots.EncryptedSnapshots`

`SharewithOtherAccounts.Result`

`MakePrivate.Result`

`MakePrivate.Commands`

## AWSConfigRemediation-ModifyEBSVolumeType

### Description

The `AWSConfigRemediation-ModifyEBSVolumeType` runbook modifies the volume type of an Amazon Elastic Block Store (Amazon EBS) volume. After the volume type is modified, the volume enters an optimizing state. For information about monitoring the progress of volume modifications, see [Monitor the progress of volume modifications](#) in the *Amazon EC2 User Guide for Linux Instances*.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `EbsVolumeId`

Type: String

Description: (Required) The ID of the Amazon EBS volume that you want to modify.

- `EbsVolumeType`

Type: String

Valid values: `standard` | `io1` | `io2` | `gp2` | `sc1` | `st1`

Description: The volume type you want to change the Amazon EBS volume to. For information about Amazon EBS volume types, see [Amazon EBS volume types](#) in the *Amazon EC2 User Guide for Linux Instances* .

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeVolumes`
- `ec2:ModifyVolume`

### Document Steps

- `aws:waitForAwsResourceProperty` - Verifies the state of the volume is `available` or `in-use` .
- `aws:executeAwsApi` - Modifies the Amazon EBS volume you specify in the `EbsVolumeId` parameter.
- `aws:waitForAwsResourceProperty` - Verifies the type of the volume has been changed to the value you specified in the `EbsVolumeType` parameter.

## Amazon EC2

AWS Systems Manager Automation provides predefined runbooks for Amazon Elastic Compute Cloud. Runbooks for Amazon Elastic Block Store are located in the [Amazon EBS \(p. 39\)](#) section of the runbook reference. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWS-ASGEnterStandby \(p. 52\)](#)
- [AWS-ASGExitStandby \(p. 53\)](#)
- [AWS-CreatedImage \(p. 54\)](#)
- [AWS-DeleteImage \(p. 55\)](#)
- [AWSEC2-CloneInstanceAndUpgradeSQLServer \(p. 56\)](#)
- [AWSEC2-CloneInstanceAndUpgradeWindows \(p. 58\)](#)
- [AWSEC2-ConfigureSTIG \(p. 61\)](#)
- [AWSEC2-PatchLoadBalancerInstance \(p. 69\)](#)

- [AWSEC2-SQLServerDBRestore](#) (p. 70)
- [AWSSupport-ActivateWindowsWithAmazonLicense](#) (p. 73)
- [AWSPremiumSupport-ChangeInstanceTypeIntelToAMD](#) (p. 75)
- [AWSSupport-CheckXenToNitroMigrationRequirements](#) (p. 79)
- [AWSSupport-ConfigureEC2Metadata](#) (p. 81)
- [AWSSupport-CopyEC2Instance](#) (p. 83)
- [AWSSupport-ExecuteEC2Rescue](#) (p. 87)
- [AWSSupport-ListEC2Resources](#) (p. 89)
- [AWSSupport-ManageRDPSettings](#) (p. 90)
- [AWSSupport-ManageWindowsService](#) (p. 92)
- [AWSSupport-MigrateEC2ClassicToVPC](#) (p. 93)
- [AWSSupport-MigrateXenToNitroLinux](#) (p. 98)
- [AWSSupport-ResetAccess](#) (p. 105)
- [AWSSupport-RestoreEC2InstanceFromSnapshot](#) (p. 107)
- [AWSSupport-SendLogBundleToS3Bucket](#) (p. 110)
- [AWSSupport-StartEC2RescueWorkflow](#) (p. 111)
- [AWSPremiumSupport-TroubleshootEC2DiskUsage](#) (p. 118)
- [AWSSupport-TroubleshootRDP](#) (p. 121)
- [AWSSupport-TroubleshootSSH](#) (p. 125)
- [AWSSupport-TroubleshootSUSERegistration](#) (p. 127)
- [AWSSupport-UpgradeWindowsAWSDrivers](#) (p. 128)

## AWS-ASGEnterStandby

### Description

Change the standby state of an Amazon Elastic Compute Cloud (Amazon EC2) instance in an Auto Scaling group.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your

behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) ID of an Amazon EC2 instance for which you want to change the standby state within an Auto Scaling group.

- LambdaRoleArn

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

## AWS-ASGExitStandby

### Description

Change the standby state of an Amazon Elastic Compute Cloud (Amazon EC2) instance in an Auto Scaling group.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) ID of an EC2 instance for which you want to change the standby state within an Auto Scaling group.

- LambdaRoleArn

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

## AWS-CreateImage

### Description

Create a new Amazon Machine Image (AMI) from an Amazon Elastic Compute Cloud (Amazon EC2) instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstancedId

Type: String

Description: (Required) The ID of the EC2 instance.

- NoReboot

Type: Boolean

Description: (Optional) Do not reboot the instance before creating the image.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateImage",
        "ec2:DescribeImages"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

```
} ]
```

## AWS-DeleteImage

### Description

Delete an Amazon Machine Image (AMI) and all associated snapshots.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ImageId

Type: String

Description: (Required) The ID of the AMI.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:DeleteSnapshot",
      "Resource": "arn:aws:ec2:{region}::snapshot/*"
    },
    {
      "Effect": "Allow",
      "Action": "ec2:DescribeImages",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "ec2:DeregisterImage",
```

```
        "Resource": "*"
    }
  ]
}
```

## AWSEC2-CloneInstanceAndUpgradeSQLServer

### Description

Create an AMI from an EC2 instance for Windows Server running SQL Server 2008 or later, and then upgrade the AMI to a later version of SQL Server.

The following upgrade paths are supported:

- SQL Server 2008 to SQL Server 2017, 2016, or 2014
- SQL Server 2008 R2 to SQL Server 2017, 2016, or 2014
- SQL Server 2012 to SQL Server 2019, 2017, 2016, or 2014
- SQL Server 2014 to SQL Server 2019, 2017, or 2016
- SQL Server 2016 to SQL Server 2019 or 2017
- SQL Server 2017 to SQL Server 2019

If you are using an earlier version of Windows Server that is incompatible with SQL Server 2019, the automation document must upgrade your Windows Server version to 2016.

The upgrade is a multi-step process that can take 2 hours to complete. The automation creates the AMI from the instance, and then launches a temporary instance from the new AMI in the specified `SubnetID`. The security groups associated with your original instance are applied to the temporary instance. The automation then performs an in-place upgrade to the `TargetSQLVersion` on the temporary instance. After the upgrade, the automation creates a new AMI from the temporary instance and then terminates the temporary instance.

You can test application functionality by launching the new AMI in your VPC. After you finish testing, and before you perform another upgrade, schedule application downtime before completely switching over to the upgraded instance.

#### Note

If you want to modify the computer name of the EC2 instance launched from the new AMI, see [Rename a Computer that Hosts a Stand-Alone Instance of SQL Server](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Windows

### Parameters

### Prerequisites

- TLS version 1.2.
- The EC2 instance must use a version of Windows Server that is Windows Server 2008 R2 (or later) and SQL Server 2008 (or later).
- Verify that SSM Agent is installed on your instance. For more information, see [Installing and configuring SSM Agent on EC2 instances for Windows Server](#).
- Configure the instance to use an AWS Identity and Access Management (IAM) instance profile role. For more information, see [Create an IAM instance profile for Systems Manager](#).
- Verify that the instance has 20 GB of free disk space in the instance boot disk.
- For instances that use a Bring Your Own License (BYOL) SQL Server version, the following additional prerequisites apply:
  - Provide an EBS snapshot ID that includes the target SQL Server installation media. To do this:
    1. Verify that the EC2 instance is running Windows Server 2008 R2 or later.
    2. Create a 6 GB EBS volume in the same Availability Zone where the instance is running. Attach the volume to the instance. Mount it, for example, as drive D.
    3. Right-click the ISO and mount it to an instance as, for example, drive E.
    4. Copy the content of the ISO from drive E:\ to drive D:\
    5. Create an EBS snapshot of the 6 GB volume created in step 2.

### Limitations

- The upgrade can be performed on only a SQL Server using Windows authentication.
- Verify that no security patch updates are pending on the instances. Open **Control Panel**, then choose **Check for updates**.
- SQL Server deployments in HA and mirroring mode are not supported.

### Parameters

- `IamInstanceProfile`

Type: String

Description: (Required) The IAM instance profile.

- `InstanceId`

Type: String

Description: (Required) The instance running Windows Server 2008 R2 (or later) and SQL Server 2008 (or later).

- `KeepPreUpgradeImageBackup`

Type: String

Description: (Optional) If set to `true`, the automation doesn't delete the AMI created from the instance before the upgrade. If set to `true`, then you must delete the AMI. By default, the AMI is deleted.

- `SubnetId`

Type: String

Description: (Required) Provide a subnet for the upgrade process. Verify that the subnet has outbound connectivity to AWS services, Amazon S3, and Microsoft (to download patches).

- `SQLServerSnapshotId`

Type: String



Description: (Conditional) Snapshot ID for target SQL Server installation media. This parameter is required for instances that use a BYOL SQL Server version. This parameter is optional for SQL Server license-included instances (instances launched using an AWS provided Amazon Machine Image for Windows Server with Microsoft SQL Server).

- `RebootInstanceBeforeTakingImage`

Type: String

Description: (Optional) If set to `true`, the automation reboots the instance before creating a pre-upgrade AMI. By default, the automation doesn't reboot before upgrade.

- `TargetSQLVersion`

Type: String

Description: (Optional) Select the target SQL Server version.

Possible targets:

- SQL Server 2019
- SQL Server 2017
- SQL Server 2016
- SQL Server 2014

Default target: SQL Server 2016

## Outputs

`AMIId`: The ID of the AMI created from the instance that was upgraded to a later version of SQL Server.

# AWSEC2-CloneInstanceAndUpgradeWindows

## Description

Create an Amazon Machine Image (AMI) from a Windows Server 2008 R2, 2012 R2, or 2016 instance, and then upgrade the AMI to Windows Server 2012 R2, 2016, or 2019. The supported upgrade paths are as follows.

- Windows Server 2008 R2 to Windows Server 2012 R2.
- Windows Server 2012 R2 to Windows Server 2016.
- Windows Server 2012 R2 to Windows Server 2019.
- Windows Server 2016 to Windows Server 2019.

To upgrade your instance from Windows Server 2008 R2 to Windows Server 2016 or 2019, the runbook performs two steps. The Windows Server 2008 R2 instance is upgraded to Windows Server 2012 R2. Then the Windows Server 2012 R2 instance is upgraded to the target version (Windows Server 2016 or 2019).

The upgrade operation is a multi-step process that can take 2 hours to complete. We recommend performing an operating system upgrade on instances with at least 2 vCPUs and 4GB of RAM. The automation creates an AMI from the instance and then launches a temporary instance from the newly created AMI in the `SubnetId` that you specify. The security groups associated with your original instance are applied to the temporary instance. The automation then performs an in-place upgrade to the `TargetWindowsVersion` on the temporary instance. To upgrade your Windows Server 2008 R2 instance to Windows Server 2016 or 2019, an in-place upgrade is performed twice because directly upgrading Windows Server 2008 R2 to Windows Server 2016 or 2019 is not supported. The automation

also updates or installs the AWS drivers required by the temporary instance. After the upgrade, the automation creates a new AMI from the temporary instance and then terminates the temporary instance.

You can test application functionality by launching a test instance from the upgraded AMI in your Amazon Virtual Private Cloud (Amazon VPC). After you finish testing, and before you perform another upgrade, schedule application downtime before completely switching over to the upgraded AMI.

[Run this Automation \(console\)](#)

## Document Type

Automation

## Owner

Amazon

## Platforms

Windows Server 2008 R2, 2012 R2 and 2016 Standard and Datacenter editions

## Prerequisites

- TLS version 1.2.
- Verify that SSM Agent is installed on your instance. For more information, see [Installing and configuring SSM Agent on EC2 instances for Windows Server](#).
- Windows PowerShell 3.0 or later must be installed on your instance.
- For instances that are joined to a Microsoft Active Directory domain, we recommend specifying a `SubnetId` that does not have connectivity to your domain controllers to help avoid hostname conflicts.
- The `SubnetId` specified must be a public subnet with the auto-assign public IPv4 address set to true. For more information, see [Modifying the Public IPv4 Addressing Attribute for Your Subnet](#) in the *Amazon VPC User Guide*.
- This Automation works only with Windows Server 2008 R2, 2012 R2, and 2016 instances.
- Configure the Windows Server instance with an AWS Identity and Access Management (IAM) instance profile that provides the requisite permissions for Systems Manager. For more information, see [Create an IAM instance profile for Systems Manager](#).
- Verify that the instance has 20 GB of free disk space in the boot disk.
- If the instance does not use an AWS-provided Windows license, then specify an Amazon EBS snapshot ID that includes Windows Server 2012 R2 installation media. To do this:
  - Verify that the EC2 instance is running Windows Server 2012 or later.
  - Create a 6 GB EBS volume in the same Availability Zone where the instance is running. Attach the volume to the instance. Mount it, for example, as drive D.
  - Right-click the ISO and mount it to an instance as, for example, drive E.
  - Copy the content of the ISO from drive E:\ to drive D:\
  - Create an EBS snapshot of the 6 GB volume created in step 2 above.

## Limitations

This Automation doesn't support upgrading Windows domain controllers, clusters, or Windows desktop operating systems. This Automation also doesn't support EC2 instances for Windows Server with the following roles installed.

- Remote Desktop Session Host (RDSH)

- Remote Desktop Connection Broker (RDCB)
- Remote Desktop Virtualization Host (RDVH)
- Remote Desktop Web Access (RDWA)

### Parameters

- AlternativeKeyPairName

Type: String

Description: (Optional) The name of an alternative key pair to use during the upgrade process. This is useful in situations where the key pair assigned to the original instance is unavailable. If the original instance was not assigned a key pair, you must specify a value for this parameter.

- BYOLWindowsMediaSnapshotId

Type: String

Description: (Optional) The ID of the Amazon EBS snapshot to copy that includes Windows Server 2012R2 installation media. Required only if you are upgrading a BYOL instance.

- iamInstanceProfile

Type: String

Description: (Required) The name of the IAM instance profile that enables Systems Manager to manage the instance.

- InstanceId

Type: String

Description: (Required) The EC2 instance running Windows Server 2008 R2 or 2012 R2.

- KeepPreUpgradeImageBackUp

Type: String

Description: (Optional) If set True, the Automation doesn't delete the AMI created from the EC2 instance before the upgrade. If set to True, then you must delete the AMI. By default, the AMI is deleted.

- SubnetId

Type: String

Description: (Required) This is the subnet for the upgrade process and where your source EC2 instance resides. Verify that the subnet has outbound connectivity to AWS services, Amazon S3, and Microsoft (to download patches).

- TargetWindowsVersion

Type: String

Description: (Required) Select the target Windows version.

Default: 2012R2

- RebootInstanceBeforeTakingImage

Type: String

Description: (Optional) If set True, the Automation reboots the instance before creating a pre-upgrade AMI. By default, the Automation doesn't reboot before upgrade.

## AWSEC2-ConfigureSTIG

Security Technical Implementation Guides (STIGs) are the configuration standards created by the Defense Information Systems Agency (DISA) to secure information systems and software. To make your systems compliant with STIG standards, you must install, configure, and test a variety of security settings.

Amazon EC2 provides a Systems Manager document, AWSEC2-ConfigureSTIG, which you can use to apply STIG to an instance. This document helps you to quickly build compliant images for STIG standards. The STIG Systems Manager document scans for misconfigurations and runs a remediation script. The STIG Systems Manager document installs InstallRoot on Windows AMIs from the Department of Defense (DoD) to install and update the DoD certificates and to remove unnecessary certificates to maintain STIG compliance. There are no additional charges for using the STIG Systems Manager document.

You can choose which STIG compliance category to apply.

### Compliance levels

- **High (Category I)**

The most severe risk. Includes any vulnerability that can result in loss of confidentiality, availability, or integrity.

- **Medium (Category II)**

Includes any vulnerability that can result in loss of confidentiality, availability, or integrity but the risk can be mitigated.

- **Low (Category III)**

Includes any vulnerability that degrades measures to protect against loss of confidentiality, availability, or integrity.

### Topics

- [Windows STIG settings \(p. 61\)](#)
- [Linux STIG settings \(p. 65\)](#)
- [Version history \(p. 68\)](#)

## Windows STIG settings

Amazon EC2 Windows STIG AMIs and components are designed for standalone servers and apply Local Group Policy. STIG-compliant components install InstallRoot from the Department of Defense (DoD) on Windows AMIs to download, install and update the DoD certificates. They also remove unnecessary certificates to maintain STIG compliance. Currently, STIG baselines are supported for the following versions of Windows Server: 2012 R2, 2016, and 2019.

You can apply low, medium, or high STIG settings.

### Windows STIG Low (Category III)

The following list contains STIG settings that apply to your infrastructure. If a setting isn't applicable for your infrastructure, Amazon EC2 skips that setting, and moves on. For example, some STIG settings might not apply to standalone servers. Organization-specific policies can also affect which settings apply, such as a requirement for administrators to review document settings. For more details about the STIGs that apply to Windows AMIs, you can download our [spreadsheet](#).

For a complete list of Windows STIGs, see the [STIGs Document Library](#). For information about how to view the complete list, see [How to View SRGs and STIGs](#).

- **Windows Server 2019 STIG Version 2 Release 4**

V-205691, V-205819, V-205858, V-205859, V-205860, V-205870, V-205871, and V-205923

- **Windows Server 2016 STIG Version 2 Release 4**

V-224916, V-224917, V-224918, V-224919, V-224931, V-224942, and V-225060

- **Windows Server 2012 R2 MS STIG Version 3 Release 3**

V-225537, V-225536, V-225526, V-225525, V-225514, V-225511, V-225490, V-225489, V-225488, V-225487, V-225485, V-225484, V-225483, V-225482, V-225481, V-225480, V-225479, V-225476, V-225473, V-225468, V-225462, V-225460, V-225459, V-225412, V-225394, V-225392, V-225376, V-225363, V-225362, V-225360, V-225359, V-225358, V-225357, V-225355, V-225343, V-225342, V-225336, V-225335, V-225334, V-225333, V-225332, V-225331, V-225330, V-225328, V-225327, V-225324, V-225319, V-225318, and V-225250

- **Microsoft .NET Framework 4.0 STIG Version 2 Release 1**

No STIG settings apply to the Microsoft .NET Framework for Category III vulnerabilities.

- **Windows Firewall STIG Version 2 Release 1**

V-241994, V-241995, V-241996, V-241999, V-242000, V-242001, V-242006, V-242007, and V-242008

- **Internet Explorer 11 STIG Version 1 Release 19**

V-46477, V-46629, and V-97527

## Windows STIG Medium (Category II)

The following list contains STIG settings that apply to your infrastructure. If a setting isn't applicable for your infrastructure, Amazon EC2 skips that setting, and moves on. For example, some STIG settings might not apply to standalone servers. Organization-specific policies can also affect which settings apply, such as a requirement for administrators to review document settings. For more details about the STIGs that apply to Windows AMIs, you can download our [spreadsheet](#).

For a complete list of Windows STIGs, see the [STIGs Document Library](#). For information about how to view the complete list, see [How to View SRGs and STIGs](#).

### Note

The Windows STIG Medium category includes all of the STIG settings that apply to Windows STIG low (Category III), in addition to the STIG settings that apply specifically for Category II vulnerabilities.

- **Windows Server 2019 STIG Version 2 Release 4**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

V-205625, V-205626, V-205627, V-205629, V-205630, V-205633, V-205634, V-205635, V-205636, V-205637, V-205638, V-205639, V-205643, V-205644, V-205648, V-205649, V-205650, V-205651, V-205652, V-205655, V-205656, V-205659, V-205660, V-205662, V-205671, V-205672, V-205673, V-205675, V-205676, V-205678, V-205679, V-205680, V-205681, V-205682, V-205683, V-205684, V-205685, V-205686, V-205687, V-205688, V-205689, V-205690, V-205692, V-205693, V-205694, V-205697, V-205698, V-205708, V-205709, V-205712, V-205714, V-205716, V-205717, V-205718, V-205719, V-205720, V-205722, V-205729, V-205730, V-205733, V-205747, V-205751, V-205752, V-205754, V-205756, V-205758, V-205759, V-205760, V-205761, V-205762, V-205764, V-205765, V-205766, V-205767, V-205768, V-205769, V-205770, V-205771, V-205772, V-205773, V-205774, V-205775, V-205776, V-205777, V-205778, V-205779, V-205780, V-205781, V-205782, V-205783,

V-205784, V-205795, V-205796, V-205797, V-205798, V-205801, V-205808, V-205809, V-205810, V-205811, V-205812, V-205813, V-205814, V-205815, V-205816, V-205817, V-205821, V-205822, V-205823, V-205824, V-205825, V-205826, V-205827, V-205828, V-205830, V-205832, V-205833, V-205834, V-205835, V-205836, V-205837, V-205838, V-205839, V-205840, V-205841, V-205861, V-205863, V-205865, V-205866, V-205867, V-205868, V-205869, V-205872, V-205873, V-205874, V-205911, V-205912, V-205915, V-205916, V-205917, V-205918, V-205920, V-205921, V-205922, V-205924, V-205925, and V-236001

- **Windows Server 2016 STIG Version 2 Release 4**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

V-224850, V-224852, V-224853, V-224854, V-224855, V-224856, V-224857, V-224858, V-224859, V-224866, V-224867, V-224868, V-224869, V-224870, V-224871, V-224872, V-224873, V-224881, V-224882, V-224883, V-224884, V-224885, V-224886, V-224887, V-224888, V-224889, V-224890, V-224891, V-224892, V-224893, V-224894, V-224895, V-224896, V-224897, V-224898, V-224899, V-224900, V-224901, V-224902, V-224903, V-224904, V-224905, V-224906, V-224907, V-224908, V-224909, V-224910, V-224911, V-224912, V-224913, V-224914, V-224915, V-224920, V-224922, V-224924, V-224925, V-224926, V-224927, V-224928, V-224929, V-224930, V-224935, V-224936, V-224937, V-224938, V-224939, V-224940, V-224941, V-224943, V-224944, V-224945, V-224946, V-224947, V-224948, V-224949, V-224951, V-224952, V-224953, V-224955, V-224956, V-224957, V-224959, V-224960, V-224962, V-224963, V-225010, V-225013, V-225014, V-225015, V-225016, V-225017, V-225018, V-225019, V-225021, V-225022, V-225023, V-225024, V-225028, V-225029, V-225030, V-225031, V-225032, V-225033, V-225034, V-225035, V-225038, V-225039, V-225040, V-225041, V-225042, V-225043, V-225047, V-225049, V-225050, V-225051, V-225052, V-225055, V-225056, V-225057, V-225058, V-225061, V-225062, V-225063, V-225064, V-225065, V-225066, V-225067, V-225068, V-225069, V-225072, V-225073, V-225074, V-225076, V-225078, V-225080, V-225081, V-225082, V-225083, V-225084, V-225086, V-225087, V-225088, V-225089, V-225092, V-225093 and V-236000

- **Windows Server 2012 R2 MS STIG Version 3 Release 3**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

V-225574, V-225573, V-225572, V-225571, V-225570, V-225569, V-225568, V-225567, V-225566, V-225565, V-225564, V-225563, V-225562, V-225561, V-225560, V-225559, V-225558, V-225557, V-225555, V-225554, V-225553, V-225551, V-225550, V-225549, V-225548, V-225546, V-225545, V-225544, V-225543, V-225542, V-225541, V-225540, V-225539, V-225538, V-225535, V-225534, V-225533, V-225532, V-225531, V-225530, V-225529, V-225528, V-225527, V-225524, V-225523, V-225522, V-225521, V-225520, V-225519, V-225518, V-225517, V-225516, V-225515, V-225513, V-225510, V-225509, V-225508, V-225506, V-225504, V-225503, V-225502, V-225501, V-225500, V-225494, V-225486, V-225478, V-225477, V-225475, V-225474, V-225472, V-225471, V-225470, V-225469, V-225464, V-225463, V-225461, V-225458, V-225457, V-225456, V-225455, V-225454, V-225453, V-225452, V-225448, V-225443, V-225442, V-225441, V-225415, V-225414, V-225413, V-225411, V-225410, V-225409, V-225408, V-225407, V-225406, V-225405, V-225404, V-225402, V-225401, V-225400, V-225398, V-225397, V-225395, V-225393, V-225391, V-225389, V-225386, V-225385, V-225384, V-225383, V-225382, V-225381, V-225380, V-225379, V-225378, V-225377, V-225375, V-225374, V-225373, V-225372, V-225371, V-225370, V-225369, V-225368, V-225367, V-225356, V-225353, V-225352, V-225351, V-225350, V-225349, V-225348, V-225347, V-225346, V-225345, V-225344, V-225341, V-225340, V-225339, V-225338, V-225337, V-225329, V-225326, V-225325, V-225317, V-225316, V-225315, V-225314, V-225305, V-225304, V-225303, V-225302, V-225301, V-225300, V-225299, V-225298, V-225297, V-225296, V-225295, V-225294, V-225293, V-225292, V-225291, V-225290, V-225289, V-225288, V-225287, V-225286, V-225285, V-225284, V-225283, V-225282, V-225281, V-225280, V-225279, V-225278, V-225277, V-225276, V-225275, V-225273, V-225272, V-225271, V-225270, V-225269, V-225268, V-225267, V-225266, V-225265, V-225264, V-225263, V-225261, V-225260, V-225259, and V-225239

- **Microsoft .NET Framework STIG 4.0 Version 2 Release 1**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

V-225238

- **Windows Firewall STIG Version 2 Release 1**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

V-241989, V-241990, V-241991, V-241993, V-241998, and V-242003

- **Internet Explorer 11 STIG Version 1 Release 19**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

V-46473, V-46475, V-46481, V-46483, V-46501, V-46507, V-46509, V-46511, V-46513, V-46515, V-46517, V-46521, V-46523, V-46525, V-46543, V-46545, V-46547, V-46549, V-46553, V-46555, V-46573, V-46575, V-46577, V-46579, V-46581, V-46583, V-46587, V-46589, V-46591, V-46593, V-46597, V-46599, V-46601, V-46603, V-46605, V-46607, V-46609, V-46615, V-46617, V-46619, V-46621, V-46625, V-46633, V-46635, V-46637, V-46639, V-46641, V-46643, V-46645, V-46647, V-46649, V-46653, V-46663, V-46665, V-46669, V-46681, V-46685, V-46689, V-46691, V-46693, V-46695, V-46701, V-46705, V-46709, V-46711, V-46713, V-46715, V-46717, V-46719, V-46721, V-46723, V-46725, V-46727, V-46729, V-46731, V-46733, V-46779, V-46781, V-46787, V-46789, V-46791, V-46797, V-46799, V-46801, V-46807, V-46811, V-46815, V-46819, V-46829, V-46841, V-46847, V-46849, V-46853, V-46857, V-46859, V-46861, V-46865, V-46869, V-46879, V-46883, V-46885, V-46889, V-46893, V-46895, V-46897, V-46903, V-46907, V-46921, V-46927, V-46939, V-46975, V-46981, V-46987, V-46995, V-46997, V-46999, V-47003, V-47005, V-47009, V-64711, V-64713, V-64715, V-64717, V-64719, V-64721, V-64723, V-64725, V-64729, V-72757, V-72759, V-72761, V-72763, V-75169, and V-75171

## Windows STIG High (Category I)

The following list contains STIG settings that apply to your infrastructure. If a setting isn't applicable for your infrastructure, Amazon EC2 skips that setting, and moves on. For example, some STIG settings might not apply to standalone servers. Organization-specific policies can also affect which settings apply, such as a requirement for administrators to review document settings. For more details about the STIGs that apply to Windows AMIs, you can download our [spreadsheet](#).

For a complete list of Windows STIGs, see the [STIGs Document Library](#). For information about how to view the complete list, see [How to View SRGs and STIGs](#).

### Note

The Windows STIG High category includes all of the STIG settings that apply for Windows STIG Medium and Low categories, in addition to the STIG settings that apply specifically to Category I vulnerabilities.

- **Windows Server 2019 STIG Version 2 Release 4**

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities, plus:

V-205653, V-205654, V-205711, V-205713, V-205724, V-205725, V-205757, V-205802, V-205804, V-205805, V-205806, V-205849, V-205908, V-205913, V-205914, and V-205919

- **Windows Server 2016 STIG Version 2 Release 4**

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities, plus:

V-224874, V-224932, V-224933, V-224934, V-224954, V-224958, V-224961, V-225025, V-225044, V-225045, V-225046, V-225048, V-225053, V-225054, and V-225079

- **Windows Server 2012 R2 MS STIG Version 3 Release 3**

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities, plus:

V-225556, V-225552, V-225547, V-225507, V-225505, V-225498, V-225497, V-225496, V-225493, V-225492, V-225491, V-225449, V-225444, V-225399, V-225396, V-225390, V-225366, V-225365, V-225364, V-225354, and V-225274

- **Microsoft .NET Framework STIG 4.0 Version 2 Release 1**

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities for the Microsoft .NET Framework. No additional STIG settings apply for Category I vulnerabilities.

- **Windows Firewall STIG Version 2 Release 1**

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities, plus:

V-241992, V-241997, and V-242002

- **Internet Explorer 11 STIG Version 1 Release 19**

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities for Internet Explorer 11. No additional STIG settings apply for Category I vulnerabilities.

## Linux STIG settings

This section contains information about Linux STIG settings. If the Linux distribution doesn't have STIG settings of its own, Amazon EC2 uses RHEL settings. STIG settings apply to Amazon EC2 Linux AMIs and components based on the Linux distribution, as follows:

- Red Hat Enterprise Linux (RHEL) 7 STIG settings
  - RHEL 7
  - CentOS 7
  - Amazon Linux 2 (AL2)
- RHEL 8 STIG settings
  - RHEL 8
  - CentOS 8

### Linux STIG Low (Category III)

The following list contains STIG settings that apply to your infrastructure. If a setting isn't applicable for your infrastructure, Amazon EC2 skips that setting, and moves on. For example, some STIG settings might not apply to standalone servers. Organization-specific policies can also affect which settings apply, such as a requirement for administrators to review document settings. For more details about the STIGs that apply to Linux AMIs, you can download our [spreadsheet](#).

For a complete list, see the [STIGs Document Library](#). For information about how to view the complete list, see [How to View SRGs and STIGs](#).

#### **RHEL 7 STIG Version 3 Release 7**

- **RHEL 7/CentOS 7**

V-204452, V-204576, and V-204605

- **AL2**

V-204452, V-204576, and V-204605

#### **RHEL 8 STIG Version 1 Release 6**



- **RHEL 8/CentOS 8**

V-230241, V-230253, V-230269, V-230270, V-230281, V-230285, V-230346, V-230381, V-230395, V-230468, V-230469, V-230485, V-230486, V-230491, V-230494, V-230495, V-230496, V-230497, V-230498, V-230499, and V-244527

- **Ubuntu 18.04 STIG Version 2 Release 6**

V-219163, V-219164, V-219165, V-219172, V-219173, V-219174, V-219175, V-219178, V-219180, V-219210, V-219301, V-219327, V-219332, and V-219333

- **Ubuntu 20.04 STIG Version 1 Release 4**

V-238202, V-238221, V-238222, V-238223, V-23822, V-238226, V-238234, V-238235, V-238237, V-238308, V-238323, V-238357, V-238362, and V-238373

## Linux STIG Medium (Category II)

The following list contains STIG settings that apply to your infrastructure. If a setting isn't applicable for your infrastructure, Amazon EC2 skips that setting, and moves on. For example, some STIG settings might not apply to standalone servers. Organization-specific policies can also affect which settings apply, such as a requirement for administrators to review document settings. For more details about the STIGs that apply to Linux AMIs, you can download our [spreadsheet](#).

For a complete list, see the [STIGs Document Library](#). For information about how to view the complete list, see [How to View SRGs and STIGs](#).

### Note

The Linux STIG Medium category includes all of the STIG settings that apply for Linux STIG Low (Category III), in addition to the STIG settings that apply specifically for Category II vulnerabilities.

- **RHEL 7 STIG Version 3 Release 7**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

- **RHEL 7/CentOS 7**

V-204405, V-204406, V-204407, V-204408, V-204409, V-204410, V-204411, V-204412, V-204413, V-204414, V-204415, V-204416, V-204417, V-204418, V-204422, V-204423, V-204426, V-204427, V-204428, V-204431, V-204435, V-204437, V-204449, V-204450, V-204451, V-204457, V-204466, V-204503, V-204516, V-204517, V-204521, V-204524, V-204531, V-204536, V-204537, V-204538, V-204539, V-204540, V-204541, V-204542, V-204543, V-204544, V-204545, V-204546, V-204547, V-204548, V-204549, V-204550, V-204551, V-204552, V-204553, V-204554, V-204555, V-204556, V-204557, V-204558, V-204559, V-204560, V-204562, V-204563, V-204564, V-204565, V-204566, V-204567, V-204568, V-204572, V-204579, V-204584, V-204585, V-204586, V-204587, V-204589, V-204590, V-204591, V-204592, V-204593, V-204598, V-204599, V-204600, V-204601, V-204602, V-204609, V-204610, V-204611, V-204612, V-204613, V-204614, V-204615, V-204616, V-204617, V-204619, V-204622, V-204624, V-204625, V-204630, V-204631, V-204633, V-233307, V-237634, V-237635, and V-251703

- **AL2:**

V-204405, V-204406, V-204407, V-204408, V-204409, V-204410, V-204411, V-204412, V-204413, V-204414, V-204415, V-204416, V-204417, V-204418, V-204422, V-204423, V-204426, V-204427, V-204428, V-204431, V-204435, V-204437, V-204449, V-204450, V-204451, V-204457, V-204466, V-204503, V-204516, V-204517, V-204521, V-204524, V-204531, V-204536, V-204537, V-204538, V-204539, V-204540, V-204541, V-204542, V-204543, V-204544, V-204545, V-204546, V-204547, V-204548, V-204549, V-204550, V-204551, V-204552, V-204553, V-204554, V-204555, V-204556,

V-204557, V-204558, V-204559, V-204560, V-204562, V-204563, V-204564, V-204565, V-204566, V-204567, V-204568, V-204572, V-204578, V-204579, V-204584, V-204585, V-204586, V-204587, V-204589, V-204590, V-204591, V-204592, V-204593, V-204595, V-204598, V-204599, V-204600, V-204601, V-204602, V-204609, V-204610, V-204611, V-204612, V-204613, V-204614, V-204615, V-204616, V-204617, V-204619, V-204622, V-204624, V-204625, V-204630, V-204631, V-204633, V-233307, V-237634, V-237635, and V-251703

#### **RHEL 8 STIG Version 1 Release 6**

Includes all STIG settings that apply for Category III (Low) vulnerabilities, plus:

- **RHEL 8/CentOS 8**

V-230228, V-230231, V-230233, V-230236, V-230237, V-230239, V-230240, V-230244, V-230255, V-230266, V-230267, V-230268, V-230273, V-230275, V-230277, V-230278, V-230279, V-230280, V-230282, V-230288, V-230289, V-230290, V-230291, V-230296, V-230298, V-230310, V-230311, V-230312, V-230313, V-230314, V-230315, V-230324, V-230330, V-230332, V-230333, V-230334, V-230335, V-230336, V-230337, V-230338, V-230339, V-230340, V-230341, V-230342, V-230343, V-230344, V-230345, V-230348, V-230349, V-230353, V-230356, V-230357, V-230358, V-230359, V-230360, V-230361, V-230362, V-230363, V-230365, V-230368, V-230369, V-230370, V-230375, V-230377, V-230378, V-230382, V-230383, V-230386, V-230387, V-230390, V-230392, V-230402, V-230403, V-230404, V-230405, V-230406, V-230407, V-230408, V-230409, V-230410, V-230411, V-230412, V-230413, V-230418, V-230419, V-230421, V-230422, V-230423, V-230424, V-230425, V-230426, V-230427, V-230428, V-230429, V-230430, V-230431, V-230432, V-230433, V-230434, V-230435, V-230436, V-230437, V-230438, V-230439, V-230444, V-230446, V-230447, V-230448, V-230449, V-230455, V-230456, V-230462, V-230463, V-230464, V-230465, V-230466, V-230467, V-230478, V-230480, V-230488, V-230489, V-230502, V-230503, V-230526, V-230527, V-230532, V-230535, V-230536, V-230537, V-230538, V-230539, V-230540, V-230541, V-230542, V-230543, V-230544, V-230545, V-230546, V-230547, V-230548, V-230549, V-230555, V-230556, V-230559, V-230560, V-230561, V-237640, V-237642, V-237643, V-244523, V-244524, V-244525, V-244526, V-244528, V-244533, V-244534, V-244537, V-244542, V-244549, V-244550, V-244551, V-244552, V-244553, V-244554, V-250317, V-251711, V-251713, V-251714, V-251715, V-251716, V-251717, and V-251718

#### **Ubuntu 18.04 STIG Version 2 Release 6**

V-219149, V-219155, V-219156, V-219160, V-219166, V-219176, V-219181, V-219184, V-219186, V-219187, V-219188, V-219189, V-219190, V-219191, V-219192, V-219193, V-219194, V-219195, V-219196, V-219197, V-219198, V-219199, V-219200, V-219201, V-219202, V-219203, V-219204, V-219205, V-219206, V-219207, V-219208, V-219209, V-219213, V-219214, V-219215, V-219216, V-219217, V-219218, V-219219, V-219220, V-219221, V-219222, V-219223, V-219224, V-219227, V-219228, V-219229, V-219230, V-219231, V-219232, V-219233, V-219234, V-219235, V-219236, V-219238, V-219239, V-219240, V-219241, V-219242, V-219243, V-219244, V-219250, V-219254, V-219257, V-219263, V-219264, V-219265, V-219266, V-219267, V-219268, V-219269, V-219270, V-219271, V-219272, V-219273, V-219274, V-219275, V-219276, V-219277, V-219279, V-219281, V-219287, V-219291, V-219297, V-219298, V-219299, V-219300, V-219303, V-219306, V-219309, V-219310, V-219311, V-219312, V-219315, V-219326, V-219328, V-219330, V-219331, V-219334, V-219335, V-219336, V-219337, V-219338, V-219339, V-219342, V-233779, V-233780, V-237768, V-237769, and V-237770

#### **Ubuntu 20.04 STIG Version 1 Release 4**

V-238200, V-238205, V-238207, V-238209, V-238211, V-238212, V-238213, V-238216, V-238220, V-238225, V-238227, V-238228, V-238230, V-238231, V-238236, V-238238, V-238239, V-238240, V-238241, V-238242, V-238244, V-238245, V-238246, V-238247, V-238248, V-238249, V-238250, V-238251, V-238252, V-238253, V-238254, V-238255, V-238256, V-238257, V-238258, V-238264, V-238268, V-238271, V-238277, V-238278, V-238279, V-238280, V-238281, V-238282, V-238283,

V-238284, V-238285, V-238286, V-238287, V-238288, V-238289, V-238290, V-238291, V-238292, V-238293, V-238294, V-238295, V-238297, V-238299, V-238300, V-238301, V-238302, V-238303, V-238304, V-238309, V-238310, V-238314, V-238315, V-238316, V-238317, V-238318, V-238319, V-238320, V-238324, V-238325, V-238329, V-238330, V-238332, V-238333, V-238334, V-238335, V-238337, V-238338, V-238339, V-238340, V-238341, V-238342, V-238343, V-238344, V-238345, V-238346, V-238347, V-238348, V-238349, V-238350, V-238351, V-238352, V-238353, V-238356, V-238358, V-238359, V-238360, V-238369, V-238370, V-238376, V-238377, V-238378, and V-251505

## Linux STIG High (Category I)

The following list contains STIG settings that apply to your infrastructure. If a setting isn't applicable for your infrastructure, Amazon EC2 skips that setting, and moves on. For example, some STIG settings might not apply to standalone servers. Organization-specific policies can also affect which settings apply, such as a requirement for administrators to review document settings. For more details about the STIGs that apply to Linux AMIs, you can download our [spreadsheet](#).

For a complete list, see the [STIGs Document Library](#). For information about how to view the complete list, see [How to View SRGs and STIGs](#).

### Note

The Linux STIG High category includes all of the STIG settings that apply for Linux STIG Medium and Low categories, in addition to the STIG settings that apply specifically for Category I vulnerabilities.

### RHEL 7 STIG Version 3 Release 7

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities, plus:

- **RHEL 7/CentOS 7**

V-204425, V-204442, V-204443, V-204447, V-204448, V-204455, V-204502, V-204620, and V-204621

- **AL2:**

V-204425, V-204442, V-204443, V-204447, V-204448, V-204455, V-204502, V-204620, and V-204621

### RHEL 8 STIG Version 1 Release 6

Includes all STIG settings that apply for Categories II and III (Medium and Low) vulnerabilities, plus:

- **RHEL 8/CentOS 8**

V-230264, V-230265, V-230487, V-230492, V-230529, V-230531, V-230533, and V-230558

### Ubuntu 18.04 STIG Version 2 Release 6

V-219157, V-219158, V-219177, V-219212, V-219308, V-219313, V-219314, V-219316, V-251506, and V-251507

### Ubuntu 20.04 STIG Version 1 Release 4

V-238201, V-238215, V-238218, V-238219, V-238326, V-238327, V-238380, V-251503, and V-251504

## Version history

The following table describes important changes to the STIG environment and settings described in this document.

Change	Description	Date
Quarterly updates	Introduced Ubuntu support, updated STIG versions, and applied STIGS for 2022 second quarter release.	08/02/2022
Quarterly updates	Updated STIG settings for 2022 Q1.	04/26/2022
Environment updates	Changed staging directory and adjusted permissions.	02/01/2022
Quarterly updates	Updated STIG settings for 2021 Q4.	12/20/2021
Quarterly updates	Updated STIG settings for 2021 Q3.	09/30/2021
Quarterly updates	Updated STIG settings for 2021 Q2.	08/06/2021
Prior changes	The STIG Systems Manager document, AWSEC2-ConfigureSTIG, was introduced in May, 2020. Since then, it has received quarterly updates to reflect changes in official STIG guidelines.	05/06/2020

## AWSEC2-PatchLoadBalancerInstance

### Description

Upgrade and patch minor version of an Amazon EC2 instance (Windows or Linux) attached to any load balancer (classic, ALB, or NLB). The default connection draining time is applied before the instance is patched. You can override the wait time by entering your custom draining time in minutes (1-59) for the **ConnectionDrainTime** parameter.

The automation workflow is as follows:

1. The load balancer or target group to which the instance is attached is determined, and the instance is verified as healthy.
2. The instance is removed from the load balancer or target group.
3. The automation waits for the period of time specified for the connection draining time.
4. The [AWS-RunPatchBaseline](#) automation is called to patch the instance.
5. The instance is reattached to the load balancer or target group.

[Run this Automation \(console\)](#)

### Document Type

Automation

### Owner

Amazon

### Prerequisites

- Verify that SSM Agent is installed on your instance. For more information, see [Working with SSM Agent on EC2 instances for Windows Server](#).

#### Parameters

- **InstancedId**

Type: String

Description: (Required) ID of the instance to patch that is associated with a load balancer (classic, ALB, or NLB).

- **ConnectionDrainTime**

Type: String

Description: (Optional) The connection draining time of the load balancer, in minutes (1-59).

## AWSEC2-SQLServerDBRestore

#### Description

The `AWSEC2-SQLServerDBRestore` runbook restores Microsoft SQL Server database backups stored in Amazon S3 to SQL Server 2017 running on an Amazon Elastic Compute Cloud (EC2) Linux instance. You may provide your own EC2 instance running SQL Server 2017 Linux. If an EC2 instance is not provided, the automation launches and configures a new Ubuntu 16.04 EC2 instance with SQL Server 2017. The automation supports restoring full, differential, and transactional log backups. This automation accepts multiple database backup files and automatically restores the most recent valid backup of each database in the files provided.

To automate both backup and restore of an on-premises SQL Server database to an EC2 instance running SQL Server 2017 Linux, you can use the AWS-signed PowerShell script [MigrateSQLServerToEC2Linux](#).

#### Important

This runbook resets the SQL Server server administrator (SA) user password every time the automation runs. After the automation is complete, you must set your own SA user password again before you connect to the SQL Server instance.

[Run this Automation \(console\)](#)

#### Document Type

Automation

#### Owner

Amazon

#### Platforms

Linux

## Prerequisites

To run this automation, you must meet the following prerequisites:

- The IAM user or role that runs this automation must have an inline policy attached with the permissions outlined in [Required IAM permissions \(p. 72\)](#).

- If you provide your own EC2 instance:
  - The EC2 instance that you provide must be a Linux instance running Microsoft SQL Server 2017.
  - The EC2 instance that you provide must be configured with an AWS Identity and Access Management (IAM) instance profile that has the `AmazonSSMManagedInstanceCore` managed policy attached. For more information, see [Create an IAM instance profile for Systems Manager](#).
  - The SSM Agent must be installed on your EC2 instance. For more information, see [Installing and configuring SSM Agent on EC2 instances for Linux](#).
  - The EC2 instance must have enough free disk space to download and restore the SQL Server backups.

## Limitations

This automation does not support restoring to SQL Server running on EC2 instances for Windows Server. This automation only restores database backups that are compatible with SQL Server Linux 2017. For more information, see [Editions and Supported Features of SQL Server 2017 on Linux](#).

## Parameters

This automation has the following parameters:

- **DatabaseNames**

Type: String

Description: (Optional) Comma-separated list of the names of databases to restore.

- **DataDirectorySize**

Type: String

Description: (Optional) Desired volume size (GiB) of the SQL Server Data directory for the new EC2 instance.

Default value: 100

- **KeyPair**

Type: String

Description: (Optional) Key pair to use when creating the new EC2 instance.

- **IamInstanceProfileName**

Type: String

Description: (Optional) The IAM instance profile to attach to the new EC2 instance. The IAM instance profile must have the `AmazonSSMManagedInstanceCore` managed policy attached.

- **InstanceId**

Type: String

Description: (Optional) The instance running SQL Server 2017 on Linux. If no InstanceId is provided, the automation launches a new EC2 instance using the InstanceType and SQLServerEdition provided.

- **InstanceType**

Type: String

Description: (Optional) The instance type of the EC2 instance to be launched.

- **IsS3PresignedUrl**

Type: String

Description: (Optional) If S3Input is a pre-signed S3 URL, indicate yes.

Default value: no

Valid values: yes | no

- **LogDirectorySize**

Type: String

Description: (Optional) Desired volume size (GiB) of the SQL Server Log directory for the new EC2 instance.

Default value: 100

- **S3Input**

Type: String

Description: (Required) S3 bucket name, comma-separated list of S3 object keys, or comma-separated list of pre-signed S3 URLs containing the SQL backup files to be restored.

- **SQLServerEdition**

Type: String

Description: (Optional) The edition of SQL Server 2017 to be installed on the newly created EC2 instance.

Valid values: Standard | Enterprise | Web | Express

- **SubnetId**

Type: String

Description: (Optional) The subnet in which to launch the new EC2 instance. The subnet must have outbound connectivity to AWS services. If a value for SubnetId is not provided, the automation uses the default subnet.

- **TempDbDirectorySize**

Type: String

Description: (Optional) Desired volume size (GiB) of the SQL Server TempDB directory for the new EC2 instance.

Default value: 100

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
```

```
        "ec2:DescribeImages",
        "ec2:DescribeInstanceStatus",
        "ec2:DescribeInstances",
        "ec2:RebootInstances",
        "ec2:RunInstances",
        "ssm:DescribeInstanceInformation",
        "ssm:GetAutomationExecution",
        "ssm:ListCommandInvocations",
        "ssm:ListCommands",
        "ssm:SendCommand",
        "ssm:StartAutomationExecution"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "arn:aws:iam::ACCOUNTID:role/ROLENAME"
  }
]
}
```

## Document Steps

To use this automation, follow the steps that apply to your instance type:

### For new EC2 instances:

1. `aws:executeAwsApi` - Retrieve the AMI ID for SQL Server 2017 on Ubuntu 16.04.
2. `aws:runInstances` - Launch a new EC2 instance for Linux.
3. `aws:waitForAwsResourceProperty` - Wait for the newly created EC2 instance to be ready.
4. `aws:executeAwsApi` - Reboot the instance if the instance is not ready.
5. `aws:assertAwsResourceProperty` - Verify that SSM Agent is installed.
6. `aws:runCommand` - Run the SQL Server restore script in PowerShell.

### For existing EC2 instances:

1. `aws:waitForAwsResourceProperty` - Verify that the EC2 instance is ready.
2. `aws:executeAwsApi` - Reboot the instance if the instance is not ready.
3. `aws:assertAwsResourceProperty` - Verify that SSM Agent is installed.
4. `aws:runCommand` - Run the SQL Server restore script in PowerShell.

### Outputs

`getInstance.InstanceId`

`restoreToNewInstance.Output`

`restoreToExistingInstance.Output`

## AWSSupport- ActivateWindowsWithAmazonLicense

### Description



The `AWSSupport-ActivateWindowsWithAmazonLicense` runbook activates an Amazon Elastic Compute Cloud (Amazon EC2) instance for Windows Server with a license provided by Amazon. The automation verifies and configures required key management service operating system settings and attempts activation. This includes operating system routes to Amazon's key management servers and key management service operating system settings. Setting the `AllowOffline` parameter to `true` allows the automation to successfully target instances that are not managed by AWS Systems Manager, but requires a stop and start of the instance.

**Note**

This runbook cannot be used on Bring Your Own License (BYOL) model Windows Server instances. For information about using your own license, see [Microsoft Licensing on AWS](#).

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Windows

**Parameters**

- `AllowOffline`

Type: String

Valid values: `true` | `false`

Default: `false`

Description: (Optional) Set it to `true` if you allow an offline Windows activation remediation in case the online troubleshooting fails, or if the provided instance is not a managed instance.

**Important**

The offline method requires that the provided EC2 instance be stopped and then started. Data stored in instance store volumes will be lost. The public IP address will change if you are not using an Elastic IP.

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `ForceActivation`

Type: String

Valid values: `true` | `false`

Default: `false`

Description: (Optional) Set it to `true` if you want to proceed even if Windows is already activated.

- `InstanceId`

Type: String

Description: (Required) ID of your managed EC2 instance for Windows Server.

- SubnetId

Type: String

Default: CreateNewVPC

Description: (Optional) Offline only - The subnet ID for the EC2Rescue instance used to perform the offline troubleshooting. Use `SelectedInstanceSubnet` to use the same subnet as your instance, or use `CreateNewVPC` to create a new VPC. **IMPORTANT:** The subnet must be in the same Availability Zone as `InstanceId`, and it must allow access to the SSM endpoints.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

We recommend that the EC2 instance receiving the command has an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. You must have at least **ssm:StartAutomationExecution** and **ssm:SendCommand** to run the automation and send the command to the instance, plus **ssm:GetAutomationExecution** to be able to read the automation output. For the offline remediation, see the permissions needed by `AWSSupport-StartEC2RescueWorkflow`.

### Document Steps

1. `aws:assertAwsResourceProperty` - Check the provided instance's platform is Windows.
2. `aws:assertAwsResourceProperty` - Confirm the provided instance is a managed instance:
  - a. (Online activation fix) If the input instance is a managed instance, then run `aws:runCommand` to run the PowerShell script to attempt to fix Windows activation.
  - b. (Offline activation fix) If the input instance is not a managed instance:
    - i. `aws:assertAwsResourceProperty` - Verifies the `AllowOffline` flag is set to `true`. If so, the offline fix starts; otherwise the automation ends.
    - ii. `aws:executeAutomation` - Invoke `AWSSupport-StartEC2RescueWorkflow` with the Windows activation offline fix script. The script uses either `EC2Config` or `EC2Launch`, depending on the OS version.
    - iii. `aws:executeAwsApi` - Read the result from `AWSSupport-StartEC2RescueWorkflow`.

### Outputs

`activateWindows.Output`

`getActivateWindowsOfflineResult.Output`

## AWSPremiumSupport- ChangeInstanceTypeIntelToAMD

### Description

The `AWSPremiumSupport-ChangeInstanceTypeIntelToAMD` runbook automates migrations from Intel powered Amazon Elastic Compute Cloud (Amazon EC2) instances to the equivalent AMD powered instance types. This runbook supports general purpose (M), burstable general purpose (T), compute optimized (C), and memory optimized (R) instances built on the Nitro system. This runbook can be used on instances that aren't managed by Systems Manager.

To reduce the potential risk of data loss and downtime, the runbook checks the instance's stop behavior, whether the instance is in an Amazon EC2 Auto Scaling group, the health of the instance, and that the equivalent AMD powered instance type is available in the same Availability Zone. By default, this runbook will not change the instance type if instance store volumes are attached, or if the instance is part of an AWS CloudFormation stack. If you want to change this behavior, specify `yes` for either of the `AllowInstanceStoreInstances` and `AllowCloudFormationInstances` parameters.

### Important

Access to `AWSPremiumSupport-*` runbooks requires either an Enterprise or Business Support Subscription. For more information, see [Compare AWS Support Plans](#).

### Considerations

- We recommend backing up your instance before using this runbook.
- Changing the instance type requires the runbook to stop your instance. When an instance is stopped, any data stored in the RAM or the instance store volumes is lost, and the automatic public IPv4 address is released. For more information, see [Stop and start your instance](#).
- If you don't specify a value for the `TargetInstanceType` parameter, the runbook attempts to identify the equivalent AMD instance in terms of virtual CPUs and memory within the same instance family. The runbook ends if it is not able to identify an equivalent AMD instance type.
- By using the `DryRun` option, you can capture the equivalent AMD instance type, and validate requirements without actually changing the instance type.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `Acknowledge`

Type: String

Description: (Required) Enter `yes` to acknowledge that your target instance will be stopped if it's running.

- `InstanceId`

Type: String

Description: (Required) The ID of Amazon EC2 instance whose type you want to change.

- `TargetInstanceType`

Type: String

Default: automatic

Description: (Optional) The AMD instance type you want to change your instance to. The default `automatic` value uses the equivalent instance type in terms of virtual CPUs and memory. For example, an `m5.large` would be changed to `m5a.large`.

- AllowInstanceStoreInstances

Type: String

Valid values: no | yes

Default: no

Description: (Optional) If you specify `yes`, the runbook runs on instances that have instance store volumes attached.

- AllowCloudFormationInstances

Type: String

Valid values: no | yes

Default: no

Description: (Optional) If set to `yes`, the runbook runs on instances that are part of a AWS CloudFormation stack.

- AllowCrossGeneration

Type: String

Valid values: no | yes

Default: no

Description: (Optional) If set to `yes`, the runbook attempts to find the newest equivalent AMD instance type within the same instance family.

- DryRun

Type: String

Valid values: no | yes

Default: no

Description: (Optional) If set to `yes`, the runbook returns the equivalent AMD instance type and validates migration requirements without making changes to the instance type.

- SleepWait

Type: String

Default: PT3S

Description: (Optional) The time the runbook should wait before starting a new automation. The value you provide for this parameter must match the ISO 8601 standard. For more information about creating ISO 8601 strings, see [Formatting date and time strings for Systems Manager](#).

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:DescribeAutomationExecutions`
- `ssm:GetAutomationExecution`
- `ssm:StartAutomationExecution`
- `ec2:GetInstanceTypesFromInstanceRequirements`
- `ec2:DescribeInstanceAttribute`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceStatus`
- `ec2:DescribeInstanceTypeOfferings`
- `ec2:DescribeInstanceTypes`
- `ec2:DescribeTags`
- `ec2:ModifyInstanceAttribute`
- `ec2:StartInstances`
- `ec2:StopInstances`

### Document Steps

1. `aws:assertAwsResourceProperty`: Confirms the status of the target Amazon EC2 instance is running, pending, stopped, or stopping. Otherwise, the automation ends.
2. `aws:executeAwsApi`: Gathers properties from the target Amazon EC2 instance.
3. `aws:branch`: Branches the automation based on the state of the Amazon EC2 instance.
  - a. If stopped or stopping, the automation runs `aws:waitForAwsResourceProperty` until the Amazon EC2 instance is fully stopped.
  - b. If running or pending, the automation runs `aws:waitForAwsResourceProperty` until the Amazon EC2 instance passes status checks.
4. `aws:assertAwsResourceProperty`: Confirms the Amazon EC2 instance is not part of an Auto Scaling group by checking if the `aws:autoscaling:groupName` tag is applied.
5. `aws:executeAwsApi`: Gathers the current instance type properties to find the equivalent AMD instance type.
6. `aws:assertAwsResourceProperty`: Confirms a AWS Marketplace product code is not associated with the Amazon EC2 instance. Some products are not available on all instance types.
7. `aws:branch`: Branches the automation depending on whether you want the automation to check if the Amazon EC2 instance is part of a AWS CloudFormation stack
  - a. If the `aws:cloudformation:stack-name` tag is applied to the instance, the automation runs `aws:assertAwsResourceProperty` to confirm the instance is not part of a AWS CloudFormation stack.
8. `aws:branch`: Branches the automation based on whether the instance root volume type is Amazon Elastic Block Store (Amazon EBS).
9. `aws:assertAwsResourceProperty`: Confirms the instance shutdown behavior is stop and not terminate.
10. `aws:executeScript`: Confirms there is only one automation of this runbook targeting the current instance. If another automation is already in progress targeting the same instance, it returns an error and ends.
11. `aws:executeAwsApi`: Returns a list of the AMD instance types with same amount of memory and vCPUs.
12. `aws:executeScript`: Checks if the current instance type is supported and returns its equivalent AMD instance type. If there is no equivalent, the automation ends.
13. `aws:executeScript`: Confirms the AMD instance type is available in the same Availability Zone, and verifies the provided IAM permissions.

- 14aws:branch: Branches the automation based on whether the DryRun parameter value is yes.
- 15aws:branch: Checks if the original and the target instance type are the same. If they're the same, the automation ends.
- 16aws:executeAwsApi: Gets the current instance state.
- 17aws:changeInstanceState: Stops the Amazon EC2 instance.
- 18aws:changeInstanceState: Forces the instance to stop if it's stuck in stopping state.
- 19aws:executeAwsApi: Changes the instance type to the target AMD instance type.
- 20aws:sleep: Waits 3 seconds after changing the instance type for eventual consistency.
- 21aws:branch: Branches the automation based on the previous instance state. If it was running, the instance is started.
- a. aws:changeInstanceState: Starts the Amazon EC2 instance if it was running before changing the instance type.
  - b. aws:waitForAwsResourceProperty: Waits for the Amazon EC2 instance to pass status checks. If the instance doesn't pass status checks, the instance is changed back to its original instance type.
    - i. aws:changeInstanceState: Stops the Amazon EC2 instance before changing it to its original instance type.
    - ii. aws:changeInstanceState: Forces the Amazon EC2 instance to stop before changing it to its original instance type in case it gets stuck in a stopping state.
    - iii. aws:executeAwsApi: Changes Amazon EC2 instance to its original type.
    - iv. aws:sleep: Waits 3 seconds after changing instance type for eventual consistency.
    - v. aws:changeInstanceState: Starts the Amazon EC2 instance if it was running before changing the instance type.
    - vi. aws:waitForAwsResourceProperty: Waits for the Amazon EC2 instance to pass status checks.
- 22aws:sleep: Waits before ending the runbook.

## AWSSupport- CheckXenToNitroMigrationRequirements

### Description

The AWSSupport-CheckXenToNitroMigrationRequirements runbook verifies that an Amazon Elastic Compute Cloud (Amazon EC2) instance meets the prerequisites to successfully change the instance type from a Xen type instance to Nitro-based instance type. This automation checks the following:

- The root device is an Amazon Elastic Block Store (Amazon EBS) volume.
- The enaSupport attribute is enabled.
- The ENA module is installed on the instance.
- The NVMe module is installed on the instance. If yes, the module is installed and a script verifies that the module is loaded in the initramfs image.
- Analyzes /etc/fstab and looks for block devices being mounted using device names.
- Determines whether the operating system (OS) uses predictable network interface names by default.

This runbook supports the following operating systems:

- Red Hat Enterprise Linux
- CentOS
- Amazon Linux 2

- Amazon Linux
- Debian Server
- Ubuntu Server
- SUSE Linux Enterprise Server 15 SP2
- SUSE Linux Enterprise Server 12 SP5

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Default: false

Description: (Required) The ID of the Amazon EC2 instance which you want to check prerequisites for before migrating to a Nitro-based instance type.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:DescribeAutomationExecutions
- ssm:DescribeAutomationStepExecutions
- ssm:DescribeAutomationStepExecutions
- ssm:DescribeInstanceInformation
- ssm:DescribeInstanceProperties
- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ssm:GetDocument
- ssm:ListCommands
- ssm:ListCommandInvocations
- ssm:ListDocuments

- `ssm:StartAutomationExecution`
- `ssm:SendCommand`
- `iam:ListRoles`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceTypes`

### Document Steps

- `aws:executeAwsApi` - Gathers details about the instance.
- `aws:executeAwsApi` - Gathers information about the hypervisor for the instance.
- `aws:branch` - Branches based on whether the target instance is already running a Nitro-based instance type.
- `aws:branch` - Checks whether the instance's OS is supported by Nitro-based instances.
- `aws:assertAwsResourceProperty` - Verifies the instance you specified is managed by Systems Manager, and that the status is `Online`.
- `aws:branch` - Branches based on whether the root device of the instance is an Amazon EBS volume.
- `aws:branch` - Branches based on whether the ENA attribute is enabled for the instance.
- `aws:runCommand` - Checks for ENA drivers on the instance.
- `aws:runCommand` - Checks for NVMe drivers on the instance.
- `aws:runCommand` - Checks the `fstab` file for unrecognized formats.
- `aws:runCommand` - Checks for predictable interface name configuration on the instance.
- `aws:executeScript` - Generates output based on previous steps.

### Outputs

`finalOutput.output` - The results of the checks performed by the automation.

## AWSSupport-ConfigureEC2Metadata

### Description

This runbook helps you configure instance metadata service (IMDS) options for Amazon Elastic Compute Cloud (Amazon EC2) instances. Using this runbook, you can configure the following:

- Enforce the use of IMDSv2 for instance metadata.
- Configure the `HttpPutResponseHopLimit` value.
- Allow or deny instance metadata access.

For more information about instance metadata, see [Configuring the Instance Metadata Service](#) in the *Amazon EC2 User Guide for Linux Instances*.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms



Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- EnforceIMDSv2

Type: String

Valid values: required | optional

Default: optional

Description: (Optional) Enforce IMDSv2. If you choose `required`, the Amazon EC2 instance will only use IMDSv2. If you choose `optional`, you can choose between IMDSv1 and IMDSv2 for metadata access.

#### Important

If you enforce IMDSv2, applications that use IMDSv1 might not function correctly. Before enforcing IMDSv2, make sure your applications that use IMDS are upgraded to a version that support IMDSv2. For information about Instance Metadata Service Version 2 (IMDSv2), see [Configuring the Instance Metadata Service](#) in the *Amazon EC2 User Guide for Linux Instances*.

- HttpPutResponseHopLimit

Type: Integer

Valid values: 0-64

Default: 0

Description: (Optional) The desired HTTP PUT response hop limit value (1-64) for instance metadata requests. This value controls the number of hops that the PUT response can traverse. To prevent the response from traveling outside of the instance, specify 1 for the parameter value.

- InstanceId

Type: String

Description: (Required) The ID of the Amazon EC2 instance whose metadata settings you want to configure.

- MetadataAccess

Type: String

Valid values: enabled | disabled

Default: enabled

Description: (Optional) Allow or deny instance metadata access in the Amazon EC2 instance. If you specify `disabled`, all other parameters will be ignored and the metadata access will be denied for the instance.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ec2:DescribeInstances`
- `ec2:ModifyInstanceMetadataOptions`
- `ssm:GetAutomationExecution`
- `ssm:StartAutomationExecution`

### Document Steps

1. `branchOnMetadataAccess` - Branches automation based on the value of `MetadataAccess` parameter.
2. `disableMetadataAccess` - Calls the `ModifyInstanceMetadataOptions` API action to disable metadata endpoint access.
3. `branchOnHttpPutResponseHopLimit` - Branches automation based on the value of `HttpPutResponseHopLimit` parameter.
4. `maintainHopLimitAndConfigureImdsVersion` - If `HttpPutResponseHopLimit` is 0, maintains current hop limit and changes other metadata options.
5. `waitBeforeAssertingIMDSv2State` - Waits 30 seconds before asserting IMDSv2 status.
6. `setHopLimitAndConfigureImdsVersion` - If `HttpPutResponseHopLimit` is greater than 0, configures the metadata options using the given input parameters.
7. `waitBeforeAssertingHopLimit` - Waits 30 seconds before asserting metadata options.
8. `assertHopLimit` - Asserts the `HttpPutResponseHopLimit` property is set to the value you specified.
9. `branchVerificationOnIMDSv2Option` - Branches verification based on the value of `EnforceIMDSv2` parameter.
10. `assertIMDSv2IsOptional` - Asserts `HttpTokens` value set to `optional`.
11. `assertIMDSv2IsEnforced` - Asserts `HttpTokens` value set to `required`.
12. `waitBeforeAssertingMetadataState` - Waits 30 seconds before asserting the metadata state is disabled.
13. `assertMetadataIsDisabled` - Asserts metadata is `disabled`.
14. `describeMetadataOptions` - Gets the metadata options after the changes you've specified have been applied.

### Outputs

`describeMetadataOptions.State`

`describeMetadataOptions.MetadataAccess`

`describeMetadataOptions.IMDSv2`

`describeMetadataOptions.HttpPutResponseHopLimit`

## AWSSupport-CopyEC2Instance

### Description

The `AWSSupport-CopyEC2Instance` runbook provides an automated solution for the procedure outlined in the Knowledge Center article [How do I move my EC2 instance to another subnet, Availability Zone, or VPC?](#) The automation branches depending on the values you specify for the `Region` and `SubnetId` parameters.

If you specify a value for the `SubnetId` parameter but not a value for the `Region` parameter, the automation creates an Amazon Machine Image (AMI) of the target instance and launches a new instance from the AMI in the subnet you specified.

If you specify a value for the `SubnetId` parameter and the `Region` parameter, the automation creates an AMI of the target instance, copies the AMI to the AWS Region you specified, and launches a new instance from the AMI in the subnet you specified.

If you specify a value for the `Region` parameter but not a value for the `SubnetId` parameter, the automation creates an AMI of the target instance, copies the AMI to the Region you specified, and launches a new instance from the AMI in the default subnet of your virtual private cloud (VPC) in the destination Region.

If no value is specified for either the `Region` or `SubnetId` parameters, the automation creates an AMI of the target instance, and launches a new instance from the AMI in the default subnet of your VPC.

To copy an AMI to a different Region, you must provide a value for the `AutomationAssumeRole` parameter. If the automation times out during the `waitForAvailableDestinationAmi` step, the AMI might still be copying. If this is the case, you can wait for the copy to complete and launch the instance manually.

Before running this automation, note the following:

- AMIs are based on Amazon Elastic Block Store (Amazon EBS) snapshots. For large file systems without a previous snapshot, AMI creation can take several hours. To decrease the AMI creation time, create an Amazon EBS snapshot before you create the AMI.
- Creating an AMI doesn't create a snapshot for instance store volumes on the instance. For information about backing up instance store volumes to Amazon EBS, see [How do I back up an instance store volume on my Amazon EC2 instance to Amazon EBS?](#)
- The new Amazon EC2 instance has a different private IPv4 or public IPv6 IP address. You must update all references to the old IP addresses (for example, in DNS entries) with the new IP addresses that are assigned to the new instance. If you're using an Elastic IP address on your source instance, be sure to attach it to the new instance.
- Domain security identifier (SID) conflict issues can occur when the copy launches and tries to contact the domain. Before you capture the AMI, use Sysprep or remove the domain-joined instance from the domain to prevent conflict issues. For more information, see [How can I use Sysprep to create and install custom reusable Windows AMIs?](#)

[Run this Automation \(console\)](#)

**Important**

We do not recommend using this runbook to copy Microsoft Active Directory Domain Controller instances.

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the instance that you want to copy.

- KeyPair

Type: String

Description: (Optional) The key pair you want to associate with the new copied instance. If you're copying the instance to a different Region, make sure the key pair exists in the specified Region.

- Region

Type: String

Description: (Optional) The Region you want to copy the instance to. If you specify a value for this parameter, but do not specify values for the `SubnetId` and `SecurityGroupIds` parameters, the automation attempts to launch the instance in the default VPC with the default security group. If EC2-Classic is enabled in the destination Region, the launch will fail.

- SubnetId

Type: String

Description: (Optional) The ID of the subnet you want to copy the instance to. If EC2-Classic is enabled in the destination Region, you must provide a value for this parameter.

- InstanceType

Type: String

Description: (Optional) The instance type the copied instance should be launched as. If you do not specify a value for this parameter, the source instance type is used. If the source instance type is not supported in the Region the instance is being copied to, the automation fails.

- SecurityGroupIds

Type: String

Description: (Optional) A comma-separated list of security group IDs you want to associate with the copied instance. If you do not specify a value for this parameter, and the instance is not being copied to a different Region, the security groups associated with the source instance are used. If you're copying the instance to a different Region, the default security group for the default VPC in the destination Region is used.

- KeepImageSourceRegion

Type: Boolean

Valid values: `true` | `false`

Default: `true`

Description: (Optional) If you specify `true` for this parameter, the automation does not delete the AMI of the source instance. If you specify `false` for this parameter, the automation deregisters the AMI and deletes the associated snapshots.

- KeepImageDestinationRegion

Type: Boolean

Valid values: true | false

Default: true

Description: (Optional) If you specify `true` for this parameter, the automation does not delete the AMI that is copied to the Region you specified. If you specify `false` for this parameter, the automation deregisters the AMI and deletes the associated snapshots.

- `NoRebootInstanceBeforeTakingImage`

Type: Boolean

Valid values: true | false

Default: false

Description: (Optional) If you specify `true` for this parameter, the source instance will not be restarted before creating the AMI. When this option is used, file system integrity on the created image can't be guaranteed.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ec2:CreateImage`
- `ec2>DeleteSnapshot`
- `ec2:DeregisterImage`
- `ec2:DescribeInstances`
- `ec2:DescribeImages`
- `ec2:RunInstances`

If you're copying the instance to a different Region, you will also need the following permissions.

- `ec2:CopyImage`

### Document Steps

- `describeOriginalInstanceDetails` - Gathers details from the instance to be copied.
- `assertRootVolumeIsEbs` - Checks if the root volume device type is `ebs`, and if not, ends the automation.
- `evalInputParameters` - Evaluates the values provided for the input parameters.
- `createLocalAmi` - Creates an AMI of the source instance.
- `tagLocalAmi` - Tags the AMI created in the previous step.
- `branchAssertRegionIsSame` - Branches based on whether the instance is being copied within the same Region or to a different Region.
- `branchAssertSameRegionWithKeyPair` - Branches based on whether a value was provided for the `KeyPair` parameter for an instance that's being copied within the same Region.
- `sameRegionLaunchInstanceWithKeyPair` - Launches an Amazon EC2 instance from the AMI of the source instance in the same subnet or the subnet you specify using the key pair that you specified.
- `sameRegionLaunchInstanceWithoutKeyPair` - Launches an Amazon EC2 instance from the AMI of the source instance in the same subnet or the subnet you specify without a key pair.

- `copyAmiToRegion` - Copies the AMI to the destination Region.
- `waitForAvailableDestinationAmi` - Waits for the copied AMI state to become `available`.
- `destinationRegionLaunchInstance` - Launches an Amazon EC2 Instance using the copied AMI.
- `branchAssertDestinationAmiToDelete` - Branches based on the value you provided for the `KeepImageDestinationRegion` parameter.
- `deregisterDestinationAmiAndDeleteSnapshots` - Deregisters the copied AMI and deletes associated snapshots.
- `branchAssertSourceAmiToDelete` - Branches based on the value you provided for the `KeepImageSourceRegion` parameter.
- `deregisterSourceAmiAndDeleteSnapshots` - Deregisters the AMI created from the source instance and deletes associated snapshots.
- `sleep` - Sleeps the automation for 2 seconds. This is a terminal step.

### Outputs

`sameRegionLaunchInstanceWithKeyPair.InstanceIds`

`sameRegionLaunchInstanceWithoutKeyPair.InstanceIds`

`destinationRegionLaunchInstance.DestinationInstanceIds`

## AWSsupport-ExecuteEC2Rescue

### Description

This runbook will use the `EC2Rescue` tool to troubleshoot and where possible repair common connectivity issues with the specified EC2 instance for Linux or Windows Server.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `EC2RescueInstanceType`

Type: String

Valid values: `t2.small` | `t2.medium` | `t2.large`

Default: t2.small

Description: (Required) The EC2 instance type for the EC2Rescue instance. Recommended size: t2.small.

- LogDestination

Type: String

Description: (Optional) S3 bucket name in your account where you want to upload the troubleshooting logs. Make sure the bucket policy does not grant unnecessary read/write permissions to parties that do not need access to the collected logs.

- SubnetId

Type: String

Default: CreateNewVPC

Description: (Optional) The subnet ID for the EC2Rescue instance. By default, AWS Systems Manager Automation creates a new VPC. Alternatively, Use SelectedInstanceSubnet to use the same subnet as your instance, or specify a custom subnet ID. IMPORTANT: The subnet must be in the same Availability Zone as UnreachableInstanceid, and it must allow access to the SSM endpoints.

- UnreachableInstanceid

Type: String

Description: (Required) ID of your unreachable EC2 instance. IMPORTANT: AWS Systems Manager Automation stops this instance, and creates an AMI before attempting any operations. Data stored in instance store volumes will be lost. The public IP address will change if you are not using an Elastic IP.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

You must have at least **ssm:StartAutomationExecution** and **ssm:GetAutomationExecution** to be able to read the automation output. For more information about the required permissions see **AWSSupport-StartEC2RescueWorkflow**

### Document Steps

1. **aws:assertAwsResourceProperty** - Assert if the provided instance is Windows
  - a. (EC2Rescue for Windows) If the provided instance is a Windows Server instance:
    - i. **aws:executeAutomation** - Invoke **AWSSupport-StartEC2RescueWorkflow** with the EC2Rescue for Windows offline script
    - ii. **aws:executeAwsApi** - Retrieve the backup AMI ID from the nested automation
    - iii. **aws:executeAwsApi** - Retrieve the EC2Rescue summary from the nested automation
  - b. (EC2Rescue for Linux) If the provided instance is a Linux instance:
    - i. **aws:executeAutomation** - Invoke **AWSSupport-StartEC2RescueWorkflow** with the EC2Rescue for Linux offline script
    - ii. **aws:executeAwsApi** - Retrieve the backup AMI ID from the nested automation
    - iii. **aws:executeAwsApi** - Retrieve the EC2Rescue summary from the nested automation

### Outputs

getEC2RescueForWindowsResult.Output

getWindowsBackupAmi.ImageId

getEC2RescueForLinuxResult.Output

getLinuxBackupAmi.ImageId

## AWSSupport-ListEC2Resources

### Description

The `AWSSupport-ListEC2Resources` runbook returns information about Amazon EC2 instances and related resources like Amazon Elastic Block Store (Amazon EBS) volumes, Elastic IP addresses, and Amazon EC2 Auto Scaling groups from the AWS Regions you specify. By default, the information is gathered from all Regions and is displayed in the output of the automation. Optionally, you can specify an Amazon Simple Storage Service (Amazon S3) bucket for the information to be uploaded to as a comma-separated values (.csv) file.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Bucket

Type: String

Description: (Optional) The name of the S3 bucket where the information gathered is uploaded to.

- DisplayResourceDeletionDocumentation

Type: String

Default: true

Description: (Optional) If set to `true`, the automation creates links in the output to documentation related to deleting your resources.

- RegionsToQuery

Type: String

Default: All



Description: (Optional) The Regions you want to gather Amazon EC2 related information from.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `autoscaling:DescribeAutoScalingGroups`
- `ec2:DescribeAddresses`
- `ec2:DescribeImages`
- `ec2:DescribeInstances`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRegions`
- `ec2:DescribeVolumes`
- `ec2:DescribeSnapshots`
- `elasticloadbalancing:DescribeLoadBalancers`

Additionally, to successfully upload the information gathered to the S3 bucket you specify, the `AutomationAssumeRole` requires the following actions:

- `s3:GetBucketAcl`
- `s3:GetBucketPolicyStatus`
- `s3:PutObject`

### Document Steps

- `aws:executeAwsApi` - Gathers the Regions enabled for the account.
- `aws:executeScript` - Confirms the Regions enabled for the account support the Regions specified in the `RegionsToQuery` parameter.
- `aws:branch` - If no Regions are enabled for the account, the automation ends.
- `aws:executeScript` - Lists all EC2 instances for the account and Regions you specify.
- `aws:executeScript` - Lists all Amazon Machine Images (AMI) for the account and Regions you specify.
- `aws:executeScript` - Lists all EBS volumes for the account and Regions you specify.
- `aws:executeScript` - Lists all Elastic IP addresses for the account and Regions you specify.
- `aws:executeScript` - Lists all elastic network interfaces for the account and Regions you specify.
- `aws:executeScript` - Lists all Auto Scaling groups for the account and Regions you specify.
- `aws:executeScript` - Lists all load balancers for the account and Regions you specify.
- `aws:executeScript` - Uploads the information gathered to the S3 bucket specified if you provide a value for the `Bucket` parameter.

## AWSSupport-ManageRDPSettings

### Description

The `AWSSupport-ManageRDPSettings` runbook allows the user to manage common Remote Desktop Protocol (RDP) settings, such as the RDP port and Network Layer Authentication (NLA). By default, the runbook reads and outputs the values of the settings.

### **Important**

Changes to the RDP settings should be carefully reviewed before running this runbook.

[Run this Automation \(console\)](#)

### **Document type**

Automation

### **Owner**

Amazon

### **Platforms**

Windows

### **Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the managed instance to manage the RDP settings of.

- NLASettingAction

Type: String

Valid values: Check | Enable | Disable

Default: Check

Description: (Required) An action to perform on the NLA setting: Check, Enable, Disable.

- RDPPort

Type: String

Default: 3389

Description: (Optional) Specify the new RDP port. Used only when the action is set to Modify. The port number must be between 1025-65535. Note: After the port is changed, the RDP service is restarted.

- RDPPortAction

Type: String

Valid values: Check | Modify

Default: Check

Description: (Required) An action to apply to the RDP port.

- RemoteConnections

Type: String

Valid values: Check | Enable | Disable

Default: Check

Description: (Required) An action to perform on the fDenyTSConnections setting.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

The EC2 instance receiving the command must have an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. The user must have at least **ssm:SendCommand** to send the command to the instance, plus **ssm:GetCommandInvocation** to be able to read the command output.

### Document Steps

`aws:runCommand` - Run the PowerShell script to change or check the RDP settings on the target instance.

### Outputs

`manageRDPSettings.Output`

## AWSSupport-ManageWindowsService

### Description

The `AWSSupport-ManageWindowsService` runbook enables you to stop, start, restart, pause, or disable any Windows service on the target instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the managed instance to manage the services of.

- ServiceAction

Type: String

Valid values: Check | Restart | Force-Restart | Start | Stop | Force-Stop | Pause

Default: Check

Description: (Required) An action to apply to the Windows service. Note that `Force-Restart` and `Force-Stop` can be used to restart and to stop a service that has dependent services.

- StartupType

Type: String

Valid values: Check | Auto | Demand | Disabled | DelayedAutoStart

Default: Check

Description: (Required) A startup type to apply to the Windows service.

- WindowsServiceName

Type: String

Description: (Required) A valid Windows service name.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

It is recommended that the EC2 instance receiving the command has an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. The user must have at least **ssm:StartAutomationExecution** and **ssm:SendCommand** to run the automation and send the command to the instance, plus **ssm:GetAutomationExecution** to be able to read the automation output.

### Document Steps

`aws:runCommand` - Run the PowerShell script to apply the desired configuration to the Windows service on the target instance.

### Outputs

`manageWindowsService.Output`

## AWSSupport-MigrateEC2ClassicToVPC

### Description

The `AWSSupport-MigrateEC2ClassicToVPC` runbook migrates an Amazon Elastic Compute Cloud (Amazon EC2) instance from EC2-Classic to a virtual private cloud (VPC). This runbook supports migrating Amazon EC2 instances of the hardware virtual machine (HVM) virtualization type with Amazon Elastic Block Store (Amazon EBS) root volumes.

[Run this Automation \(console\)](#)

### Document type

Automation

**Owner**

Amazon

**Platforms**

Linux

**Parameters**

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- ApproverIAM

Type: StringList

Description: (Optional) The Amazon Resource Names (ARNs) of IAM users who can approve or deny the action. This parameter only applies if you specify the `CutOver` value for the `MigrationType` parameter.

- DestinationSecurityGroupId

Type: StringList

Description: (Optional) The ID of the security group you want to associate with the Amazon EC2 instance that is launched in your VPC. If you do not specify a value for this parameter, the automation creates a security group in your VPC and copies the rules from the security group in EC2-Classic. If the rules fail to copy to the new security group, the default security group of your VPC is associated with the Amazon EC2 instance.

- DestinationSubnetId

Type: String

Description: (Optional) The ID of the subnet you want to migrate your Amazon EC2 instance to. If you do not specify a value for this parameter, the automation randomly chooses a subnet from your VPC.

- InstanceId

Type: String

Description: (Required) The ID of the Amazon EC2 instance you want to migrate.

- MigrationType

Type: String

Valid values: `CutOver | Test`

Description: (Required) The type of migration you want to perform.

The `CutOver` option requires approval to stop your Amazon EC2 instance that's running in EC2-Classic. After this action is approved, the Amazon EC2 instance is stopped and the automation creates an Amazon Machine Image (AMI). When the AMI status is `available`, a new Amazon EC2 instance is

launched from this AMI in the `DestinationSubnetId` you specify in your VPC. If your Amazon EC2 instance that's running in EC2-Classic has an Elastic IP address attached, the instance will be moved to the newly created Amazon EC2 instance in your VPC. If the Amazon EC2 instance launching in your VPC fails to create for any reason, it is terminated and approval is requested to start your Amazon EC2 instance in EC2-Classic.

The `Test` option creates an AMI of your Amazon EC2 instance that's running in EC2-Classic without rebooting. Because the Amazon EC2 instance does not reboot, we can't guarantee the file system integrity of the created image. When the AMI status is `available`, a new Amazon EC2 instance is launched from this AMI in the `DestinationSubnetId` you specify in your VPC. If your Amazon EC2 instance that's running in EC2-Classic has an Elastic IP address attached, the automation verifies the `DestinationSubnetId` you specify is public. If the Amazon EC2 instance launching in your VPC fails to create for any reason, it is terminated and the automation ends.

- `SNSNotificationARNforApproval`

Type: String

Description: (Optional) The ARN of the Amazon Simple Notification Service (Amazon SNS) topic you want to send approval requests to. This parameter only applies if you specify the `CutOver` value for the `MigrationType` parameter.

- `TargetInstanceType`

Type: String

Default: `t2.2xlarge`

Description: (Optional) The type of Amazon EC2 instance you want to launch in your VPC. Only Xen-based instance types, such as T2, M4 or C4, are supported.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:GetDocument`
- `ssm:ListDocumentVersions`
- `ssm:ListDocuments`
- `ssm:StartAutomationExecution`
- `sns:GetTopicAttributes`
- `sns:ListSubscriptions`
- `sns:ListTopics`
- `sns:Publish`
- `ec2:AssociateAddress`
- `ec2:AuthorizeSecurityGroupIngress`
- `ec2:CreateImage`
- `ec2:CreateSecurityGroup`
- `ec2>DeleteSecurityGroup`
- `ec2:MoveAddressToVpc`
- `ec2:RunInstances`
- `ec2:StopInstances`
- `ec2:CreateTags`
- `ec2:DescribeAddresses`
- `ec2:DescribeInstanceAttribute`

- `ec2:DescribeInstances`
- `ec2:DescribeInstanceStatus`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroupReferences`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeTags`
- `ec2:DescribeVpcs`
- `ec2:DescribeInstanceTypes`
- `ec2:DescribeImages`

### Document Steps

- `aws:executeAwsApi` - Gathers details about the Amazon EC2 instance you specify in the `InstanceId` parameter.
- `aws:assertAwsResourceProperty` - Confirms the instance type you specify in the `TargetInstanceType` parameter is Xen-based.
- `aws:assertAwsResourceProperty` - Confirms the Amazon EC2 instance you specify in the `InstanceId` parameter is of the HVM virtualization type.
- `aws:assertAwsResourceProperty` - Confirms the Amazon EC2 instance you specify in the `InstanceId` parameter has an Amazon EBS root volume.
- `aws:executeScript` - Creates a security group as needed depending on the value you specify for the `DestinationSecurityGroupId` parameter.
- `aws:branch` - Branches based on the value you specify in the `DestinationSubnetId` parameter.
- `aws:executeAwsApi` - Identifies the default VPC in the AWS Region where you run this automation.
- `aws:executeAwsApi` - Randomly chooses the ID of a subnet located in the default VPC.
- `aws:createImage` - Creates an AMI without rebooting the Amazon EC2 instance.
- `aws:branch` - Branches based on the value you specify for the `MigrationType` parameter.
- `aws:branch` - Branches based on the value you specify for the `DestinationSubnetId` parameter.
- `aws:runInstances` - Launches a new instance from the AMI created without rebooting the Amazon EC2 instance in EC2-Classic.
- `aws:changeInstanceState` - Terminates the newly launched Amazon EC2 instance if the previous step fails for any reason.
- `aws:runInstances` - Launches a new instance from the AMI created without rebooting the Amazon EC2 instance in EC2-Classic in the `DestinationSubnetId` if provided.
- `aws:changeInstanceState` - Terminates the newly launched Amazon EC2 instance if the previous step fails for any reason.
- `aws:assertAwsResourceProperty` - Confirms the stop behavior for the Amazon EC2 instance running in EC2-Classic.
- `aws:approve` - Waits for approval to stop the Amazon EC2 instance.
- `aws:changeInstanceState` - Stops the Amazon EC2 instance running in EC2-Classic.
- `aws:changeInstanceState` - Force stops the Amazon EC2 instance running in EC2-Classic if needed.
- `aws:createImage` - Creates an AMI of the Amazon EC2 instance after it has stopped.
- `aws:branch` - Branches based on the value specified for the `DestinationSubnetId` parameter.
- `aws:runInstances` - Launches a new instance from the AMI created of the stopped Amazon EC2 instance in EC2-Classic.
- `aws:approve` - Waits for approval to terminate the newly launched instance and start the Amazon EC2 instance in EC2-Classic if the previous step fails for any reason.

- `aws:changeInstanceState` - Terminates the newly launched Amazon EC2 instance.
- `aws:runInstances` - Launches a new instance from the AMI created of the stopped Amazon EC2 instance in EC2-Classic in the `DestinationSubnetId`.
- `aws:approve` - Waits for approval to terminate the newly launched instance and start the Amazon EC2 instance in EC2-Classic if the previous step fails for any reason.
- `aws:changeInstanceState` - Terminates the newly launched Amazon EC2 instance.
- `aws:changeInstanceState` - Starts the Amazon EC2 instance that was stopped in EC2-Classic.
- `aws:branch` - Branches based on whether the Amazon EC2 instance has a public IP address.
- `aws:executeAwsApi` - Verifies whether the public IP address is an EIP.
- `aws:branch` - Branches based on the value you specify in the `MigrationType` parameter.
- `aws:executeAwsApi` - Moves the EIP to your VPC.
- `aws:executeAwsApi` - Gathers the allocation ID of the EIP that was moved to your VPC.
- `aws:branch` - Branches based on which subnet the Amazon EC2 instance running in your VPC was launched.
- `aws:executeAwsApi` - Attaches the EIP to the newly launched instance in your VPC.
- `aws:executeScript` - Confirms the subnet your newly launched Amazon EC2 instance running in your VPC is public.

## Outputs

`getInstanceProperties.virtualizationType` - The virtualization type of the Amazon EC2 instance running in EC2-Classic.

`getInstanceProperties.rootDeviceType` - The root device type of the Amazon EC2 instance running in EC2-Classic.

`createAMIWithoutReboot.ImageId` - The ID of the AMI created without rebooting the Amazon EC2 instance running in EC2-Classic.

`getDefaultVPC.VpcId` - The ID of the default VPC where the new Amazon EC2 instance will be launched if a value for the `DestinationSubnetId` parameter is not provided.

`getSubnetIdInDefaultVPC.subnetIdFromDefaultVpc` - The ID of the subnet in the default VPC where the new Amazon EC2 instance will be launched if a value for the `DestinationSubnetId` parameter is not provided.

`launchTestInstanceDefaultVPC.InstanceIds` - The ID of the newly launched Amazon EC2 instance in your default VPC during the `Test` migration type.

`launchTestInstanceProvidedSubnet.InstanceIds` - The ID of the newly launched Amazon EC2 instance in the `DestinationSubnetId` you specified during the `Test` migration type.

`createAMIAfterStoppingInstance.ImageId` - The ID of the AMI created after stopping the Amazon EC2 instance running in EC2-Classic.

`launchCutOverInstanceProvidedSubnet.InstanceIds` - The ID of the newly launched Amazon EC2 instance in the `DestinationSubnetId` you specified during the `CutOver` migration type.

`launchCutOverInstanceDefaultVPC.InstanceIds` - The ID of the newly launched Amazon EC2 instance in your default VPC during the `CutOver` migration type.

`verifySubnetIsPublicTestDefaultVPC.IsSubnetPublic` - Whether the subnet chosen by the automation in your default VPC is public or not.

`verifySubnetIsPublicTestProvidedSubnet.IsSubnetPublic` - Whether the subnet you specified in the `DestinationSubnetId` is public or not.



## AWSSupport-MigrateXenToNitroLinux

### Description

The `AWSSupport-MigrateXenToNitroLinux` runbook clones, prepares, and migrates an Amazon Elastic Compute Cloud (Amazon EC2) Linux Xen instance to a [Nitro instance type](#). This runbook provides two options for operation types:

- **Clone and Migrate** - This option's workflow consists of the **Preliminary Checks**, **Testing**, and **Clone&Migrate** phases. The workflow is run using `AWSSupport-CloneXenEC2InstanceAndMigrateToNitro` runbook.
- **Full Migration** - This option runs the `Clone&Migrate` workflow and then performs the additional step of **Replace root Amazon EBS volumes**.

### Preliminary checks

The automation performs the following preliminary checks before continuing with the migration. If any of the checks fail, the automation ends. This phase is only part of the `Clone&Migrate` workflow.

- Checks if the target instance is already a Nitro instance type.
- Checks if the Spot Instances purchasing option was used for the target instance.
- Checks if instance store volumes are attached to the target instance.
- Verifies the target instance operating system (OS) is Linux.
- Checks if the target instance is a part of an Amazon EC2 Auto Scaling group. If it is part of an Auto Scaling group, the automation verifies that the instance is in the `standby` state.
- Verifies that the instance is managed by AWS Systems Manager.

### Testing

The automation creates an Amazon Machine Image (AMI) from the target instance and launches a test instance from the newly created AMI. This phase is only part of the `Clone&Migrate` workflow.

If the test instance passes all status checks, the automation pauses and approval from the designated principals is requested via Amazon Simple Notification Service (Amazon SNS) notification. If approval is provided, the automation terminates the test instance, stops the target instance, and continues with the migration, while the newly created AMI is deregistered at the end of the `Clone&Migrate` workflow.

#### Note

Before providing approval, we recommend verifying that all applications running on the target instance have been gracefully closed.

### Clone and Migrate

The automation creates another AMI from the target instance, and launches a new instance to change to a Nitro instance type. The automation completes the following prerequisites before continuing with the migration. If any of the checks fail, the automation ends. This phase is also only part of the `Clone&Migrate` workflow.

- Turns on the enhanced networking (ENA) attribute.
- Installs the latest version of ENA drivers if they're not already installed, or updates ENA drivers version to the latest version. Updating to the latest ENA driver version is required if the Nitro instance type is the 6th generation to ensure maximum network performance.
- Verifies that the NVMe module is installed. If the module is installed, the automation verifies that the module is loaded in the `initramfs`.

- Analyzes `/etc/fstab` and replaces entries with block device names (`/dev/sd*` or `/dev/xvd*`) with their respective UUIDs. Before modifying the configuration, the automation creates a backup of the file at the path `/etc/fstab*`.
- Turns off predictable interface naming by adding the `net.ifnames=0` option to the `GRUB_CMDLINE_LINUX` line in the `/etc/default/grub` file if it exists, or to the kernel in `/boot/grub/menu.lst`.
- Removes the `/etc/udev/rules.d/70-persistent-net.rules` file if it exists. Before removing the file, the automation creates a backup of the file at the path `/etc/udev/rules.d/`.

After verifying all requirements, the instance type is changed to the Nitro instance type that you specify. The automation waits for the newly created instance to pass all status checks after starting as a Nitro instance type. The automation then waits for approval from the designated principals to create an AMI of the successfully launched Nitro instance. If approval is denied, the automation ends leaving the newly created instance running and the target instance remains stopped.

### Replace root Amazon EBS Volume

If you choose `FullMigration` as the `OperationType`, the automation migrates the target Amazon EC2 instance to the Nitro instance type you specify. Automation requests approval from designated principals to replace the root Amazon EBS volume of the target Amazon EC2 instance with the cloned Amazon EC2 instance's. After the migration is successful, the cloned Amazon EC2 instance is terminated. If the automation fails, the original Amazon EBS root volume is attached to the target Amazon EC2 instance. If the root Amazon EBS volume attached to the target Amazon EC2 instance has tags with the `aws:` prefix applied, the `FullMigration` operation are not supported.

### Before you begin

The target instance must have outbound internet access. This is to access repositories for drivers and dependencies like `kernel-devel`, `gcc`, `patch`, `rpm-build`, `wget`, `dracut`, `make`, `linux-headers`, and `unzip`. Package manager is used if needed.

An Amazon SNS topic is required to send notifications for approvals and updates. For more information about creating an Amazon SNS topic, see [Creating an Amazon SNS topic](#) in the *Amazon Simple Notification Service Developer Guide*.

This runbook supports the following operating systems:

- RHEL 7.x - 8.5
- Amazon Linux (2018.03), Amazon Linux 2
- Debian Server
- Ubuntu Server 18.04 LTS, 20.04 LTS, and 20.10 STR
- SUSE Linux Enterprise Server (SUSE12SP5, SUSE15SP2)

### Important

Using this runbook incurs costs to your account for the running time of Amazon EC2 instances, creation of Amazon Elastic Block Store (Amazon EBS) volumes, and AMIs. For more details, see [Amazon EC2 Pricing](#) and [Amazon EBS Pricing](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Linux

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Acknowledgement

Type: String

Description: (Required) Read the complete details of the actions performed by this automation runbook, and enter **Yes, I understand and acknowledge** to proceed with using the runbook.

- ApproverIAM

Type: String

Description: (Required) The ARNs of IAM roles, users, or IAM user names who can provide approvals to the automation. You can specify a maximum of 10 approvers.

- DeleteResourcesOnFailure

Type: Boolean

Description: (Optional) Determines whether the newly created instance and AMI for the migration are deleted if the automation fails.

Valid values: True | False

Default: True

- MinimumRequiredApprovals

Type: String

Description: (Optional) The minimum number of approvals required to continue running the automation when approvals are requested.

Valid values: 1-10

Default: 1

- NitroInstanceType

Type: String

Description: (Required) The Nitro instance type that you want to change the instance to. Supported instance types include M5, M6, C5, C6, R5, R6, and T3.

Default: m5.xlarge

- OperationType

Type: String

Description: (Required) The operation you want to perform. The `FullMigration` option performs the same tasks as `Clone&Migrate`, but additionally replaces the root volume of your target instance. The root volume of the target instance is replaced with the root volume from the newly created instance following the migration process. The `FullMigration` operation does not support root volumes defined by Logical Volume Manager (LVM).

Valid values: `Clone&Migrate` | `FullMigration`

- `SNSTopicArn`

Type: String

Description: (Required) The ARN of the Amazon SNS topic for approval notification. The Amazon SNS topic is used to send required approval notifications during the automation.

- `TargetInstanceid`

Type: String

Description: (Required) The ID of the Amazon EC2 instances to migrate.

## Clone&Migrate workflow

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:DescribeAutomationExecutions`
- `ssm:StartAutomationExecution`
- `ssm:DescribeInstanceInformation`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:SendCommand`
- `ssm:GetAutomationExecution`
- `ssm:ListCommands`
- `ssm:ListCommandInvocations`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceTypeOfferings`
- `ec2:DescribeInstanceTypes`
- `ec2:DescribeImages`
- `ec2:CreateImage`
- `ec2:RunInstances`
- `ec2:DescribeInstanceStatus`
- `ec2:DeregisterImage`
- `ec2>DeleteSnapshot`
- `ec2:TerminateInstances`
- `ec2:StartInstances`
- `ec2:DescribeKeyPairs`
- `ec2:StopInstances`
- `kms:CreateGrant*`
- `kms:ReEncrypt`
- `ec2:ModifyInstanceAttribute`
- `autoscaling:DescribeAutoScalingInstances`

- `iam:passRole`
- `iam:ListRoles`

### Document Steps

- `startOfPreliminaryChecksBranch` - Branches to the Preliminary checks workflow.
- `getTargetInstanceProperties` - Gathers details from the target instance.
- `checkIfNitroInstanceTypeIsSupportedInAZ` - Determines if the target Amazon EC2 instance type is supported in the same Availability Zone as the target instance.
- `getXenInstanceTypeDetails` - Gathers details about the source instance type.
- `checkIfInstanceHypervisorIsNitroAlready` - Checks if the target instance is already running as a Nitro instance type.
- `checkIfTargetInstanceLifecycleIsSpot` - Checks if the purchasing option of the target instance is Spot.
- `checkIfOperatingSystemIsLinux` - Checks if the target instance OS is Linux.
- `verifySSMConnectivityForTargetInstance` - Verifies that the target instance is managed by Systems Manager.
- `checkIfEphemeralVolumeAreSupported` - Checks if the current instance type of the target instance supports instance store volumes.
- `verifyIfTargetInstanceHasEphemeralVolumesAttached` - Checks if the target instance has instance store volumes attached.
- `checkIfRootVolumeIsEBS` - Checks if the target instance's root volume type is EBS.
- `checkIfTargetInstanceIsInASG` - Checks if the target instance is a part of an Auto Scaling group.
- `endOfPreliminaryChecksBranch` - End of the Preliminary checks branch.
- `startOfTestBranch` - Branches to the Testing workflow.
- `createTestImage` - Creates a test AMI of the target instance.
- `launchTestInstanceInSameSubnet` - Launches a test instance from the test AMI using the same configuration as target instance.
- `cleanupTestInstance` - Terminates the test instance.
- `endOfTestBranch` - End of the Testing branch.
- `checkIfTestingBranchSucceeded` - Checks the status of the Testing branch.
- `approvalToStopTargetInstance` - Waits for approval from the designated principals to stop the target instance.
- `stopTargetEC2Instance` - Stops the target instance.
- `forceStopTargetEC2Instance` - Force stops the target instance only if the previous step fails to stop the instance.
- `startOfCloneAndMigrateBranch` - Branches to the Clone and Migrate workflow.
- `createBackupImage` - Creates an AMI of the target instance to serve as a backup.
- `launchInstanceInSameSubnet` - Launches a new instance from the backup AMI using the same configuration as the source instance.
- `waitForClonedInstanceToPassStatusChecks` - Waits for the newly created instance to pass all status checks.
- `verifySSMConnectivityForClonedInstance` - Verifies that the newly created instance is managed by Systems Manager.
- `checkAndInstallENADrivers` - Checks if ENA drivers are installed on the newly created instance, and installs the drivers if needed.
- `checkAndAddNVMEDrivers` - Checks if NVMe drivers are installed on the newly created instance, and installs the drivers if needed.

- `checkAndModifyFSTABEntries` - Checks if device names are used in `/etc/fstab` and replaces them with UUIDs if needed.
- `stopClonedInstance` - Stops the newly created instance.
- `forceStopClonedInstance` - Force stops the newly created instance only if the previous step fails to stop the instance.
- `checkENAAtributeForClonedInstance` - Checks if the enhanced networking attribute is turned on for the newly created instance.
- `setNitroInstanceTypeForClonedInstance` - Changes the instance type for the newly created instance to the Nitro instance type that you specify.
- `startClonedInstance` - Starts the newly created instance whose instance type you changed.
- `approvalForCreatingImageAfterDriversInstallation` - If the instance successfully starts as a Nitro instance type, the automation waits for approval from the required principals. If approval is provided, an AMI is created to be used as a Golden AMI.
- `createImageAfterDriversInstallation` - Creates an AMI to be used as a Golden AMI.
- `endOfCloneAndMigrateBranch` - End of Clone and Migrate branch.
- `cleanupTestImage` - Deregisters the AMI created for testing.
- `failureHandling` - Checks if you chose to terminate resources on failure.
- `onFailureTerminateClonedInstance` - Terminates the newly created instance if the automation fails.
- `onFailurecleanupTestImage` - Deregisters the AMI created for testing.
- `onFailureApprovalToStartTargetInstance` - If the automation fails, waits for approval from the designated principals to start the target instance.
- `onFailureStartTargetInstance` - If the automation fails, starts the target instance.

## FullMigration workflow

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:DescribeAutomationExecutions`
- `ssm:DescribeInstanceInformation`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:SendCommand`
- `ssm:GetAutomationExecution`
- `ssm:ListCommands`
- `ssm:ListCommandInvocations`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceTypeOfferings`
- `ec2:DescribeInstanceTypes`
- `ec2:DescribeImages`
- `ec2:CreateImage`
- `ec2:RunInstances`
- `ec2:DescribeInstanceStatus`
- `ec2:DeregisterImage`
- `ec2>DeleteSnapshot`
- `ec2:TerminateInstances`
- `ec2:StartInstances`
- `ec2:DescribeKeyPairs`

- `ec2:StopInstances`
- `kms:CreateGrant*`
- `kms:ReEncrypt`
- `ec2:ModifyInstanceAttribute`
- `ec2:DetachVolume`
- `ec2:AttachVolume`
- `ec2:DescribeVolumes`
- `autoscaling:DescribeAutoScalingInstances`
- `iam:PassRole`
- `ec2:CreateTags`
- `cloudformation:DescribeStackResources`

### Document Steps

The `FullMigration` workflow runs the same steps as the `Clone&Migrate` workflow and additionally performs the following steps:

- `checkConcurrency` - Verifies that there is only one automation of this runbook targeting the Amazon EC2 instance you specify. If the runbook finds another automation in progress targeting the same instance, the automation ends.
- `getTargetInstanceProperties` - Gathers details from the target instance.
- `checkRootVolumeTags` - Determines if the root volume of the target Amazon EC2 instance contains any AWS reserved tags.
- `cloneTargetInstanceAndMigrateToNitro` - Starts a child automation using the `AWS-CloneXenInstanceToNitro` runbook.
- `branchOnTheOperationType` - Branches on the value you specify for the `OperationType` parameter.
- `getClonedInstanceId` - Retrieves the ID of the newly launched instance from the child automation.
- `checkIfRootVolumeIsBasedOnLVM` - Determines if the root partition is managed by LVM.
- `branchOnTheRootVolumeLVMStatus` - If the minimum required approvals are received from the principals, the automation proceeds with the root volume replacement.
- `manualInstructionsInCaseOfLVM` - If the root volume is managed by LVM, the automation sends output containing instructions for how to manually replace the root volumes.
- `startOfReplaceRootEBSVolumeBranch` - Starts the Replace Root EBS Volume branch workflow.
- `checkIfTargetInstanceIsManagedByCFN` - Determines if the target instance is managed by an AWS CloudFormation stack.
- `branchOnCFNStackStatus` - Branches based on the status of the CloudFormation stack.
- `approvalForRootVolumesReplacement(WithCFN)` - If the target instance was launched by CloudFormation, the automation waits for approval after the newly launched instance successfully starts as a Nitro instance type. When approvals are provided, the Amazon EBS volumes of the target instance are replaced with the root volumes from the newly launched instance.
- `approvalForRootVolumesReplacement` - Waits for approval after the newly launched instance successfully starts as a Nitro instance type. When approvals are provided, the Amazon EBS volumes of the target instance are replaced with the root volumes from the newly launched instance.
- `assertIfTargetEC2InstanceIsStillStopped` - Verifies that the target instance is in a stopped state before replacing the root volume.
- `stopTargetInstanceForRootVolumeReplacement` - If the target instance is running, the automation stops the instance before replacing the root volume.
- `forceStopTargetInstanceForRootVolumeReplacement` - Force stops the target instance if the previous step fails.

- `stopClonedInstanceForRootVolumeReplacement` - Stops the newly created instance before replacing the Amazon EBS volumes.
- `forceStopClonedInstanceForRootVolumeReplacement` - Force stops the newly created instance if the previous step fails.
- `getBlockDeviceMappings` - Retrieves the block device mappings for both the target and newly created instances.
- `replaceRootEbsVolumes` - Replaces the root volume of the target instance with the root volume of the newly created instance.
- `EndOfReplaceRootEBSVolumeBranch` - End of Replace Root EBS Volume branch workflow.
- `checkENAAttributeForTargetInstance` - Checks if the enhanced networking (ENA) attribute is turned on for the target Amazon EC2 instance.
- `enableENAAttributeForTargetInstance` - Turns on the enhanced networking (ENA) attribute for the target Amazon EC2 instance if needed.
- `setNitroInstanceTypeForTargetInstance` - Changes the target instance to the Nitro instance type you specify.
- `replicateRootVolumeTags` - Replicates the tags on the root Amazon EBS volume from the target Amazon EC2 instance.
- `startTargetInstance` - Starts target Amazon EC2 instance after changing the instance type.
- `onFailureStopTargetEC2Instance` - Stops target Amazon EC2 instance if it fails to start as a Nitro instance type.
- `onFailureForceStopTargetEC2Instance` - Force stops the target Amazon EC2 instance if the previous step fails.
- `OnFailureRevertOriginalInstanceType` - Reverts the target Amazon EC2 instance to the original instance type if the target instance fails to start as a Nitro instance type.
- `onFailureRollbackRootVolumeReplacement` - Reverts all the changes made by the `replaceRootEbsVolumes` step if needed.
- `onFailureApprovalToStartTargetInstance` - Waits for designated principal's approval to start the target Amazon EC2 instance after rolling back the previous changes.
- `onFailureStartTargetInstance` - Starts the target Amazon EC2 instance.
- `terminateClonedEC2Instance` - Terminates the cloned Amazon EC2 instance after replacing the root Amazon EBS volume.

## AWSSupport-ResetAccess

### Description

This runbook will use the EC2Rescue tool on the specified EC2 instance to re-enable password decryption using the EC2 Console (Windows) or to generate and add a new SSH key pair (Linux). If you lost your key pair, this automation will create a password-enabled AMI that you can use to launch a new EC2 instance with a key pair you own (Windows).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows



## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- EC2RescueInstanceType

Type: String

Valid values: t2.small | t2.medium | t2.large

Default: t2.small

Description: (Required) The EC2 instance type for the EC2Rescue instance. Recommended size: t2.small.

- InstanceId

Type: String

Description: (Required) ID of the EC2 instance you want to reset access for.

### Important

Systems Manager Automation stops this instance, and creates an AMI before attempting any operations. Data stored in instance store volumes will be lost. The public IP address will change if you are not using an Elastic IP.

- SubnetId

Type: String

Default: CreateNewVPC

Description: (Optional) The subnet ID for the EC2Rescue instance. By default, Systems Manager Automation creates a new VPC. Alternatively, Use SelectedInstanceSubnet to use the same subnet as your instance, or specify a custom subnet ID.

### Important

The subnet must be in the same Availability Zone as InstanceId, and it must allow access to the SSM endpoints.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

You must have at least `ssm:StartAutomationExecution`, `ssm:GetParameter` (to retrieve the SSH key parameter name) and `ssm:GetAutomationExecution` to be able to read the automation output. For more information about the required permissions, see [AWSSupport-StartEC2RescueWorkflow \(p. 111\)](#).

## Document Steps

- `aws:assertAwsResourceProperty` - Assert if the provided instance is Windows.
  - (EC2Rescue for Windows) If the provided instance is Windows:
    - `aws:executeAutomation` - Invoke `AWSSupport-StartEC2RescueWorkflow` with the EC2Rescue for Windows offline password reset script
    - `aws:executeAwsApi` - Retrieve the backup AMI ID from the nested automation

- iii. `aws:executeAwsApi` - Retrieve the password-enabled AMI ID from the nested automation
- iv. `aws:executeAwsApi` - Retrieve the EC2Rescue summary from the nested automation
- b. (EC2Rescue for Linux) If the provided instance is Linux:
  - i. `aws:executeAutomation` - Invoke `AWSSupport-StartEC2RescueWorkflow` with the EC2Rescue for Linux offline SSH key injection script
  - ii. `aws:executeAwsApi` - Retrieve the backup AMI ID from the nested automation
  - iii. `aws:executeAwsApi` - Retrieve the SSM parameter name for the injected SSH key
  - iv. `aws:executeAwsApi` - Retrieve the EC2Rescue summary from the nested automation

### Outputs

`getEC2RescueForWindowsResult.Output`

`getWindowsBackupAmi.ImageId`

`getWindowsPasswordEnabledAmi.ImageId`

`getEC2RescueForLinuxResult.Output`

`getLinuxBackupAmi.ImageId`

`getLinuxSSHKeyParameter.Name`

## AWSSupport- RestoreEC2InstanceFromSnapshot

### Description

The `AWSSupport-RestoreEC2InstanceFromSnapshot` runbook helps you identify and restore an Amazon Elastic Compute Cloud (Amazon EC2) instance from a working Amazon Elastic Block Store (Amazon EBS) snapshot of the root volume.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `EndDate`

Type: String

Description: (Optional) The last date you want the automation to look for a snapshot.

- InplaceSwap

Type: Boolean

Valid values: true | false

Description: (Optional) If the value for this parameter is set to `true`, the newly created volume from the snapshot replaces the existing root volume attached to your instance.

- InstanceId

Type: String

Description: (Required) The ID of the instance you want to restore from a snapshot.

- LookForInstanceStatusCheck

Type: Boolean

Valid values: true | false

Default: true

Description: (Optional) If the value for this parameter is set to `true`, the automation checks whether instance status checks fail on the test instances launched from the snapshots.

- SkipSnapshotsBy

Type: String

Description: (Optional) The interval at which snapshots are skipped when searching for snapshots to restore your instance. For example, if there are 100 snapshots available, and you specify a value of 2 for this parameter, then every third snapshot is reviewed.

Default: 0

- SnapshotId

Type: String

Description: (Optional) The ID of a snapshot you want to restore the instance from.

- StartDate

Type: String

Description: (Optional) The first date you want the automation to look for a snapshot.

- TotalSnapshotsToLook

Type: String

Description: (Optional) The number of snapshots the automation reviews.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

- `ssm:DescribeInstanceInformation`
- `ec2:AttachVolume`
- `ec2:CreateImage`
- `ec2:CreateTags`
- `ec2:CreateVolume`
- `ec2>DeleteTags`
- `ec2:DeregisterImage`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceState`
- `ec2:DescribeImages`
- `ec2:DescribeSnapshots`
- `ec2:DescribeVolumes`
- `ec2:DetachVolume`
- `ec2:RunInstances`
- `ec2:StartInstances`
- `ec2:StopInstances`
- `ec2:TerminateInstances`
- `cloudwatch:GetMetricData`

### Document Steps

1. `aws:executeAwsApi` - Gathers details about the target instance.
2. `aws:assertAwsResourceProperty` - Verifies the target instance exists.
3. `aws:assertAwsResourceProperty` - Verifies the root volume is an Amazon EBS volume.
4. `aws:assertAwsResourceProperty` - Verifies that another automation isn't already running that targets this instance.
5. `aws:executeAwsApi` - Tags the target instance.
6. `aws:executeAwsApi` - Creates an AMI of the instance.
7. `aws:executeAwsApi` - Gathers details about the AMI created in the previous step.
8. `aws:waitForAwsResourceProperty` - Waits for the AMI state to become available before proceeding.
9. `aws:executeScript` - Launches a new instance from the newly created AMI.
10. `aws:assertAwsResourceProperty` - Verifies the instance state is available.
11. `aws:executeAwsApi` - Gathers details about the newly launched instance.
12. `aws:branch` - Branches based on whether you provided a value for the `SnapshotId` parameter.
13. `aws:executeScript` - Returns a list of snapshots within the time period specified.
14. `aws:executeAwsApi` - Stops the instance.
15. `aws:waitForAwsResourceProperty` - Waits for the volume state to be available.
16. `aws:waitForAwsResourceProperty` - Waits for the instance state to be stopped.
17. `aws:executeAwsApi` - Detaches the root volume.
18. `aws:waitForAwsResourceProperty` - Waits for the root volume to be detached.
19. `aws:executeAwsApi` - Attaches the new root volume.
20. `aws:waitForAwsResourceProperty` - Waits for the new volume to be attached.
21. `aws:executeAwsApi` - Starts the instance.
22. `aws:waitForAwsResourceProperty` - Waits for the instance state to be available.
23. `aws:waitForAwsResourceProperty` - Waits for system and instance status checks to pass for the instance.

24aws:executeScript - Runs a script to find a snapshot that can be used to successfully create a volume.

25aws:executeScript - Runs a script to recover the instance using the newly created volume from the snapshot identified by the automation, or using the volume created from the snapshot you specified in the SnapshotId parameter.

26aws:executeScript - Deletes resources created by the automation.

### Outputs

launchCloneInstance.InstanceIds

ListSnapshotByDate.finalSnapshots

ListSnapshotByDate.remainingSnapshotToBeCheckedInSameDateRange

findWorkingSnapshot.workingSnapshot

InstanceRecovery.result

## AWSSupport-SendLogBundleToS3Bucket

### Description

The AWSSupport-SendLogBundleToS3Bucket runbook uploads a log bundle generated by the EC2Rescue tool from the target instance to the specified S3 bucket. The runbook installs the platform specific version of EC2Rescue based on the platform of the target instance. EC2Rescue is then used to collect all the available operating system (OS) logs.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the Windows or Linux managed instance you want to collect logs from.

- S3BucketName

Type: String

Description: (Required) S3 bucket to upload the logs to.

- S3Path

Type: String

Default: AWSSupport-SendLogBundleToS3Bucket/

Description: (Optional) S3 path for the collected logs.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

It is recommended that the EC2 instance receiving the command has an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. The user must have at least **ssm:StartAutomationExecution** and **ssm:SendCommand** to run the automation and send the command to the instance, plus **ssm:GetAutomationExecution** to be able to read the automation output.

### Document Steps

1. `aws:runCommand` - Install EC2Rescue via `AWS-ConfigureAWSPackage`.
2. `aws:runCommand` - Run the PowerShell script to collect Windows troubleshooting logs with EC2Rescue.
3. `aws:runCommand` - Run the bash script to collect Linux troubleshooting logs with EC2Rescue.

### Outputs

`collectAndUploadWindowsLogBundle.Output`

`collectAndUploadLinuxLogBundle.Output`

## AWSSupport-StartEC2RescueWorkflow

### Description

The `AWSSupport-StartEC2RescueWorkflow` runbook runs the provided base64 encoded script (Bash or Powershell) on a helper instance created to rescue your instance. The root volume of your instance is attached and mounted to the helper instance, also known as the EC2Rescue instance. If your instance is Windows, provide a Powershell script. Otherwise, use Bash. The runbook sets some environment variables which you can use in your script. The environment variables contain information about the input you provided, as well as information about the offline root volume. The offline volume is already mounted and ready to use. For example, you can save a Desired State Configuration file to an offline Windows root volume, or chroot to an offline Linux root volume and perform an offline remediation.

[Run this Automation \(console\)](#)

#### Important

Amazon EC2 instances created from Marketplace Amazon Machine Images (AMIs) are not supported by this automation.

### Additional Information

To base64 encode a script, you can use either Powershell or Bash. Powershell:

```
[System.Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes([System.IO.File]::ReadAllText
```

Bash:

```
base64 PATH_TO_FILE
```

Here is a list of environment variables you can use in your offline scripts, depending on the target OS

Windows:

Variable	Description	Example value
\$env:EC2RESCUE_ACCOUNT_ID	{{ global:ACCOUNT_ID }}	123456789012
\$env:EC2RESCUE_DATE	{{ global:DATE }}	2018-09-07
\$env:EC2RESCUE_DATE_TIME	{{ global:DATE_TIME }}	2018-09-07_18.09.59
\$env:EC2RESCUE_EC2RW_DIR	EC2Rescue for Windows installation path	C:\Program Files\Amazon\EC2Rescue
\$env:EC2RESCUE_EC2RW_DIR	EC2Rescue for Windows installation path	C:\Program Files\Amazon\EC2Rescue
\$env:EC2RESCUE_EXECUTION_ID	{{ automation:EXECUTION_ID }}	7ef8008e-219b-4aca-8bb5-65e2e898e20b
\$env:EC2RESCUE_OFFLINE_CURRENT_CONTROLSET_PATH	Offline Windows Current Control Set path	HKLM:\AWSTempSystem\ControlSet001
\$env:EC2RESCUE_OFFLINE_DRIVE	Offline Windows drive letter	D:\
\$env:EC2RESCUE_OFFLINE_EBS_DEVICE	Offline Windows root volume EBS device	xvdf
\$env:EC2RESCUE_OFFLINE_KERNEL_VERSION	Offline Windows Kernel version	6.1.7601.24214
\$env:EC2RESCUE_OFFLINE_OS_ARCHITECTURE	Offline Windows architecture	AMD64
\$env:EC2RESCUE_OFFLINE_OS_CAPTION	Offline Windows caption	Windows Server 2008 R2 Datacenter
\$env:EC2RESCUE_OFFLINE_OS_TYPE	Offline Windows OS type	Server
\$env:EC2RESCUE_OFFLINE_PROGRAM_FILES_DIR	Offline Windows Program files directory path	D:\Program Files
\$env:EC2RESCUE_OFFLINE_PROGRAM_FILES_DIR_x86	Offline Windows x86 Program files directory path	D:\Program Files (x86)
\$env:EC2RESCUE_OFFLINE_REGISTRY_DIR	Offline Windows registry directory path	D:\Windows\System32\config
\$env:EC2RESCUE_OFFLINE_SYSTEM_ROOT	Offline Windows system root directory path	D:\Windows
\$env:EC2RESCUE_REGION	{{ global:REGION }}	us-west-1
\$env:EC2RESCUE_S3_BUCKET	{{ S3BucketName }}	mybucket
\$env:EC2RESCUE_S3_PREFIX	{{ S3Prefix }}	myprefix/

Variable	Description	Example value
<code>\$env:EC2RESCUE_SOURCE_INSTANCEID</code>	{{ InstanceId }}	i-abcdefgh123456789
<code>\$script:EC2RESCUE_OFFLINE_WINDOWS_INSTALLATION_METADATA</code>	Customer Powershell Installation metadata	Customer Powershell Object

Linux:

Variable	Description	Example value
<code>EC2RESCUE_ACCOUNT_ID</code>	{{ global:ACCOUNT_ID }}	123456789012
<code>EC2RESCUE_DATE</code>	{{ global:DATE }}	2018-09-07
<code>EC2RESCUE_DATE_TIME</code>	{{ global:DATE_TIME }}	2018-09-07_18.09.59
<code>EC2RESCUE_EC2RL_DIR</code>	EC2Rescue for Linux installation path	/usr/local/ec2rl-1.1.3
<code>EC2RESCUE_EXECUTION_ID</code>	{{ automation:EXECUTION_ID }}	7ef8008e-219b-4aca-8bb5-65e2e898e20b
<code>EC2RESCUE_OFFLINE_DEVICE</code>	Offline device name	/dev/xvdf1
<code>EC2RESCUE_OFFLINE_EBS_DEVICE</code>	Offline root volume EBS device	/dev/sdf
<code>EC2RESCUE_OFFLINE_SYSTEM_ROOT</code>	Offline root volume mount point	/mnt/mount
<code>EC2RESCUE_PYTHON</code>	Python version	python2.7
<code>EC2RESCUE_REGION</code>	{{ global:REGION }}	us-west-1
<code>EC2RESCUE_S3_BUCKET</code>	{{ S3BucketName }}	mybucket
<code>EC2RESCUE_S3_PREFIX</code>	{{ S3Prefix }}	myprefix/
<code>EC2RESCUE_SOURCE_INSTANCE</code>	{{ InstanceId }}	i-abcdefgh123456789

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AMIPrefix

Type: String

Default: AWSsupport-EC2Rescue

Description: (Optional) A prefix for the backup AMI name.



- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- CreatePostEC2RescueBackup

Type: String

Valid values: true | false

Default: false

Description: (Optional) Set it to `true` to create an AMI of InstanceId after running the script, before starting it. The AMI will persist after the automation completes. It is your responsibility to secure access to the AMI, or to delete it.

- CreatePreEC2RescueBackup

Type: String

Valid values: true | false

Default: false

Description: (Optional) Set it to `true` to create an AMI of InstanceId before running the script. The AMI will persist after the automation completes. It is your responsibility to secure access to the AMI, or to delete it.

- EC2RescueInstanceType

Type: String

Valid values: t2.small | t2.medium | t2.large

Default: t2.small

Description: (Optional) The EC2 instance type for the EC2Rescue instance.

- InstanceId

Type: String

Description: (Required) ID of your EC2 instance. IMPORTANT: AWS Systems Manager Automation stops this instance. Data stored in instance store volumes will be lost. The public IP address will change if you are not using an Elastic IP.

- OfflineScript

Type: String

Description: (Required) Base64 encoded script to run against the helper instance. Use Bash if your source instance is Linux, and PowerShell if it is Windows.

- S3BucketName

Type: String

Description: (Optional) S3 bucket name in your account where you want to upload the troubleshooting logs. Make sure the bucket policy does not grant unnecessary read/write permissions to parties that do not need access to the collected logs.

- S3Prefix

Type: String

Default: AWSSupport-EC2Rescue

Description: (Optional) A prefix for the S3 logs.

- SubnetId

Type: String

Default: SelectedInstanceSubnet

Description: (Optional) The subnet ID for the EC2Rescue instance. By default, the same subnet where the provided instance resides is used. **IMPORTANT:** If you provide a custom subnet, it must be in the same Availability Zone as InstanceId, and it must allow access to the SSM endpoints.

- UniqueId

Type: String

Default: {{ automation:EXECUTION\_ID }}

Description: (Optional) A unique identifier for the automation.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

It is recommended the user who runs the automation have the **AmazonSSMAutomationRole** IAM managed policy attached. In addition to that policy, the user must have:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "lambda:InvokeFunction",
        "lambda:DeleteFunction",
        "lambda:GetFunction"
      ],
      "Resource": "arn:aws:lambda:*:An-AWS-Account-ID:function:AWSSupport-
EC2Rescue-*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::awssupport-ssm.*/*.template",
        "arn:aws:s3:::awssupport-ssm.*/*.zip"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "iam:CreateRole",
        "iam:CreateInstanceProfile",
        "iam:GetRole",
        "iam:GetInstanceProfile",
```

```
        "iam:PutRolePolicy",
        "iam:DetachRolePolicy",
        "iam:AttachRolePolicy",
        "iam:PassRole",
        "iam:AddRoleToInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile",
        "iam>DeleteRole",
        "iam>DeleteRolePolicy",
        "iam>DeleteInstanceProfile"
    ],
    "Resource": [
        "arn:aws:iam::An-AWS-Account-ID:role/AWSSupport-EC2Rescue-*",
        "arn:aws:iam::An-AWS-Account-ID:instance-profile/AWSSupport-
EC2Rescue-*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "lambda:CreateFunction",
        "ec2:CreateVpc",
        "ec2:ModifyVpcAttribute",
        "ec2>DeleteVpc",
        "ec2:CreateInternetGateway",
        "ec2:AttachInternetGateway",
        "ec2:DetachInternetGateway",
        "ec2>DeleteInternetGateway",
        "ec2:CreateSubnet",
        "ec2>DeleteSubnet",
        "ec2:CreateRoute",
        "ec2>DeleteRoute",
        "ec2:CreateRouteTable",
        "ec2:AssociateRouteTable",
        "ec2:DisassociateRouteTable",
        "ec2>DeleteRouteTable",
        "ec2:CreateVpcEndpoint",
        "ec2>DeleteVpcEndpoints",
        "ec2:ModifyVpcEndpoint",
        "ec2:Describe*"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
}
```

## Document Steps

1. `aws:executeAwsApi` - Describe the provided instance
2. `aws:executeAwsApi` - Describe the provided instance's root volume
3. `aws:assertAwsResourceProperty` - Check the root volume device type is EBS
4. `aws:assertAwsResourceProperty` - Check the root volume is not encrypted
5. `aws:assertAwsResourceProperty` - Check the provide subnet ID
  - a. (Use current instance subnet) - If `*SubnetId = SelectedInstanceSubnet*` then run `aws:createStack` to deploy the EC2Rescue CloudFormation stack
  - b. (Create new VPC) - If `*SubnetId = CreateNewVPC*` then run `aws:createStack` to deploy the EC2Rescue CloudFormation stack
  - c. (Use custom subnet) - In all other cases:

`aws:assertAwsResourceProperty` - Check the provided subnet is in the same Availability Zone as the provided instance

- `aws:createStack` - Deploy the EC2Rescue CloudFormation stack
6. `aws:invokeLambdaFunction` - Perform additional input validation
  7. `aws:executeAwsApi` - Update the EC2Rescue CloudFormation stack to create the EC2Rescue helper instance
  8. `aws:waitForAwsResourceProperty` - Wait for the EC2Rescue CloudFormation stack update to complete
  9. `aws:executeAwsApi` - Describe the EC2Rescue CloudFormation stack output to obtain the EC2Rescue helper instance ID
  10. `aws:waitForAwsResourceProperty` - Wait for the EC2Rescue helper instance to become a managed instance
  11. `aws:changeInstanceState` - Stop the provided instance
  12. `aws:changeInstanceState` - Stop the provided instance
  13. `aws:changeInstanceState` - Force stop the provided instance
  14. `aws:assertAwsResourceProperty` - Check the `CreatePreEC2RescueBackup` input value
    - a. (Create pre-EC2Rescue backup) - If `*CreatePreEC2RescueBackup = true*`
    - b. `aws:executeAwsApi` - Create an AMI backup of the provided instance
    - c. `aws:createTags` - Tag the AMI backup
  15. `aws:runCommand` - Install EC2Rescue on the EC2Rescue helper instance
  16. `aws:executeAwsApi` - Detach the root volume from the provided instance
  17. `aws:assertAwsResourceProperty` - Check the provided instance platform
    - a. (Instance is Windows):  
  
`aws:executeAwsApi` - Attach the root volume to the EC2Rescue helper instance as `*xvdf*`  
  
`aws:sleep` - Sleep 10 seconds  
  
`aws:runCommand` - Run the provided offline script in Powershell
    - b. (Instance is Linux):  
  
`aws:executeAwsApi` - Attach the root volume to the EC2Rescue helper instance as `*/dev/sdf*`  
  
`aws:sleep` - Sleep 10 seconds  
  
`aws:runCommand` - Run the provided offline script in Bash
  18. `aws:changeInstanceState` - Stop the EC2Rescue helper instance
  19. `aws:changeInstanceState` - Force stop the EC2Rescue helper instance
  20. `aws:executeAwsApi` - Detach the root volume from the EC2Rescue helper instance
  21. `aws:executeAwsApi` - Attach the root volume back to the provided instance
  22. `aws:assertAwsResourceProperty` - Check the `CreatePostEC2RescueBackup` input value
    - a. (Create post-EC2Rescue backup) - If `*CreatePostEC2RescueBackup = true*`
    - b. `aws:executeAwsApi` - Create an AMI backup of the provided instance
    - c. `aws:createTags` - Tag the AMI backup
  23. `aws:executeAwsApi` - Restore the initial delete on termination state for the root volume of the provided instance
  24. `aws:changeInstanceState` - Restore the initial state of the provided instance (running/stopped)
  25. `aws:deleteStack` - Delete the EC2Rescue CloudFormation stack

## Outputs

`runScriptForLinux.Output`

runScriptForWindows.Output

preScriptBackup.ImageId

postScriptBackup.ImageId

## AWSPremiumSupport- TroubleshootEC2DiskUsage

### Description

The `AWSPremiumSupport-TroubleshootEC2DiskUsage` runbook helps you investigate and potentially remediate issues with Amazon Elastic Compute Cloud (Amazon EC2) instance root and non-root disk usage. If possible, the runbook attempts to remediate issues by extending the volume and its file system. To perform these tasks, this runbook orchestrates the execution of several runbooks based on the operating system of the affected instance.

The first runbook, `AWSPremiumSupport-DiagnoseDiskUsageOnWindows` or `AWSPremiumSupport-DiagnoseDiskUsageOnLinux`, determines if disk issues can be mitigated by expanding the volume.

The second runbook, `AWSPremiumSupport-ExtendVolumesOnWindows` or `AWSPremiumSupport-ExtendVolumesOnLinux`, uses the output of the first runbook to run Python code that modifies the volume. After the volume has been modified, the runbook extends the partition and file system of the affected volumes.

#### Important

Access to `AWSPremiumSupport-*` runbooks requires an Enterprise or Business Support Subscription. For more information, see [Compare AWS Support Plans](#).

This document was built in collaboration with AWS Managed Services (AMS). AMS helps you manage your AWS infrastructure more efficiently and securely. AMS also provides operational flexibility, enhanced security and compliance, capacity optimization, and cost-savings identification. For more information, see [AWS Managed Services](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, Windows

### Parameters

- `InstanceId`

Type: String

Allowed values: `^[a-z0-9]{8,17}$`

Description: (Required) ID of your Amazon EC2 instance.

- `VolumeExpansionEnabled`

Type: Boolean

Description: (Optional) Flag to control whether the document will extend the volumes and partitions affected.

Default: true

- VolumeExpansionUsageTrigger

Type: String

Description: (Optional) Minimum usage of partition space required to trigger extension (in percentage).

Allowed values: `^[0-9]{1,2}$`

Default: 85

- VolumeExpansionCapSize

Type: String

Description: (Optional) Maximum size that the Amazon Elastic Block Store (Amazon EBS) volume will be increased to (in GiB).

Allowed values: `^[0-9]{1,4}$`

Default: 2048

- VolumeExpansionGibIncrease

Type: String

Description: (Optional) Increase in GiB of the volume. The biggest net increase between VolumeExpansionGibIncrease and VolumeExpansionPercentageIncrease will be used.

Allowed values: `^[0-9]{1,4}$`

Default: 20

- VolumeExpansionPercentageIncrease

Type: String

Description: (Optional) Increase in percentage of the volume. The biggest net increase between VolumeExpansionGibIncrease and VolumeExpansionPercentageIncrease will be used.

Allowed values: `^[0-9]{1,2}$`

Default: 20

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ec2:DescribeVolumes`
- `ec2:DescribeVolumesModifications`

- `ec2:ModifyVolume`
- `ec2:DescribeInstances`
- `ec2:CreateImage`
- `ec2:DescribeImages`
- `ec2:DescribeTags`
- `ec2:CreateTags`
- `ec2>DeleteTags`
- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:DescribeAutomationExecutions`
- `ssm:SendCommand`
- `ssm:DescribeInstanceInformation`
- `ssm:ListCommands`
- `ssm:ListCommandInvocations`

### Document Steps

1. `aws:assertAwsResourceProperty` - Check if the instance is managed by Systems Manager
2. `aws:executeAwsApi` - Describes the instance to get the platform.
3. `aws:branch` - Branches automation based on the instance's platform.
  - a. If the instance is Windows:
    - i. `aws:executeAutomation` - Run the `AWSPremiumSupport-DiagnoseDiskUsageOnWindows` runbook in order to diagnose disk usage issues on the instance.
    - ii. `aws:executeAwsApi` - Gets the output of the previous automation.
    - iii. `aws:branch` - Branches based on the output of the diagnostics, and if there are volumes that can be expanded to mitigate the alert.
      - A. There are no volumes that need to be expanded: End the automation.
      - B. There are volumes that need to be expanded:
        - I. `aws:executeAwsApi` - Create an Amazon Machine Image (AMI) of the instance.
        - II. `aws:waitForAwsResourceProperty` - Waits for the AMI state to be available.
        - III. `aws:executeAutomation` - Run the `AWSPremiumSupport-ExtendVolumesOnWindows` runbook in order to perform the volume modification as well as the required steps in the operating system (OS) to make the new space available.
  - b. (Platform is not windows) If the input instance is not Windows:
    - i. `aws:executeAutomation` - Run the `AWSPremiumSupport-DiagnoseDiskUsageOnLinux` runbook in order to diagnose disk usage issues on the instance.
    - ii. `aws:executeAwsApi` - Gets the output of the previous automation.
    - iii. `aws:branch` - Branches based on the output of the diagnostics, and if there are volumes that can be expanded to mitigate the alert.
      - A. There are no volumes that need to be expanded: End the automation.
      - B. There are volumes that need to be expanded:
        - I. `aws:executeAwsApi` - Create an AMI of the instance.
        - II. `aws:waitForAwsResourceProperty` - Waits for AMI state to be available.
        - III. `aws:executeAutomation` - Run the `AWSPremiumSupport-ExtendVolumesOnLinux` runbook in order to perform the volume modification as well as the required steps in the OS to make the new space available.

## Outputs

diagnoseDiskUsageAlertOnWindows.Output

extendVolumesOnWindows.Output

diagnoseDiskUsageAlertOnLinux.Output

extendVolumesOnLinux.Output

BackupAMILinux.Imageld

BackupAMIWindows.Imageld

## AWSSupport-TroubleshootRDP

### Description

The `AWSSupport-TroubleshootRDP` runbook allows the user to check or modify common settings on the target instance which may impact Remote Desktop Protocol (RDP) connections, such as the RDP port, Network Layer Authentication (NLA) and Windows Firewall profiles. Optionally, changes can be applied offline by stopping and starting the instance, if the user explicitly allows for offline remediation. By default, the runbook reads and outputs the values of the settings.

#### Important

Changes to the RDP settings, RDP service and Windows Firewall profiles should be carefully reviewed before using this runbook.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Windows

### Parameters

- Action

Type: String

Valid values: CheckAll | FixAll | Custom

Default: Custom

Description: (Optional) [Custom] Use the values from Firewall, RDPServiceStartupType, RDPServiceAction, RDPPortAction, NLASettingAction and RemoteConnections to manage the settings. [CheckAll] Read the values of the settings without changing them. [FixAll] Restore RDP default settings, and disable the Windows Firewall.

- AllowOffline

Type: String



Valid values: true | false

Default: false

Description: (Optional) Fix only - Set it to true if you allow an offline RDP remediation in case the online troubleshooting fails, or the provided instance is not a managed instance. Note: For the offline remediation, SSM Automation stops the instance, and creates an AMI before attempting any operations.

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Firewall

Type: String

Valid values: Check | Disable

Default: Check

Description: (Optional) Check or disable the Windows firewall (all profiles).

- InstanceId

Type: String

Description: (Required) The ID of the instance to troubleshoot the RDP settings of.

- NLASettingAction

Type: String

Valid values: Check | Disable

Default: Check

Description: (Optional) Check or disable Network Layer Authentication (NLA).

- RDPPortAction

Type: String

Valid values: Check | Modify

Default: Check

Description: (Optional) Check the current port used for RDP connections, or modify the RDP port back to 3389 and restart the service.

- RDPServiceAction

Type: String

Valid values: Check | Start | Restart | Force-Restart

Default: Check

Description: (Optional) Check, start, restart, or force-restart the RDP service (TermService).

- **RDPServiceStartupType**  
Type: String  
Valid values: Check | Auto  
Default: Check  
Description: (Optional) Check or set the RDP service to automatically start when Windows boots.
- **RemoteConnections**  
Type: String  
Valid values: Check | Enable  
Default: Check  
Description: (Optional) An action to perform on the fDenyTSConnections setting: Check, Enable.
- **S3BucketName**  
Type: String  
Description: (Optional) Offline only - S3 bucket name in your account where you want to upload the troubleshooting logs. Make sure the bucket policy does not grant unnecessary read/write permissions to parties that do not need access to the collected logs.
- **SubnetId**  
Type: String  
Default: SelectedInstanceSubnet  
Description: (Optional) Offline only - The subnet ID for the EC2Rescue instance used to perform the offline troubleshooting. If no subnet ID is specified, AWS Systems Manager Automation will create a new VPC. **IMPORTANT:** The subnet must be in the same Availability Zone as InstanceId, and it must allow access to the SSM endpoints.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

It is recommended that the EC2 instance receiving the command has an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. For the online remediation, the user must have at least **ssm:DescribeInstanceInformation**, **ssm:StartAutomationExecution** and **ssm:SendCommand** to run the automation and send the command to the instance, plus **ssm:GetAutomationExecution** to be able to read the automation output. For the offline remediation, the user must have at least **ssm:DescribeInstanceInformation**, **ssm:StartAutomationExecution**, **ec2:DescribeInstances**, plus **ssm:GetAutomationExecution** to be able to read the automation output. `AWSSupport-TroubleshootRDP` calls `AWSSupport-ExecuteEC2Rescue` to perform the offline remediation - please review the permissions for `AWSSupport-ExecuteEC2Rescue` to ensure you can run the automation successfully.

### Document Steps

1. `aws:assertAwsResourceProperty` - Check if the instance is a Windows Server instance
2. `aws:assertAwsResourceProperty` - Check if the instance is a managed instance
3. (Online troubleshooting) If the instance is a managed instance, then:
  - a. `aws:assertAwsResourceProperty` - Check the provided Action value
  - b. (Online check) If the **Action = CheckAll**, then:

`aws:runPowerShellScript` - Runs the PowerShell script to get the Windows Firewall profiles status.

`aws:executeAutomation` - Calls `AWSSupport-ManageWindowsService` to get the RDP service status.

`aws:executeAutomation` - Calls `AWSSupport-ManageRDPSettings` to get the RDP settings.

- c. (Online fix) If the **Action = FixAll**, then:

`aws:runPowerShellScript` - Runs the PowerShell script to disable all Windows Firewall profiles.

`aws:executeAutomation` - Calls `AWSSupport-ManageWindowsService` to start the RDP service.

`aws:executeAutomation` - Calls `AWSSupport-ManageRDPSettings` to enable remote connections and disable NLA.

- d. (Online management) If the **Action = Custom**, then:

`aws:runPowerShellScript` - Runs the PowerShell script to manage the Windows Firewall profiles.

`aws:executeAutomation` - Calls `AWSSupport-ManageWindowsService` to manage the RDP service.

`aws:executeAutomation` - Calls `AWSSupport-ManageRDPSettings` to manage the RDP settings.

4. (Offline remediation) If the instance is not a managed instance then:

- a. `aws:assertAwsResourceProperty` - Assert **AllowOffline = true**

- b. `aws:assertAwsResourceProperty` - Assert **Action = FixAll**

- c. `aws:assertAwsResourceProperty` - Assert the value of `SubnetId`

(Use the provided instance's subnet) If `SubnetId` is `SELECTED_INSTANCE_SUBNET`

`aws:executeAwsApi` - Retrieve the current instance's subnet.

`aws:executeAutomation` - Run `AWSSupport-ExecuteEC2Rescue` with provided instance's subnet.

- d. (Use the provided custom subnet) If `SubnetId` is not `SELECTED_INSTANCE_SUBNET`

`aws:executeAutomation` - Run `AWSSupport-ExecuteEC2Rescue` with provided `SubnetId` value.

## Outputs

`manageFirewallProfiles.Output`

`manageRDPServiceSettings.Output`

`manageRDPSettings.Output`

`checkFirewallProfiles.Output`

`checkRDPServiceSettings.Output`

`checkRDPSettings.Output`

`disableFirewallProfiles.Output`

restoreDefaultRDPSvcSettings.Output

restoreDefaultRDPSvcSettings.Output

troubleshootRDPOffline.Output

troubleshootRDPOfflineWithSubnetId.Output

## AWSSupport-TroubleshootSSH

### Description

The `AWSSupport-TroubleshootSSH` runbook installs the Amazon EC2Rescue tool for Linux, and then uses the EC2Rescue tool to check or attempt to fix common issues that prevent a remote connection to the Linux machine via SSH. Optionally, changes can be applied offline by stopping and starting the instance, if the user explicitly allows for offline remediation. By default, the runbook operates in read-only mode.

[Run this Automation \(console\)](#)

For information about working with the `AWSSupport-TroubleshootSSH` runbook, see this [AWSSupport-TroubleshootSSH troubleshooting topic](#) from AWS Premium Support.

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- Action

Type: String

Valid values: CheckAll | FixAll

Default: CheckAll

Description: (Required) Specify whether to check for issues without fixing them or to check and automatically fix any discovered issues.

- AllowOffline

Type: String

Valid values: true | false

Default: false

Description: (Optional) Fix only - Set it to true if you allow an offline SSH remediation in case the online troubleshooting fails, or the provided instance is not a managed instance. Note: For the offline remediation, SSM Automation stops the instance, and creates an AMI before attempting any operations.

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) ID of your EC2 instance for Linux.

- S3BucketName

Type: String

Description: (Optional) Offline only - S3 bucket name in your account where you want to upload the troubleshooting logs. Make sure the bucket policy does not grant unnecessary read/write permissions to parties that do not need access to the collected logs.

- SubnetId

Type: String

Default: SelectedInstanceSubnet

Description: (Optional) Offline only - The subnet ID for the EC2Rescue instance used to perform the offline troubleshooting. If no subnet ID is specified, AWS Systems Manager Automation will create a new VPC.

### Important

The subnet must be in the same Availability Zone as InstanceId, and it must allow access to the SSM endpoints.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

It is recommended that the EC2 instance receiving the command has an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. For the online remediation, the user must have at least **ssm:DescribeInstanceInformation**, **ssm:StartAutomationExecution** and **ssm:SendCommand** to run the automation and send the command to the instance, plus **ssm:GetAutomationExecution** to be able to read the automation output. For the offline remediation, the user must have at least **ssm:DescribeInstanceInformation**, **ssm:StartAutomationExecution**, **ec2:DescribeInstances**, plus **ssm:GetAutomationExecution** to be able to read the automation output. `AWSSupport-TroubleshootSSH` calls `AWSSupport-ExecuteEC2Rescue` to perform the offline remediation - please review the permissions for `AWSSupport-ExecuteEC2Rescue` to ensure you can run the automation successfully.

## Document Steps

1. `aws:assertAwsResourceProperty` - Check if the instance is a managed instance
  - a. (Online remediation) If the instance is a managed instance, then:
    - i. `aws:configurePackage` - Install EC2Rescue for Linux via `AWS-ConfigureAWSPackage`.
    - ii. `aws:runCommand` - Run the bash script to run EC2Rescue for Linux.
  - b. (Offline remediation) If the instance is not a managed instance then:
    - i. `aws:assertAwsResourceProperty` - Assert **AllowOffline = true**
    - ii. `aws:assertAwsResourceProperty` - Assert **Action = FixAll**

- iii. `aws:assertAwsResourceProperty` - Assert the value of SubnetId
- iv. (Use the provided instance's subnet) If SubnetId is SelectedInstanceSubnet use `aws:executeAutomation` to run `AWSSupport-ExecuteEC2Rescue` with provided instance's subnet.
- v. (Use the provided custom subnet) If SubnetId is not SelectedInstanceSubnet use `aws:executeAutomation` to run `AWSSupport-ExecuteEC2Rescue` with provided SubnetId value.

### Outputs

troubleshootSSH.Output

troubleshootSSHOffline.Output

troubleshootSSHOfflineWithSubnetId.Output

## AWSSupport-TroubleshootSUSERegistration

### Description

The `AWSSupport-TroubleshootSUSERegistration` runbook helps you to identify why registering an Amazon Elastic Compute Cloud (Amazon EC2) SUSE Linux Enterprise Server instance with SUSE Update Infrastructure failed. The automation output provides steps to resolve, or helps you troubleshoot, the issue. If the instance passes all checks during the automation, the instance is registered with SUSE Update Infrastructure.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the Amazon EC2 instance you want to troubleshoot.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:DescribeInstanceProperties`
- `ssm:DescribeInstanceInformation`
- `ssm:ListCommandInvocations`
- `ssm:SendCommand`
- `ssm:ListCommands`

### Document Steps

- `aws:assertAwsResourceProperty` - Checks if the Amazon EC2 instance is managed by AWS Systems Manager.
- `aws:runCommand` - Checks if the Amazon EC2 instance platform is SLES.
- `aws:runCommand` - Checks if the package `cloud-regionsrv-client` version is greater than or equal to the required version 9.0.10.
- `aws:runCommand` - Checks if the symbolic link for base product is broken, and fixes the link if it is broken.
- `aws:runCommand` - Checks if the hosts file (`/etc/hosts`) contains records for `smt-ec2-susecloud.net`. The automation removes any duplicate entries.
- `aws:runCommand` - Checks if the `curl` command is installed.
- `aws:runCommand` - Checks if the Amazon EC2 instance can access the Instance Metadata Service (IMDS) address 169.254.169.254.
- `aws:runCommand` - Checks if the Amazon EC2 instance has a billing code or AWS Marketplace product code.
- `aws:runCommand` - Checks if the Amazon EC2 instance can reach at least 1 regional server over HTTPS.
- `aws:runCommand` - Checks if the Amazon EC2 instance can reach the Subscription Management Tool (SMT) servers over HTTP.
- `aws:runCommand` - Checks if the Amazon EC2 instance can reach the Subscription Management Tool (SMT) servers over HTTPS.
- `aws:runCommand` - Checks if the Amazon EC2 instance can reach the `smt-ec2.susecloud.net` address over HTTPS.
- `aws:runCommand` - Registers the Amazon EC2 instance with SUSE Update Infrastructure.
- `aws:executeScript` - Gathers and outputs the output of all the previous steps.

## AWSSupport-UpgradeWindowsAWSDrivers

### Description

The `AWSSupport-UpgradeWindowsAWSDrivers` runbook upgrades or repairs storage and network AWS drivers on the specified EC2 instance. The runbook attempts to install the latest versions of AWS drivers online by calling SSM Agent. If SSM Agent is not contactable, the runbook can perform an offline installation of the AWS drivers if explicitly requested.

### Note

Both the online and offline upgrade will create an AMI before attempting any operations, which will persist after the automation completes. It is your responsibility to secure access to the AMI, or to delete it. The online method restarts the instance as part of the upgrade process, while the offline method requires the provided EC2 instance be stopped and then started.

### Important

If your instances connect to AWS Systems Manager using VPC endpoints, this runbook will fail unless used in the us-east-1 Region. This runbook will also fail on a domain controller. To update AWS PV drivers on a domain controller, see [Upgrade a Domain Controller \(AWS PV Upgrade\)](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AllowOffline

Type: String

Valid values: true | false

Default: false

Description: (Optional) Set it to true if you allow an offline drivers upgrade in case the online installation cannot be performed. Note: The offline method requires the provided EC2 instance be stopped and then started. Data stored in instance store volumes will be lost. The public IP address will change if you are not using an Elastic IP.

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ForceUpgrade

Type: String

Valid values: true | false

Default: false

Description: (Optional) Offline only - Set it to true if you allow the offline drivers upgrade to proceed even though your instance already has the latest drivers installed.

- InstanceId

Type: String

Description: (Required) ID of your EC2 instance for Windows Server.

- SubnetId

Type: String



Default: SelectedInstanceSubnet

Description: (Optional) Offline only - The subnet ID for the EC2Rescue instance used to perform the offline drivers upgrade. If no subnet ID is specified, Systems Manager Automation will create a new VPC.

**Important**

The subnet must be in the same Availability Zone as InstanceId, and it must allow access to the SSM endpoints.

**Required IAM permissions**

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

The EC2 instance receiving the command must at minimum have an IAM role that includes permissions for **ssm:StartAutomationExecution** and **ssm:SendCommand** to run the automation and send the command to the instance, plus **ssm:GetAutomationExecution** to be able to read the automation output. You can attach the AmazonSSManagedInstanceCore Amazon managed policy to your IAM role to provide these permissions. We recommend, however, using the Automation IAM role AmazonSSMAutomationRole for this purpose. For more information, see [Use IAM to configure roles for Automation](#).

If you are performing an offline upgrade, see the permissions required by [AWSSupport-StartEC2RescueWorkflow \(p. 111\)](#).

**Document Steps**

1. `aws:assertAwsResourceProperty` - Verifies the input instance is Windows.
2. `aws:assertAwsResourceProperty` - Verifies the input instance is a managed instance. If so, the online upgrade starts, otherwise the offline upgrade is evaluated.
  - a. (Online upgrade) If the input instance is a managed instance:
    - i. `aws:createImage` - Creates an AMI backup.
    - ii. `aws:createTags` - Tags the AMI backup.
    - iii. `aws:runCommand` - Installs ENA network driver via `AWS-ConfigureAWSPackage`.
    - iv. `aws:runCommand` - Installs NVMe driver via `AWS-ConfigureAWSPackage`.
    - v. `aws:runCommand` - Installs AWS PV driver via `AWS-ConfigureAWSPackage`.
  - b. (Offline upgrade) If the input instance is not a managed instance:
    - i. `aws:assertAwsResourceProperty` - Verifies the AllowOffline flag is set to `true`. If so, the offline upgrade starts, otherwise the automation ends.
    - ii. `aws:changeInstanceState` - Stop the source instance.
    - iii. `aws:changeInstanceState` - Force-stop the source instance.
    - iv. `aws:createImage` - Create an AMI backup of the source instance.
    - v. `aws:createTags` - Tag the AMI backup of the source instance.
    - vi. `aws:executeAwsApi` - Enable ENA for the instance
    - vii. `aws:assertAwsResourceProperty` - Assert the ForceUpgrade flag.
    - viii. (Offline upgrade) If **ForceUpgrade = true** then run `aws:executeAutomation` to invoke `AWSSupport-StartEC2RescueWorkflow` with the drivers force upgrade script. This installs the drivers regardless of the current version that is installed
    - ix. (Offline upgrade) If **ForceUpgrade = false** then run `aws:executeAutomation` to invoke `AWSSupport-StartEC2RescueWorkflow` with the drivers upgrade script.

**Outputs**

```
preUpgradeBackup.Imageld  
preOfflineUpgradeBackup.Imageld  
installAwsEnaNetworkDriverOnInstance.Output  
installAWSNVMeOnInstance.Output  
installAWSPVDriverOnInstance.Output  
upgradeDriversOffline.Output  
forceUpgradeDriversOffline.Output
```

## Amazon ECS

AWS Systems Manager Automation provides predefined runbooks for Amazon Elastic Container Service. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSSupport-CollectECSInstanceLogs \(p. 131\)](#)
- [AWS-InstallECSContainerAgent \(p. 133\)](#)
- [AWSSupport-TroubleshootECSContainerInstance \(p. 134\)](#)
- [AWS-UpdateECSContainerAgent \(p. 135\)](#)

## AWSSupport-CollectECSInstanceLogs

### Description

The `AWSSupport-CollectECSInstanceLogs` runbook collects operating system and Amazon Elastic Container Service (Amazon ECS) related log files from an Amazon Elastic Compute Cloud (Amazon EC2) instance to help you troubleshoot common Amazon ECS issues. While the automation is collecting the associated log files, changes are made to the file system. These changes include the creation of temporary directories and a log directory, the copying of log files to these directories, and compressing the log files into an archive.

If you specify a value for the `LogDestination` parameter, the automation evaluates the policy status of the Amazon Simple Storage Service (Amazon S3) bucket you specify. To help with the security of the logs gathered from your Amazon EC2 instance, if the policy status `isPublic` is set to `true` , or if the access control list (ACL) grants `READ|WRITE` permissions to the `All Users` Amazon S3 predefined group, the logs are not uploaded. Additionally, if the provided bucket is not available in your account, the logs are not uploaded. For more information about Amazon S3 predefined groups, see [Amazon S3 predefined groups](#) in the *Amazon Simple Storage Service User Guide* .

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Linux, Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ECSInstanceId

Type: String

Description: (Required) The ID of the instance you want to collect logs from. The instance you specify must be managed by Systems Manager.

- LogDestination

Type: String

Description: (Optional) The Amazon S3 bucket in your AWS account to upload the archived logs to.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ssm:SendCommand
- ssm:DescribeInstanceInformation

We recommend that the Amazon EC2 instance you specify in the ECSInstanceId parameter has an IAM role with the AmazonSSManagedInstanceCore Amazon managed policy attached. To upload the log archive to the Amazon S3 bucket you specify in the LogDestination parameter, you must add following permissions:

- s3:PutObject
- s3:ListBucket
- s3:GetBucketPolicyStatus
- s3:GetBucketAcl

## Document Steps

- assertInstanceIsManaged - Verifies whether the instance you specify in the ECSInstanceId parameter is managed by Systems Manager.
- getInstancePlatform - Gets information about the operating system (OS) platform of the instance specified in the ECSInstanceId parameter.
- verifyInstancePlatform - Branches the automation based on the OS platform.
- runLogCollectionScriptOnLinux - Gathers operating system and Amazon ECS related log files on Linux instances and creates an archive file in the /var/log/collectECSlogs directory.

- `runLogCollectionScriptOnWindows` - Gathers operating system and Amazon ECS related log files on Windows instances and creates an archive file in the `C:\ProgramData\collectECSlogs` directory.
- `verifyIfS3BucketProvided` - Verifies whether a value was specified for the `LogDestination` parameter.
- `runUploadScript` - Branches the automation step based on the OS platform.
- `runUploadScriptOnLinux` - Uploads the log archive to the Amazon S3 bucket specified in the `LogDestination` parameter and deletes the archived log file from OS.
- `runUploadScriptOnWindows` - Uploads the log archive to the Amazon S3 bucket specified in the `LogDestination` parameter and deletes the archived log file from OS.

## AWS-InstallECSContainerAgent

### Description

The `AWS-InstallECSContainerAgent` runbook installs the Amazon Elastic Container Service (Amazon ECS) agent on the Amazon Elastic Compute Cloud (Amazon EC2) instance you specify. This runbook only supports Amazon Linux and Amazon Linux 2 instances.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `InstanceIds`

Type: StringList

Description: (Required) The IDs of the Amazon EC2 instances you want to install the Amazon ECS agent on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ssm:GetCommandInvocation`

- `ec2:DescribeImages`
- `ec2:DescribeInstanceAttribute`
- `ec2:DescribeInstances`

### Document Steps

`aws:executeScript` - Installs the Amazon ECS agent on the Amazon EC2 instances you specify in the `InstanceIds` parameter.

### Outputs

`InstallAmazonECSAgent.SuccessfulInstances` - The ID of the instance where installation of the Amazon ECS agent succeeded.

`InstallAmazonECSAgent.FailedInstances` - The ID of the instance where installation of the Amazon ECS agent failed.

`InstallAmazonECSAgent.InProgressInstances` - The ID of the instance where installation of the Amazon ECS agent is in progress.

## AWSSupport-TroubleshootECSContainerInstance

### Description

The `AWSSupport-TroubleshootECSContainerInstance` runbook helps you troubleshoot an Amazon Elastic Compute Cloud (Amazon EC2) instance that fails to register with an Amazon ECS cluster. This automation reviews whether the user data for the instance contains the correct cluster information, whether the instance profile contains the required permissions, and network configuration issues.

#### Important

To successfully run this automation, the state of your Amazon EC2 instance must be `running`, and the Amazon ECS cluster state must be `ACTIVE`.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `ClusterName`

Type: String

Description: (Required) The name of the Amazon ECS cluster that the instance failed to register with.

- InstanceId

Type: String

Description: (Required) The ID of the Amazon EC2 instance you want to troubleshoot.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ec2:DescribeIamInstanceProfileAssociations`
- `ec2:DescribeInstanceAttribute`
- `ec2:DescribeInstances`
- `ec2:DescribeNetworkAcls`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcEndpoints`
- `ec2:DescribeVpcs`
- `iam:GetInstanceProfile`
- `iam:GetRole`
- `iam:SimulateCustomPolicy`
- `iam:SimulatePrincipalPolicy`

### Document Steps

`aws:executeScript`: Reviews whether the Amazon EC2 instance meets the prerequisites needed to register with an Amazon ECS cluster.

## AWS-UpdateECSContainerAgent

### Description

The `AWS-UpdateECSContainerAgent` runbook updates the Amazon Elastic Container Service (Amazon ECS) agent on the Amazon Elastic Compute Cloud (Amazon EC2) instance you specify. This runbook only supports Amazon Linux and Amazon Linux 2 instances.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ClusterARN

Type: StringList

Description: (Required) The Amazon Resource Name (ARN) of the Amazon ECS cluster your container instances is registered with.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ssm:GetCommandInvocation
- ec2:DescribeImages
- ec2:DescribeInstanceAttribute
- ec2:DescribeImage
- ec2:DescribeInstance
- ec2:DescribeInstanceAttribute
- ecs:DescribeContainerInstances
- ecs:DescribeClusters
- ecs>ListContainerInstances
- ecs:UpdateContainerAgent

### Document Steps

aws:executeScript - Updates the Amazon ECS agent on the Amazon ECS cluster you specify in the ClusterARN parameters.

### Outputs

UpdateAmazonECSAgent.UpdatedContainers - The ID of the instance where the update of the Amazon ECS agent succeeded.

UpdateAmazonECSAgent.FailedContainers - The ID of the instance where the update of the Amazon ECS agent failed.

UpdateAmazonECSAgent.InProgressContainers - The ID of the instance where the update of the Amazon ECS agent is in progress.

## Amazon EFS

AWS Systems Manager Automation provides predefined runbooks for Amazon Elastic File System. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

## Topics

- [AWSSupport-CheckAndMountEFS \(p. 137\)](#)

# AWSSupport-CheckAndMountEFS

## Description

The `AWSSupport-CheckAndMountEFS` runbook verifies the prerequisites to mount your Amazon Elastic File System (Amazon EFS) file system and mounts the file system on the Amazon Elastic Compute Cloud (Amazon EC2) instance you specify. This runbook supports mounting your Amazon EFS file system with the DNS name, or using the mount target's IP address.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux

## Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `Action`

Type: String

Valid values: `Check` | `CheckAndMount`

Description: (Required) Determines whether the runbook verifies prerequisites, or verifies prerequisites and mounts the file system.

- `EfsId`

Type: String

Description: (Required) The ID of the file system you want to mount.

- `InstanceId`

Type: String

Description: (Required) The ID of the Amazon EC2 instance on which you want to mount the file system.

- `MountOptions`

Type: String



Description: (Optional) The options supported by the Amazon EFS mount helper that you want to use when mounting the file system. If you specify the `tls` option, verify stunnel has been upgraded on the target instance.

- `MountPoint`

Type: String

Description: (Optional) The directory where you want to mount the file system. If you specify the `Check` value for the `Action` parameter, this parameter should not be specified.

- `MountTargetIP`

Type: String

Description: (Optional) The mount target's IP address. Mounting by IP address works in environments where DNS is disabled, such as virtual private clouds (VPCs) with DNS hostnames disabled. Also, you can use this option if your environment uses a DNS provider other than Amazon Route 53 (Route 53).

- `Region`

Type: String

Description: (Required) The AWS Region where the Amazon EC2 instance and file system are located.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:DescribeAutomationExecutions`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:DescribeInstanceInformation`
- `ssm:DescribeInstanceProperties`
- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ssm:GetDocument`
- `ssm:ListCommands`
- `ssm:ListCommandInvocations`
- `ssm:ListDocuments`
- `ssm:StartAutomationExecution`
- `iam:ListRoles`
- `ec2:DescribeInstances`
- `ec2:DescribeSecurityGroups`
- `elasticfilesystem:DescribeFileSystemPolicy`
- `elasticfilesystem:DescribeMountTargets`
- `elasticfilesystem:DescribeMountTargetSecurityGroups`
- `resource-groups:*`

### Document Steps

- `aws:executeScript` - Gathers details about the Amazon EC2 instance you specify in the `InstanceId` parameter.

- `aws:executeScript` - Gathers details about the file system you specify in the `EfsId` parameter.
- `aws:executeScript` - Verifies the security group associated with the file system allows traffic on port 2049 from the Amazon EC2 instance you specify in the `InstanceId` parameter.
- `aws:assertAwsResourceProperty` - Verifies the Amazon EC2 instance you specify in the `InstanceId` parameter is managed by Systems Manager and that the status is `Online`.
- `aws:branch` - Branches based on the value you specify for the `Action` parameter.
- `aws:runCommand` - Verifies prerequisites for mounting the file system you specify in the `EfsId` parameter.
- `aws:runCommand` - Verifies prerequisites for mounting the file system you specify in the `EfsId` parameter, and mounts the file system on the Amazon EC2 instance you specify in the `InstanceId` parameter.

## Amazon EKS

AWS Systems Manager Automation provides predefined runbooks for Amazon Elastic Kubernetes Service. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWSSupport-CollectEKSIInstanceLogs \(p. 139\)](#)
- [AWS-DeleteEKSCluster \(p. 140\)](#)
- [AWSPremiumSupport-TroubleshootEKSCluster \(p. 142\)](#)
- [AWSSupport-TroubleshootEKSWorkerNode \(p. 145\)](#)
- [AWS-UpdateEKSMangedNodegroupVersion \(p. 147\)](#)

## AWSSupport-CollectEKSIInstanceLogs

### Description

The `AWSSupport-CollectEKSIInstanceLogs` runbook gathers operating system and Amazon Elastic Kubernetes Service (Amazon EKS) related log files from an Amazon Elastic Compute Cloud (Amazon EC2) instance to help you troubleshoot common issues. While the automation is gathering the associated log files, changes are made to the file system structure including the creation of temporary directories, the copying of log files to the temporary directories, and compressing the log files into an archive. This activity can result in increased CPU utilization on the EC2 instance. For more information about CPU utilization, see [Instance metrics](#) in the *Amazon CloudWatch User Guide*.

If you specify a value for the `LogDestination` parameter, the automation evaluates the policy status of the Amazon Simple Storage Service (Amazon S3) bucket you specify. To help with the security of the logs gathered from your EC2 instance, if the policy status `isPublic` is set to `true`, or if the access control list (ACL) grants `READ|WRITE` permissions to the `All Users` Amazon S3 predefined group, the logs are not uploaded. For more information about Amazon S3 predefined groups, see [Amazon S3 predefined groups](#) in the *Amazon Simple Storage Service User Guide*.

### Note

This automation requires at least 10 percent of available disk space on the root Amazon Elastic Block Store (Amazon EBS) volume attached to your EC2 instance. If there is not enough available disk space on the root volume, the automation stops.

[Run this Automation \(console\)](#)

### Document type

Automation

**Owner**

Amazon

**Platforms**

Linux

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- EKSInstanceId

Type: String

Description: (Required) ID of the Amazon EKS EC2 instance you want to collect logs from.

- LogDestination

Type: String

Description: (Optional) The S3 bucket in your account to upload the archived logs to.

**Required IAM permissions**

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ssm:SendCommand`

We recommend that the EC2 instance receiving the command has an IAM role with the **AmazonSSMManagedInstanceCore** Amazon managed policy attached. To upload the log archive to the S3 bucket you specify in the LogDestination parameter, you must add the `s3:PutObject` permission.

**Document Steps**

- `aws:assertAwsResourceProperty` - Confirms the operating system of the value specified in the EKSInstanceId parameter is Linux.
- `aws:runCommand` - Gathers operating system and Amazon EKS related log files, compressing them into an archive in the `/var/log` directory.
- `aws:branch` - Confirms whether a value was specified for the LogDestination parameter.
- `aws:runCommand` - Uploads the log archive to the S3 bucket you specify in the LogDestination parameter.

## AWS-DeleteEKSCluster

**Description**

This runbook deletes the resources associated with an Amazon EKS cluster, including node groups and Fargate profiles. Optionally, you can choose to delete all self-managed nodes, the AWS CloudFormation stacks used to create the nodes, and the VPC CloudFormation stack for your cluster. For more information about deleting a cluster, see [Deleting a cluster](#) in the *Amazon EKS User Guide*.

**Note**

If you have active services in your cluster that are associated with a load balancer, you must delete those services before deleting the cluster. If you don't, the system can't delete the load balancers. Use the following procedure to find and delete services before you run the `AWS-DeleteEKSCluster` runbook.

**To locate and delete services in your cluster**

1. Install the Kubernetes command line utility, `kubectl`. For more information, see [Installing kubectl](#) in the *Amazon EKS User Guide*.
2. Run the following command to list all services running in your cluster.

```
kubectl get svc --all-namespaces
```

3. Run the following command to delete any services that have an associated EXTERNAL-IP value. These services are fronted by a load balancer, and you must delete them in Kubernetes to allow the load balancer and associated resources to be properly released.

```
kubectl delete svc  
service-name
```

You can now run the `AWS-DeleteEKSCluster` runbook.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- EKSClusterName

Type: String

Description: (Required) The name of the Amazon EKS Cluster to be deleted.

- VPCCloudFormationStack

Type: String

Description: (Optional) AWS CloudFormation stack name for VPC for the EKS cluster being deleted. This deletes the AWS CloudFormation stack for VPC and any resources created by the stack.

- VPCCloudFormationStackRole

Type: String

Description: (Optional) The ARN of an IAM role that AWS CloudFormation assumes to delete the VPC CloudFormation stack. AWS CloudFormation uses the role's credentials to make calls on your behalf.

- SelfManagedNodeStacks

Type: String

Description: (Optional) Comma-separated list of AWS CloudFormation stack names for self-managed nodes, This will delete the AWS CloudFormation stacks for self-managed nodes.

- SelfManagedNodeStacksRole

Type: String

Description: (Optional) The ARN of an IAM role that AWS CloudFormation assumes to delete the Self-managed Node Stacks. AWS CloudFormation uses the role's credentials to make calls on your behalf.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `sts:AssumeRole`
- `eks:ListNodegroups`
- `eks>DeleteNodegroup`
- `eks:ListFargateProfiles`
- `eks>DeleteFargateProfile`
- `eks>DeleteCluster`
- `cfn:DescribeStacks`
- `cfn>DeleteStack`

### Document Steps

- `aws:executeScript` - `DeleteNodeGroups`: Find and delete all node groups in the EKS cluster.
- `aws:executeScript` - `DeleteFargateProfiles`: Find and delete all Fargate profiles in the EKS cluster.
- `aws:executeScript` - `DeleteSelfManagedNodes`: Delete all self-managed nodes and the CloudFormation stacks used to create the nodes.
- `aws:executeScript` - `DeleteEKSCluster`: Delete EKS cluster.
- `aws:executeScript` - `DeleteVPCCloudFormationStack`: Delete the VPC CloudFormation stack.

## AWSPremiumSupport-TroubleshootEKSCluster

### Description

The `AWSPremiumSupport-TroubleshootEKSCluster` runbook diagnoses common issues with an Amazon Elastic Kubernetes Service (Amazon EKS) cluster, underlying infrastructure, and provides recommended remediation steps.

### Important

Access to `AWSPremiumSupport-*` runbooks requires either an Enterprise or Business Support Subscription. For more information, see [Compare AWS Support Plans](#).

If you specify a value for the `S3BucketName` parameter, the automation evaluates the policy status of the Amazon Simple Storage Service (Amazon S3) bucket you specify. To help with the security of the logs gathered from your EC2 instance, if the policy status `isPublic` is set to `true`, or if the access control list (ACL) grants `READ|WRITE` permissions to the `All Users` Amazon S3 predefined group, the logs are not uploaded. For more information about Amazon S3 predefined groups, see [Amazon S3 predefined groups](#) in the *Amazon Simple Storage Service User Guide*.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `ClusterName`

Type: String

Description: (Required) The name of the Amazon EKS cluster that you want to troubleshoot.

- `S3BucketName`

Type: String

Description: (Optional) The name of the private Amazon S3 bucket where the report generated by the runbook should be uploaded.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceTypes`

- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeRouteTables`
- `ec2:DescribeNatGateways`
- `ec2:DescribeVpcs`
- `ec2:DescribeNetworkAcls`
- `iam:GetInstanceProfile`
- `iam:ListInstanceProfiles`
- `iam:ListAttachedRolePolicies`
- `eks:DescribeCluster`
- `eks:ListNodegroups`
- `eks:DescribeNodegroup`
- `autoscaling:DescribeAutoScalingGroups`

In addition, the AWS Identity and Access Management (IAM) policy attached to the IAM user or role that starts the automation must allow the `ssm:GetParameter` operation to the following public AWS Systems Manager parameters to get the latest recommended Amazon EKS Amazon Machine Image (AMI) for the worker nodes.

- `arn:aws:ssm::parameter/aws/service/eks/optimized-ami/*/amazon-linux-2/recommended/image_id`
- `arn:aws:ssm::parameter/aws/service/ami-windows-latest/Windows_Server-2019-English-Core-EKS_Optimized-*/image_id`
- `arn:aws:ssm::parameter/aws/service/ami-windows-latest/Windows_Server-2019-English-Full-EKS_Optimized-*/image_id`
- `arn:aws:ssm::parameter/aws/service/ami-windows-latest/Windows_Server-1909-English-Core-EKS_Optimized-*/image_id`
- `arn:aws:ssm::parameter/aws/service/eks/optimized-ami/*/amazon-linux-2-gpu/recommended/image_id`

To upload the report generated by the runbook to an Amazon S3 bucket, the following permissions are required for the specified Amazon S3 bucket you specify.

- `s3:GetBucketPolicyStatus`
- `s3:GetBucketAcl`
- `s3:PutObject`

### Document Steps

- `aws:executeAwsApi` - Gathers details for the specified Amazon EKS cluster.
- `aws:executeScript` - Gathers details of the Amazon Elastic Compute Cloud (Amazon EC2) instances, Auto Scaling groups, AMIs, and Amazon EC2 GPU graphic instance types.
- `aws:executeScript` - Gathers details of the virtual private cloud (VPC), subnets, network address translation (NAT) gateways, subnet routes, security groups and network access control lists (ACLs) of the Amazon EKS cluster.
- `aws:executeScript` - Gathers details of attached IAM instance profiles and role policies.
- `aws:executeScript` - Gathers details of the Amazon S3 bucket you specify in the `S3BucketName` parameter.
- `aws:executeScript` - Classifies the Amazon VPC subnets as public or private.

- `aws:executeScript` - Checks the Amazon VPC subnets for tags that are required as part of an Amazon EKS cluster.
- `aws:executeScript` - Checks the Amazon VPC subnets for the tags that are required for Elastic Load Balancing subnets.
- `aws:executeScript` - Checks if the worker node Amazon EC2 instances use the latest Amazon EKS optimized AMIs
- `aws:executeScript` - Checks if the Amazon VPC security groups attached to worker nodes for the tags that are required.
- `aws:executeScript` - Checks the Amazon EKS cluster and worker node Amazon VPC security group rules for the recommended ingress rules to the Amazon EKS cluster.
- `aws:executeScript` - Checks the Amazon EKS cluster and worker node Amazon VPC security group rules for the recommended egress rules from the Amazon EKS cluster.
- `aws:executeScript` - Checks the network ACL configuration of the Amazon VPC subnets.
- `aws:executeScript` - Checks if the worker node Amazon EC2 instances have the required managed policies.
- `aws:executeScript` - Checks if the Auto Scaling groups have the necessary tags for cluster autoscaling.
- `aws:executeScript` - Checks if the worker node Amazon EC2 instances are connected to the internet.
- `aws:executeScript` - Generates a report based on the outputs from the previous steps. If a value is specified for the `S3BucketName` parameter, the generated report is uploaded to the Amazon S3 bucket.

## AWSSupport-TroubleshootEKSWorkerNode

### Description

The `AWSSupport-TroubleshootEKSWorkerNode` runbook analyzes an Amazon Elastic Compute Cloud (Amazon EC2) worker node and Amazon Elastic Kubernetes Service (Amazon EKS) cluster to help you identify and troubleshoot common causes that prevent worker nodes from joining a cluster. The runbook outputs guidance to help you resolve any issues that are identified.

#### Important

To successfully run this automation, the state of your Amazon EC2 worker node must be `running`, and the Amazon EKS cluster state must be `ACTIVE`.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- `AutomationAssumeRole`

Type: String



Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ClusterName

Type: String

Description: (Required) The name of the Amazon EKS cluster.

- WorkerID

Type: String

Description: (Required) The ID of the Amazon EC2 worker node that failed to join the cluster.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ec2:DescribeDhcpOptions`
- `ec2:DescribeImages`
- `ec2:DescribeInstanceAttribute`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceState`
- `ec2:DescribeNatGateways`
- `ec2:DescribeNetworkAcls`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcAttribute`
- `ec2:DescribeVpcEndpoints`
- `ec2:DescribeVpcs`
- `eks:DescribeCluster`
- `iam:GetInstanceProfile`
- `iam:GetRole`
- `iam:ListAttachedRolePolicies`
- `ssm:DescribeInstanceInformation`
- `ssm:ListCommandInvocations`
- `ssm:ListCommands`
- `ssm:SendCommand`

### Document Steps

- `aws:assertAwsResourceProperty` - Confirms that the Amazon EKS cluster you specify in the `ClusterName` parameter exists and is in an `ACTIVE` state.
- `aws:assertAwsResourceProperty` - Confirms that the Amazon EC2 worker node you specify in the `WorkerID` parameter exists and is in a `running` state.
- `aws:executeScript` - Runs a Python script that helps identify possible causes for the worker node failing to join the cluster.

# AWS-UpdateEKSMangedNodegroupVersion

## Description

This runbook updates managed node groups in your Amazon EKS cluster to the latest AMI version. For more information about this update process, see [Updating a managed node group](#) in the *Amazon EKS User Guide*. We also recommend that you review the following topics before you use the `AWS-UpdateEKSMangedNodegroupVersion` runbook.

- [Managed node groups](#)
- [Managed node update behavior](#)
- [UpdateNodegroupVersion](#)

If your cluster uses autoscaling, scale the deployment down to zero replicas to avoid conflicting scaling actions.

## To scale a deployment to zero replicas

1. Install the Kubernetes command line utility, `kubectl`. For more information, see [Installing kubectl](#) in the *Amazon EKS User Guide*.
2. Run the following command.

```
kubectl scale deployments/cluster-autoscaler --replicas=0 -n kube-system
```

3. Run the `AWS-UpdateEKSMangedNodegroupVersion` runbook.
4. Scale the deployment back to the desired number of replicas by running the following command.

```
kubectl scale deployments/cluster-autoscaler --replicas=  
number  
-n kube-system
```

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your

behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `ClusterName`

Type: String

Description: (Required) The name of the Amazon EKS cluster.

- `NodeGroupName`

Type: String

Description: (Required) The name of the managed node group.

- `LaunchTemplateVersion`

Type: String

Description: (Optional) The Amazon Elastic Compute Cloud (Amazon EC2) launch template version. This parameter is only valid if a node group was created from a launch template.

- `ForceUpgrade`

Type: Boolean

Description: (Optional) If true, the update won't fail in response to a pod disruption budget violation.

#### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `eks:DescribeNodegroup`
- `eks:UpdateNodegroupVersion`

#### Document Steps

`aws:executeScript - UpdateEKSMangedNodegroupVersion`: Updates the AMI version used by a managed node group in an Amazon EKS cluster.

## Elastic Beanstalk

AWS Systems Manager Automation provides predefined runbooks for AWS Elastic Beanstalk. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

#### Topics

- [AWSSupport-CollectElasticBeanstalkLogs \(p. 148\)](#)
- [AWSConfigRemediation-EnableElasticBeanstalkEnvironmentLogStreaming \(p. 150\)](#)
- [AWSConfigRemediation-EnableBeanstalkEnvironmentNotifications \(p. 151\)](#)

## AWSSupport-CollectElasticBeanstalkLogs

#### Description

The `AWSSupport-CollectElasticBeanstalkLogs` runbook gathers AWS Elastic Beanstalk related log files from an Amazon Elastic Compute Cloud (Amazon EC2) Windows Server instance launched

by Elastic Beanstalk to help you troubleshoot common issues. While the automation is gathering the associated log files, changes are made to the file system structure including the creation of temporary directories, the copying of log files to the temporary directories, and compressing the log files into an archive. This activity can result in increased CPU utilization on the Amazon EC2 instance. For more information about CPU utilization, see [Instance metrics](#) in the *Amazon CloudWatch User Guide*.

If you specify a value for the `S3BucketName` parameter, the automation evaluates the policy status of the Amazon Simple Storage Service (Amazon S3) bucket you specify. To help with the security of the logs gathered from your Amazon EC2 instance, if the policy status `isPublic` is set to `true`, or if the access control list (ACL) grants `READ|WRITE` permissions to the `All Users` Amazon S3 predefined group, the logs are not uploaded. For more information about Amazon S3 predefined groups, see [Amazon S3 predefined groups](#) in the *Amazon Simple Storage Service User Guide*.

If you do not specify a value for the `S3BucketName` parameter, the automation uploads the log bundle to the default Elastic Beanstalk Amazon S3 bucket in the AWS Region where you run the automation. The directory is named according to the following structure, `elasticbeanstalk-region - accountID`. The *region* and *accountID* values will differ based on the Region and AWS account you run the automation in. The log bundle will be saved to the `resources/environments/logs/bundle/environmentID / instanceID` directory. The *environmentID* and *instanceID* values will differ based on your Elastic Beanstalk environment and the Amazon EC2 instance you're gathering logs from.

By default, the AWS Identity and Access Management (IAM) instance profile attached to the Amazon EC2 instances of the Elastic Beanstalk environment has the required permissions to upload the bundle to the default Elastic Beanstalk Amazon S3 bucket for your environment. If you specify a value for the `S3BucketName` parameter, the instance profile attached to the Amazon EC2 instance must allow the `s3:GetBucketAcl`, `s3:GetBucketPolicy`, `s3:GetBucketPolicyStatus`, and `s3:PutObject` actions for the specified Amazon S3 bucket and path.

**Note**

This automation requires at least 500 MB of available disk space on the root Amazon Elastic Block Store (Amazon EBS) volume attached to your Amazon EC2 instance. If there is not enough available disk space on the root volume, the automation stops.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- EnvironmentId

Type: String

Description: (Required) The ID of your Elastic Beanstalk environment you want to collect the log bundle from.

- InstanceId

Type: String

(Required) The ID of the Amazon EC2 instance in your Elastic Beanstalk environment you want to collect the log bundle from.

- S3BucketName

Type: String

(Optional) The Amazon S3 bucket you want to upload the archived logs to.

- S3BucketPath

Type: String

(Optional) The Amazon S3 bucket path you want to upload the log bundle to. This parameter is ignored if you do not specify a value for the S3BucketName parameter.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ssm:SendCommand
- ssm:DescribeInstanceInformation
- ec2:DescribeInstances

### Document Steps

- aws:assertAwsResourceProperty - Confirms the Amazon EC2 instance you specify in the InstanceId parameter is managed by AWS Systems Manager.
- aws:assertAwsResourceProperty - Confirms the Amazon EC2 instance you specify in the InstanceId parameter is a Windows Server instance.
- aws:runCommand - Checks whether the instance is part of an Elastic Beanstalk environment, if there is sufficient disk space to bundle the logs, and whether the Amazon S3 bucket to which the logs would be uploaded to is public.
- aws:runCommand - Collects the log files and uploads the archive to the Amazon S3 bucket specified in the S3BucketName parameter or to the default bucket for your Elastic Beanstalk environment if a value is not specified.

## AWSConfigRemediation- EnableElasticBeanstalkEnvironmentLogStreaming

### Description

The AWSConfigRemediation-EnableElasticBeanstalkEnvironmentLogStreaming runbook enables logging on the AWS Elastic Beanstalk (Elastic Beanstalk) environment you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[w+=,./-]+\d{12}

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- EnvironmentId

Type: String

Description: (Required) The ID of the Elastic Beanstalk environment that you want to enable logging on.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- elasticbeanstalk:DescribeConfigurationSettings
- elasticbeanstalk:DescribeEnvironments
- elasticbeanstalk:UpdateEnvironment

### Document Steps

- aws:executeAwsApi - Enables logging on the Elastic Beanstalk environment you specify in the EnvironmentId parameter.
- aws:waitForAwsResourceProperty - Waits for the status of the environment to change to Ready .
- aws:executeScript - Verifies logging has been enabled on the Elastic Beanstalk environment.

## AWSConfigRemediation- EnableBeanstalkEnvironmentNotifications

### Description

The AWSConfigRemediation-EnableBeanstalkEnvironmentNotifications runbook enables notifications for the AWS Elastic Beanstalk (Elastic Beanstalk) environment you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- EnvironmentId

Type: String

Description: (Required) The ID of the Elastic Beanstalk environment that you want to enable notifications for.

- TopicArn

Type: String

Description: (Required) The ARN of the Amazon Simple Notification Service (Amazon SNS) topic you want to send notifications to.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- elasticbeanstalk:DescribeConfigurationSettings
- elasticbeanstalk:DescribeEnvironments
- elasticbeanstalk:UpdateEnvironment

### Document Steps

- aws:executeAwsApi - Enables notifications for the Elastic Beanstalk environment you specify in the EnvironmentId parameter.
- aws:waitForAwsResourceProperty - Waits for the status of the environment to change to Ready .

- `aws:executeScript` - Verifies notifications have been enabled for the Elastic Beanstalk environment.

## Elastic Load Balancing

AWS Systems Manager Automation provides predefined runbooks for Elastic Load Balancing. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-DropInvalidHeadersForALB \(p. 153\)](#)
- [AWSConfigRemediation-EnableCLBCrossZoneLoadBalancing \(p. 154\)](#)
- [AWSConfigRemediation-EnableELBDeletionProtection \(p. 155\)](#)
- [AWSConfigRemediation-EnableLoggingForALBAndCLB \(p. 156\)](#)
- [AWSSupport-TroubleshootCLBConnectivity \(p. 157\)](#)
- [AWSConfigRemediation-EnableNLBCrossZoneLoadBalancing \(p. 159\)](#)

## AWSConfigRemediation-DropInvalidHeadersForALB

### Description

The `AWSConfigRemediation-DropInvalidHeadersForALB` runbook enables the application load balancer you specify to remove HTTP headers with invalid headers.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([w+=,._@/-]+)$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LoadBalancerArn`

Type: String



Description: (Required) The Amazon Resource Name (ARN) of the load balancer that you want to drop invalid headers.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `elasticloadbalancing:DescribeLoadBalancerAttributes`
- `elasticloadbalancing:ModifyLoadBalancerAttributes`

### Document Steps

- `aws:executeAwsApi` - Enables the drop invalid headers setting for the load balancer you specify in the `LoadBalancerArn` parameter.
- `aws:executeScript` - Verifies the drop invalid headers setting has been enabled on the load balancer you specify in the `LoadBalancerArn` parameter.

## AWSConfigRemediation- EnableCLBCrossZoneLoadBalancing

### Description

The `AWSConfigRemediation-EnableCLBCrossZoneLoadBalancing` runbook enables cross-zone load balancing for the Classic Load Balancer (CLB) you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LoadBalancerName`

Type: String

Description: (Required) The name of the CLB that you want to enable cross-zone load balancing on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `elb:DescribeLoadBalancerAttributes`
- `elb:ModifyLoadBalancerAttributes`

### Document Steps

- `aws:executeAwsApi` - Enables cross-zone load balancing for the CLB you specify in the `LoadBalancerName` parameter.
- `aws:assertAwsResourceProperty` - Verifies cross-zone load balancing has been enabled on the CLB.

## AWSConfigRemediation- EnableELBDeletionProtection

### Description

The `AWSConfigRemediation-EnableELBDeletionProtection` runbook enables deletion protection for the elastic load balancer (ELB) you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LoadBalancerArn`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the ELB that you want to enable deletion protection on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `elasticloadbalancing:DescribeLoadBalancerAttributes`
- `elasticloadbalancing:DescribeLoadBalancers`
- `elasticloadbalancing:ModifyLoadBalancerAttributes`

### Document Steps

- `aws:executeScript` - Enables deletion protection on the ELB you specify in the `LoadBalancerArn` parameter.

## AWSConfigRemediation- EnableLoggingForALBAndCLB

### Description

The `AWSConfigRemediation-EnableLoggingForALBAndCLB` runbook enables logging for the specified AWS Application Load Balancer or a Classic Load Balancer (CLB).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @/ -\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LoadBalancerId`

Type: String

Description: (Required) The Classic Load Balancer name or the Application Load Balancer ARN.

- S3BucketName

Type: String

Description: (Required) The Amazon S3 bucket name.

- S3BucketPrefix

Type: String

Description: (Optional) The logical hierarchy you created for your Amazon Simple Storage Service (Amazon S3) bucket, for example `my-bucket-prefix/prod`. If the prefix is not provided, the log is placed at the root level of the bucket.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `elasticloadbalancing:DescribeLoadBalancerAttributes`
- `elasticloadbalancing:ModifyLoadBalancerAttributes`

### Document Steps

- `aws:executeScript` - Enables and verifies the logging for the Classic Load Balancer or the Application Load Balancer.

## AWSSupport-TroubleshootCLBConnectivity

### Description

The `AWSSupport-TroubleshootCLBConnectivity` runbook help you troubleshoot connectivity issues between a Classic Load Balancer (CLB) and Amazon Elastic Compute Cloud (Amazon EC2) instances. Also, connectivity issues between a client and the CLB are reviewed. This runbook also reviews health checks for the CLB, verifies that best practices are being followed, and creates a troubleshooting dashboard for you.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InvestigationType

Type: String

Valid values: Best Practices | Connectivity Issues | Troubleshooting Dashboard

Description: (Required) The operations you want the runbook to perform.

- LoadBalancerName

Type: String

Description: (Required) The name of the CLB.

- S3Location

Type: String

Description: (Optional) The name of the Amazon Simple Storage Service (Amazon S3) bucket you want to send the automation results to.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ec2:DescribeInstances
- ec2:DescribeNetworkAcls
- ec2:DescribeRouteTables
- ec2:DescribeSecurityGroups
- ec2:DescribeVpcs
- ec2:DescribeSubnets
- elasticloadbalancing:DescribeLoadBalancers
- elasticloadbalancing:DescribeLoadBalancerPolicies
- elasticloadbalancing:DescribeInstanceHealth
- elasticloadbalancing:DescribeLoadBalancerAttributes
- iam:ListRoles
- cloudwatch:PutDashboard
- ssm:GetAutomationExecution
- ssm:StartAutomationExecution
- s3:PutObject

### Document Steps

- aws:executeScript - Verifies that the CLB you specify in the LoadBalancerName parameter exists.
- aws:branch - Branches based on the value specified for the InvestigationType parameter.
- aws:executeScript - Performs connectivity checks to the CLB.

- `aws:executeScript` - Verifies that the CLB configuration adheres to Elastic Load Balancing best practices.
- `aws:executeScript` - Creates an Amazon CloudWatch dashboard for your CLB.
- `aws:executeScript` - Creates a text file with the results of the automation and uploads it to the Amazon S3 bucket you specify in the `S3Location` parameter.

### Outputs

`runConnectivityChecks.output`

`runBestPractices.Output`

`createTroubleshootingDashboard.Output`

`verifyInputs.Output`

## AWSConfigRemediation- EnableNLBCrossZoneLoadBalancing

### Description

The `AWSConfigRemediation-EnableNLBCrossZoneLoadBalancing` runbook enables cross zone load balancing for the network load balancer (NLB) you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LoadBalancerArn`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the NLB that you want to enable cross zone load balancing on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `elbv2:DescribeLoadBalancerAttributes`
- `elbv2:ModifyLoadBalancerAttributes`

#### Document Steps

- `aws:executeAwsApi` - Enables cross zone load balancing for the NLB you specify in the `LoadBalancerArn` parameter.
- `aws:executeScript` - Verifies cross zone load balancing has been enabled on the NLB.

## Amazon EMR

AWS Systems Manager Automation provides predefined runbooks for Amazon EMR. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

#### Topics

- [AWSSupport-AnalyzeEMRLogs \(p. 160\)](#)

## AWSSupport-AnalyzeEMRLogs

#### Description

This runbook helps identify errors while running a job on an Amazon EMR cluster. The runbook analyzes a list of defined logs on the file system and looks for a list of predefined keywords. These log entries are used to create Amazon CloudWatch Events events so you can take any needed actions based on the events. Optionally, the runbook publishes log entries to the Amazon CloudWatch Logs log group of your choosing. This runbook currently looks for the following errors and patterns in log files:

- `container_out_of_memory` – YARN container ran out of memory, running job may fail.
- `yarn_nodemanager_health`: CORE or TASK node is running low on disk space and will not be able to run tasks.
- `node_state_change`: CORE or TASK node is unreachable by the MASTER node.
- `step_failure`: An EMR Step has failed.
- `no_core_nodes_running`: No CORE nodes are currently running, cluster is unhealthy.
- `hdfs_missing_blocks`: There are missing HDFS blocks which could lead to data loss.
- `hdfs_high_util`: HDFS Utilization is high, which may affect jobs and cluster health.
- `instance_controller_restart`: Instance-Controller process has restarted. This process is essential for cluster health.
- `instance_controller_restart_legacy`: Instance-Controller process has restarted. This process is essential for cluster health.
- `high_load`: High Load Average detected, may affect node health reporting or result in timeouts or slowdowns.
- `yarn_node_blacklisted`: CORE or TASK node has been blacklisted by YARN from running tasks.
- `yarn_node_lost`: CORE or TASK node has been marked as LOST by YARN, possible connectivity issues.

Instances associated with the `ClusterID` that you specify must be managed by AWS Systems Manager. You can run this automation once, schedule the automation to run at a specific time interval, or remove a schedule created previously by an automation. This runbook supports Amazon EMR release versions 5.20 to 6.30.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ClusterID

Type: String

Description: (Required) The ID of the cluster whose nodes logs you want to analyze.

- Operation

Type: String

Valid values: Run Once | Schedule | Remove Schedule

Description: (Required) The operation to perform on the cluster.

- IntervalTime

Type: String

Valid values: 5 minutes | 10 minutes | 15 minutes

Description: (Optional) The duration of time between running the automation. This parameter is only applicable if you specify `Schedule` for the `Operation` parameter.

- LogToCloudWatchLogs

Type: String

Valid values: yes | no

Description: (Optional) If you specify `yes` for the value of this parameter, the automation creates a CloudWatch Logs log group with the name specified in the `CloudWatchLogGroup` parameter to store any matched log entries.

- CloudWatchLogGroup



Type: String

Description: (Optional) The name of the CloudWatch Logs log group you want to store any matched log entries in. This parameter is only applicable if you specify `yes` for the `LogToCloudWatchLogs` parameter.

- `CreateLogInsightsDashboard`

Type: String

Valid values: `yes` | `no`

Description: (Optional) If you specify `yes`, CloudWatch dashboard is created if it does not already exist. This parameter is only applicable if you specify `yes` for the `LogToCloudWatchLogs` parameter.

- `CreateMetricFilters`

Type: String

Valid values: `yes` | `no`

Description: (Optional) Specify `yes` if you want to create metric filters for the CloudWatch Logs log group. This parameter is only applicable if you specify `yes` for the `LogToCloudWatchLogs` parameter.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetDocument`
- `ssm:ListDocuments`
- `ssm:DescribeAutomationExecutions`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:GetAutomationExecution`
- `ssm:DescribeInstanceInformation`
- `ssm:ListCommandInvocations`
- `ssm:ListCommands`
- `ssm:SendCommand`
- `iam:CreateRole`
- `iam>DeleteRole`
- `iam:GetRolePolicy`
- `iam:PutRolePolicy`
- `iam>DeleteRolePolicy`
- `iam:passrole`
- `cloudformation:DescribeStacks`
- `cloudformation>DeleteStack`
- `cloudformation>CreateStack`
- `events>DeleteRule`
- `events:RemoveTargets`
- `events:PutTargets`
- `events:PutRule`

- `events:DescribeRule`
- `logs:DescribeLogGroups`
- `logs>CreateLogGroup`
- `logs:PutMetricFilter`
- `cloudwatch:PutDashboard`
- `elasticmapreduce:ListInstances`
- `elasticmapreduce:DescribeCluster`

## Document Steps

- `aws:executeAwsApi` - Gathers information about the Amazon EMR cluster specified in the `ClusterID` parameter.
- `aws:branch` - Branches based on input.
  - If the provided operation is `Run Once` or `Schedule` :
    - `aws:assertAwsResourceProperty` - Verifies the cluster is available.
    - `aws:executeAwsApi` - Gathers the IDs of all instances running in the cluster.
    - `aws:assertAwsResourceProperty` - Verifies the SSM Agent is running on all instances in the cluster.
  - `aws:branch` - Branches based on whether you specified to run the automation once or on a schedule.
    - If the provided operation is `Run Once` :
      - `aws:branch` - Branches based on the value specified in the `LogToCloudWatchLogs` parameter.
      - If `LogToCloudWatchLogs` value is `yes` :
        - `aws:executeScript` - Checks if a CloudWatch Logs log group with the name specified in parameter `CloudWatchLogGroup` already exists. If not, the group is created with the name specified.
        - `aws:branch` - Branches based on the value specified in the `CreateMetricFilters` parameter.
          - If `CreateMetricFilters` value is `yes` :
            - `aws:executeAwsApi` - 12 steps are ran for each metric filter
            - `aws:branch` - Branches based on the value specified in the `CreateLogInsightsDashboard` parameter.
              - If `CreateLogInsightsDashboard` value is `yes` :
                - `aws:executeAwsApi` - Creates a CloudWatch dashboard with the same name specified in the `CloudWatchLogGroup` parameter, if it does not already exist.
                - If `CreateLogInsightsDashboard` value is `no` :
                  - `aws:runCommand` - Runs a shell script to find log patterns on each instance in the cluster.
    - If `CreateMetricFilters` value is `no` :
      - `aws:branch` - Branches based on the value specified in `CreateLogInsightsDashboard` parameter.
        - If `CreateLogInsightsDashboard` value is `yes` :
          - `aws:executeAwsApi` - Creates a CloudWatch dashboard with the same name specified in the `CloudWatchLogGroup` parameter, if it does not already exist.
          - If `CreateLogInsightsDashboard` value is `no` :
            - `aws:runCommand` - Runs a shell script to find log patterns on each instance in the cluster.
    - If `LogToCloudWatchLogs` value is `no` :

- `aws:executeAwsApi` - Runs a shell script to find log patterns on each instance in the cluster.
- If the provided operation is `Schedule` :
  - `aws:createStack` - Creates an Amazon EventBridge event that targets this runbook.
- If the provided operation is `Remove Schedule` :
  - `aws:executeAwsApi` - Verifies a schedule exists for the cluster.
  - `aws:deleteStack` - Deletes the schedule.

### Outputs

`GetClusterInformation.ClusterName`

`GetClusterInformation.ClusterState`

`ListingClusterInstances.InstanceIDs`

`CreatingScheduleCloudFormationStack.StackStatus`

`RemovingScheduleByDeletingScheduleCloudFormationStack.StackStatus`

`CheckIfLogGroupExists.output`

`FindLogPatternOnEMRNode.CommandId`

## Amazon OpenSearch Service

AWS Systems Manager Automation provides predefined runbooks for Amazon OpenSearch Service. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-DeleteOpenSearchDomain \(p. 164\)](#)
- [AWSConfigRemediation-EnforceHTTPSOnOpenSearchDomain \(p. 165\)](#)
- [AWSConfigRemediation-UpdateOpenSearchDomainSecurityGroups \(p. 166\)](#)

## AWSConfigRemediation-DeleteOpenSearchDomain

### Description

The `AWSConfigRemediation-DeleteOpenSearchDomain` runbook deletes the given Amazon OpenSearch Service domain using the [DeleteDomain](#) API.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- DomainName

Type: String

Allowed values: (\d{12}/)?[a-z]{1}[a-z0-9-]{2,28}

Description: (Required) The name of the Amazon OpenSearch Service domain that you want to delete.

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-]+\d{12}

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- es>DeleteDomain
- es:DescribeDomain

## Document Steps

- aws:executeScript - Accepts the Amazon OpenSearch Service domain name as input, deletes it, and verifies the deletion.

# AWSConfigRemediation- EnforceHTTPSOnOpenSearchDomain

## Description

The AWSConfigRemediation-EnforceHTTPSOnOpenSearchDomain runbook enables EnforceHTTPS on a given Amazon OpenSearch Service domain using the [UpdateDomainConfig](#) API.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

### Parameters

- DomainName

Type: String

Allowed values: (\d{12}/)?[a-z]{1}[a-z0-9-]{2,28}

Description: (Required) The name of the Amazon OpenSearch Service domain that you want to use to enforce HTTPS.

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[w+=,./-]+\d{12}

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `es:DescribeDomain`
- `es:UpdateDomainConfig`

### Document Steps

- `aws:executeScript` - Enables the `EnforceHTTPS` endpoint option on the Amazon OpenSearch Service domain you specify in the `DomainName` parameter.

## AWSConfigRemediation- UpdateOpenSearchDomainSecurityGroups

### Description

The `AWSConfigRemediation-UpdateOpenSearchDomainSecurityGroups` runbook updates the security group configuration on a given Amazon OpenSearch Service domain using the [UpdateDomainConfig](#) API.

#### Note

AWS Security groups can only be applied to Amazon OpenSearch Service domains configured for Amazon Virtual Private Cloud (VPC) Access, and not to Amazon OpenSearch Service domains configured for Public Access.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `DomainName`

Type: String

Description: (Required) The name of the Amazon OpenSearch Service domain that you want to use to update security groups.

- `SecurityGroupList`

Type: StringList

Description: (Required) The security group IDs that you want to assign to the Amazon OpenSearch Service domain.

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `es:DescribeDomain`
- `es:UpdateDomainConfig`

### Document Steps

- `aws:executeScript` - Updates the security group configuration on the Amazon OpenSearch Service domain you specify in the `DomainName` parameter.

## EventBridge

AWS Systems Manager Automation provides predefined runbooks for Amazon EventBridge. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWS-AddOpsItemDedupStringToEventBridgeRule \(p. 168\)](#)
- [AWS-DisableEventBridgeRule \(p. 169\)](#)

## AWS- AddOpsItemDedupStringToEventBridgeRule

### Description

The AWS-AddOpsItemDedupStringToEventBridgeRule runbook adds a deduplication string for all AWS Systems Manager OpsItems associated with an Amazon EventBridge rule. The runbook doesn't add a deduplication string to the rule if one has already been applied. To learn more deduplication strings and OpsItems, see [Reducing duplicate OpsItems](#) in the *AWS Systems Manager User Guide* .

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- DedupString

Type: String

Description: (Required) The deduplication string you want to add to the rule.

- RuleName

Type: String

Description: (Required) The name of the rule you want to add the deduplication string to.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- events:ListTargetsByRule
- events:PutTargets

### Document Steps

- `aws:executeScript` - Adds a deduplication string to the EventBridge rule you specify in the `RuleName` parameter.

## AWS-DisableEventBridgeRule

### Description

The `AWS-DisableEventBridgeRule` runbook disables the Amazon EventBridge rule you specify. To learn more about EventBridge rules, see [Amazon EventBridge rules](#) in the *Amazon EventBridge User Guide*.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `EventBusName`

Type: String

Default: default

Description: (Optional) The event bus associated with the rule you want to disable.

- `RuleName`

Type: String

Description: (Required) The name of the rule you want to disable.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `events:DisableRule`

### Document Steps



- `aws:executeAwsApi` - Disables the EventBridge rule you specify in the `RuleName` parameter.

## GuardDuty

AWS Systems Manager Automation provides predefined runbooks for Amazon GuardDuty. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-CreateGuardDutyDetector \(p. 170\)](#)

## AWSConfigRemediation-CreateGuardDutyDetector

### Description

The `AWSConfigRemediation-CreateGuardDutyDetector` runbook creates an Amazon GuardDuty (GuardDuty) detector in the AWS Region where you run the automation.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `guardduty:CreateDetector`
- `guardduty:GetDetector`

### Document Steps

- `aws:executeAwsApi` - Creates a GuardDuty detector.
- `aws:assertAwsResourceProperty` - Verifies the Status of the detector is `ENABLED` .

## IAM

AWS Systems Manager Automation provides predefined runbooks for AWS Identity and Access Management. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWS-AttachIAMToInstance \(p. 171\)](#)
- [AWSConfigRemediation-DeleteIAMRole \(p. 173\)](#)
- [AWSConfigRemediation-DeleteIAMUser \(p. 174\)](#)
- [AWSConfigRemediation-DeleteUnusedIAMGroup \(p. 175\)](#)
- [AWSConfigRemediation-DeleteUnusedIAMPolicy \(p. 176\)](#)
- [AWSConfigRemediation-DetachIAMPolicy \(p. 177\)](#)
- [AWSConfigRemediation-EnableAccountAccessAnalyzer \(p. 178\)](#)
- [AWSSupport-GrantPermissionsToIAMUser \(p. 179\)](#)
- [AWSConfigRemediation-RemoveUserPolicies \(p. 183\)](#)
- [AWSConfigRemediation-ReplaceIAMInlinePolicy \(p. 184\)](#)
- [AWSConfigRemediation-RevokeUnusedIAMUserCredentials \(p. 185\)](#)
- [AWSConfigRemediation-SetIAMPasswordPolicy \(p. 187\)](#)

## AWS-AttachIAMToInstance

### Description

Attach an AWS Identity and Access Management (IAM) role to a managed instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- ForceReplace  
Type: Boolean  
Description: (Optional) Flag to specify whether to replace the existing IAM profile or not.  
Default: true
- InstanceId  
Type: String  
Description: (Required) The ID of the instance on which you want to assign an IAM role.
- RoleName  
Type: String  
Description: (Required) The IAM role name to add to the managed instance.

### Document Steps

1. `aws:executeAwsApi - DescribeInstanceProfile` - Find the IAM instance profile attached to the EC2 instance.
2. `aws:branch - CheckInstanceProfileAssociations` - Check the IAM instance profile attached to the EC2 instance.
  - a. If an IAM instance profile is attached and `ForceReplace` is set to `true` :
    - i. `aws:executeAwsApi - DisassociateIAMInstanceProfile` - Disassociate the IAM instance profile from the EC2 instance.
    - b. `aws:executeAwsApi - ListInstanceProfilesForRole` - List instance profiles for the IAM role provided.
    - c. `aws:branch - CheckInstanceProfileCreated` - Check if the IAM role provided has an associated instance profile.
      - i. If the IAM role has an associated instance profile:
        - A. `aws:executeAwsApi - AttachIAMProfileToInstance` - Attach the IAM instance profile role to the EC2 instance.
      - i. If the IAM role does not have an associated instance profile:
        - A. `aws:executeAwsApi - CreateInstanceProfileForRole` - Create an instance profile role for the specified IAM role.
        - B. `aws:executeAwsApi - AddRoleToInstanceProfile` - Attach the instance profile role to the specified IAM role.
        - C. `aws:executeAwsApi - GetInstanceProfile` - Get the instance profile data for the specified IAM role.
        - D. `aws:executeAwsApi - AttachIAMProfileToInstanceWithRetry` - Attach the IAM instance profile role to the EC2 instance.

### Outputs

`AttachIAMProfileToInstanceWithRetry.AssociationId`

`GetInstanceProfile.InstanceProfileName`

`GetInstanceProfile.InstanceProfileArn`

`AttachIAMProfileToInstance.AssociationId`

`ListInstanceProfilesForRole.InstanceProfileName`

ListInstanceProfilesForRole.InstanceProfileArn

## AWSConfigRemediation-DeleteIAMRole

### Description

The `AWSConfigRemediation-DeleteIAMRole` runbook deletes the AWS Identity and Access Management (IAM) role you specify. This automation does not delete instance profiles associated with the IAM role, or service-linked roles.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- IAMRoleID

Type: String

Description: (Required) The ID of the IAM role you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `iam>DeleteRole`
- `iam>DeleteRolePolicy`
- `iam:GetRole`
- `iam>ListAttachedRolePolicies`
- `iam>ListInstanceProfilesForRole`
- `iam>ListRolePolicies`
- `iam>ListRoles`
- `iam:RemoveRoleFromInstanceProfile`

### Document Steps

- `aws:executeScript` - Gathers the name of the IAM role you specify in the `IAMRoleID` parameter.
- `aws:executeScript` - Gathers policies and instance profiles associated with the IAM role.
- `aws:executeScript` - Deletes attached policies.
- `aws:executeScript` - Deletes the IAM role and verifies the role has been deleted.

## AWSConfigRemediation-DeleteIAMUser

### Description

The `AWSConfigRemediation-DeleteIAMUser` runbook deletes the AWS Identity and Access Management (IAM) user you specify. This automation deletes or detaches the following resources associated with the IAM user:

- Access keys
- Attached managed policies
- Git credentials
- IAM group memberships
- IAM user password
- Inline policies
- Multi-factor authentication (MFA) devices
- Signing certificates
- SSH public keys

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `IAMUserId`

Type: String

Description: (Required) The ID of the IAM user you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `iam:DeactivateMFADevice`
- `iam>DeleteAccessKey`
- `iam>DeleteLoginProfile`
- `iam>DeleteServiceSpecificCredential`
- `iam>DeleteSigningCertificate`
- `iam>DeleteSSHPublicKey`
- `iam>DeleteVirtualMFADevice`
- `iam>DeleteUser`
- `iam>DeleteUserPolicy`
- `iam:DetachUserPolicy`
- `iam:GetUser`
- `iam>ListAttachedUserPolicies`
- `iam>ListAccessKeys`
- `iam>ListGroupsForUser`
- `iam>ListMFADevices`
- `iam>ListServiceSpecificCredentials`
- `iam>ListSigningCertificates`
- `iam>ListSSHPublicKeys`
- `iam>ListUserPolicies`
- `iam>ListUsers`
- `iam:RemoveUserFromGroup`

### Document Steps

- `aws:executeScript` - Gathers the user name of the IAM user you specify in the `IAMUserId` parameter.
- `aws:executeScript` - Gathers access keys, certificates, credentials, MFA devices, and SSH keys associated with the IAM user.
- `aws:executeScript` - Gathers group memberships and policies for the IAM user.
- `aws:executeScript` - Deletes access keys, certificates, credentials, MFA devices, and SSH keys associated with the IAM user.
- `aws:executeScript` - Deletes group memberships and policies for the IAM user.
- `aws:executeScript` - Deletes the IAM user and verifies the user has been deleted.

## AWSConfigRemediation-DeleteUnusedIAMGroup

### Description

The AWSConfigRemediation-DeleteUnusedIAMGroup runbook deletes an IAM group that does not contain any IAM users.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([\w+=,./-]+)$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- GroupName

Type: String

Description: (Required) The name of the IAM group that you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `iam>DeleteGroup`
- `iam>DeleteGroupPolicy`
- `iam:DetachGroupPolicy`

### Document Steps

- `aws:executeScript` - Removes managed and inline IAM policies attached to the target IAM group, and then deletes the IAM group.

## AWSConfigRemediation-DeleteUnusedIAMPolicy

### Description

The AWSConfigRemediation-DeleteUnusedIAMPolicy runbook deletes an AWS Identity and Access Management (IAM) policy that is not attached to any IAM users, groups, or roles.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- IAMResourceId

Type: String

Description: (Required) The resource identifier of the IAM policy that you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `config>ListDiscoveredResources`
- `iam>DeletePolicy`
- `iam>DeletePolicyVersion`
- `iam:GetPolicy`
- `iam>ListEntitiesForPolicy`
- `iam>ListPolicyVersions`

### Document Steps

- `aws:executeScript` - Deletes the policy you specify in the `IAMResourceId` parameter, and verifies the policy was deleted.

## AWSConfigRemediation-DetachIAMPolicy

### Description

The `AWSConfigRemediation-DetachIAMPolicy` runbook detaches the AWS Identity and Access Management (IAM) policy you specify.



[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- IAMResourceId

Type: String

Description: (Required) The ID of the IAM policy you want to detach.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `config>ListDiscoveredResources`
- `iam:DetachGroupPolicy`
- `iam:DetachRolePolicy`
- `iam:DetachUserPolicy`
- `iam:GetPolicy`
- `iam>ListEntitiesForPolicy`

### Document Steps

- `aws:executeScript` - Detaches the IAM policy from all resources.

## AWSConfigRemediation- EnableAccountAccessAnalyzer

### Description

The `AWSConfigRemediation-EnableAccountAccessAnalyzer` runbook creates an AWS Identity and Access Management (IAM) Access Analyzer in your AWS account. For information about Access Analyzer, see [Using AWS IAM Access Analyzer](#) in the *IAM User Guide* .

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AnalyzerName

Type: String

Description: (Required) The name of the analyzer to create.

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `access-analyzer:CreateAnalyzer`
- `access-analyzer:GetAnalyzer`

### Document Steps

- `aws:executeAwsApi` - Creates an access analyzer for your account.
- `aws:waitForAwsResourceProperty` - Waits for the status of the access analyzer to be `ACTIVE` .
- `aws:assertAwsResourceProperty` - Confirms the status of the access analyzer is `ACTIVE` .

## AWSSupport-GrantPermissionsToIAMUser

### Description

This runbook grants the specified permissions to an IAM group (new or existing), and adds the existing IAM user to it. Policies you can choose from: [Billing](#) or [Support](#) . To enable billing access for IAM,

remember to also activate [IAM user and federated user access to the Billing and Cost Management pages](#)

**Important**

If you provide an existing IAM group, all current IAM users in the group receive the new permissions.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- IAMGroupName

Type: String

Default: ExampleSupportAndBillingGroup

Description: (Required) Can be a new or existing group. Must comply with [IAM Entity Name Limits](#).

- IAMUserName

Type: String

Default: ExampleUser

Description: (Required) Must be an existing user.

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role assumed by lambda.

- Permissions

Type: String

Valid values: SupportFullAccess | BillingFullAccess | SupportAndBillingFullAccess

Default: SupportAndBillingFullAccess

Description: (Required) Choose one of: `SupportFullAccess` grants full access to the Support center. `BillingFullAccess` grants full access to the Billing dashboard. `SupportAndBillingFullAccess`

grants full access to both Support center and the Billing dashboard. More info on policies under Document details.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

The permissions required depend on how `AWSSupport-GrantPermissionsToIAMUser` is run.

### Running as the currently logged in user or role

It is recommended you have the `AmazonSSMAutomationRole` Amazon managed policy attached, and the following additional permissions to be able to create the Lambda function and the IAM role to pass to Lambda:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "lambda:InvokeFunction",
        "lambda:CreateFunction",
        "lambda>DeleteFunction",
        "lambda:GetFunction"
      ],
      "Resource": "arn:aws:lambda:*:ACCOUNTID:function:AWSSupport-*",
      "Effect": "Allow"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateGroup",
        "iam:AddUserToGroup",
        "iam:ListAttachedGroupPolicies",
        "iam:GetGroup",
        "iam:GetUser"
      ],
      "Resource": [
        "arn:aws:iam:*:user/*",
        "arn:aws:iam:*:group/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:AttachGroupPolicy"
      ],
      "Resource": "*",
      "Condition": {
        "ArnEquals": {
          "iam:PolicyArn": [
            "arn:aws:iam::aws:policy/job-function/Billing",
            "arn:aws:iam::aws:policy/AWSSupportAccess"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListAccountAliases",
        "iam:GetAccountSummary"
      ]
    }
  ]
}
```

```
        "Resource" : "*"
      }
    ]
  }
}
```

### Using AutomationAssumeRole and LambdaAssumeRole

The user must have the `ssm:StartAutomationExecution` permissions on the runbook, and `iam:PassRole` on the IAM roles passed as `AutomationAssumeRole` and `LambdaAssumeRole`. Here are the permissions each IAM role needs:

#### AutomationAssumeRole

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "lambda:InvokeFunction",
        "lambda:CreateFunction",
        "lambda>DeleteFunction",
        "lambda:GetFunction"
      ],
      "Resource":
"arn:aws:lambda:*:ACCOUNTID:function:AWSSupport-*",
      "Effect": "Allow"
    }
  ]
}
```

#### LambdaAssumeRole

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect" : "Allow",
      "Action" : [
        "iam:CreateGroup",
        "iam:AddUserToGroup",
        "iam:ListAttachedGroupPolicies",
        "iam:GetGroup",
        "iam:GetUser"
      ],
      "Resource" : [
        "arn:aws:iam::*:user/*",
        "arn:aws:iam::*:group/*"
      ]
    },
    {
      "Effect" : "Allow",
      "Action" : [
        "iam:AttachGroupPolicy"
      ],
      "Resource": "*",
      "Condition": {
        "ArnEquals": {
          "iam:PolicyArn": [
            "arn:aws:iam::aws:policy/job-function/Billing",
            "arn:aws:iam::aws:policy/AWSSupportAccess"
          ]
        }
      }
    }
  ]
}
```

```
    }
  },
  {
    "Effect" : "Allow",
    "Action" : [
      "iam:ListAccountAliases",
      "iam:GetAccountSummary"
    ],
    "Resource" : "*"
  }
]
```

### Document Steps

1. `aws:createStack` - Run AWS CloudFormation Template to create a Lambda function.
2. `aws:invokeLambdaFunction` - Run Lambda to set IAM permissions.
3. `aws:deleteStack` - Delete CloudFormation Template.

### Outputs

`configureIAM.Payload`

## AWSConfigRemediation-RemoveUserPolicies

### Description

The `AWSConfigRemediation-RemoveUserPolicies` runbook deletes the AWS Identity and Access Management (IAM) inline policies and detaches any managed policies attached to the IAM user you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `IAMUserID`

Type: String

Description: (Required) The ID of the IAM user you want to remove policies from.

- PolicyType

Type: String

Valid values: All | Inline | Managed

Default: All

Description: (Required) The type of IAM policies you want to remove from the IAM user.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `iam>DeleteUserPolicy`
- `iam:DetachUserPolicy`
- `iam>ListAttachedUserPolicies`
- `iam>ListUserPolicies`
- `iam>ListUsers`

### Document Steps

- `aws:executeScript` - Deletes and detaches IAM policies from the IAM user you specify in the IAMUserID parameter.

## AWSConfigRemediation- ReplaceIAMInlinePolicy

### Description

The AWSConfigRemediation-ReplaceIAMInlinePolicy runbook replaces an inline AWS Identity and Access Management (IAM) policy with a replicated managed IAM policy. For an inline policy attached to an IAM user, group, or role, the inline policy permissions are cloned into a managed IAM policy. The managed IAM policy is added to the resource, and the inline policy is removed. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=,./-]+\]\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- InlinePolicyName

Type: StringList

Description: (Required) The inline IAM policy you want to replace.

- ResourceId

Type: String

Description: (Required) The ID of the IAM user, group, or role whose inline policy you want to replace.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- iam:AttachGroupPolicy
- iam:AttachRolePolicy
- iam:AttachUserPolicy
- iam:CreatePolicy
- iam:CreatePolicyVersion
- iam>DeleteGroupPolicy
- iam>DeleteRolePolicy
- iam>DeleteUserPolicy
- iam:GetGroupPolicy
- iam:GetRolePolicy
- iam:GetUserPolicy
- iam:ListGroupPolicies
- iam:ListRolePolicies
- iam:ListUserPolicies

### Document Steps

- aws:executeScript - Replace the inline IAM policy with an AWS replicated policy on the resource that you specify.

## AWSConfigRemediation- RevokeUnusedIAMUserCredentials

### Description



The `AWSConfigRemediation-RevokeUnusedIAMUserCredentials` runbook revokes unused AWS Identity and Access Management (IAM) passwords and active access keys. This runbook also deactivates expired access keys, and deletes expired login profiles. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([w+=,.-]+)$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- IAMResourceId

Type: String

Description: (Required) The ID of the IAM resource you want to revoke unused credentials from.

- MaxCredentialUsageAge

Type: String

Default: 90

Description: (Required) The number of days within which the credential must have been used.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config>ListDiscoveredResources`
- `iam>DeleteAccessKey`
- `iam>DeleteLoginProfile`
- `iam:GetAccessKeyLastUsed`
- `iam:GetLoginProfile`
- `iam:GetUser`
- `iam>ListAccessKeys`

- iam:UpdateAccessKey

### Document Steps

- aws:executeScript - Revokes IAM credentials for the user specified in the IAMResourceId parameter. Expired access keys are deactivated, and expired login profiles are deleted.

## AWSConfigRemediation-SetIAMPasswordPolicy

### Description

The AWSConfigRemediation-SetIAMPasswordPolicy runbook sets the AWS Identity and Access Management (IAM) user password policy for your AWS account.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=,./-]+\]\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- AllowUsersToChangePassword

Type: Boolean

Default: false

Description: (Optional) If set to true, all IAM users in your AWS account can use the AWS Management Console to change their passwords.

- HardExpiry

Type: Boolean

Default: false

Description: (Optional) If set to true, IAM users are prevented from resetting their passwords after their password expires.

- MaxPasswordAge

Type: Integer

Default: 0

Description: (Optional) The number of days an IAM user's password is valid.

- `MinimumPasswordLength`

Type: Integer

Default: 6

Description: (Optional) The minimum number of characters an IAM user's password can be.

- `PasswordReusePrevention`

Type: Integer

Default: 0

Description: (Optional) The number of previous passwords that an IAM user is prevented from reusing.

- `RequireLowercaseCharacters`

Type: Boolean

Default: false

Description: (Optional) If set to `true`, an IAM user's password must contain a lowercase character from the ISO basic Latin alphabet (a to z).

- `RequireNumbers`

Type: Boolean

Default: false

Description: (Optional) If set to `true`, an IAM user's password must contain a numeric character (0-9).

- `RequireSymbols`

Type: Boolean

Default: false

Description: (Optional) If set to `true`, an IAM user's password must contain a non-alphanumeric character (!@#\$%^\*()\_+ - = [ ] { } | ').

- `RequireUppercaseCharacters`

Type: Boolean

Default: false

Description: (Optional) If set to `true`, an IAM user's password must contain an uppercase character from the ISO basic Latin alphabet (A to Z).

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

- `iam:GetAccountPasswordPolicy`
- `iam:UpdateAccountPasswordPolicy`

### Document Steps

- `aws:executeScript` - Sets the IAM user password policy based on the values you specify for the runbook parameters for your AWS account.

## AWS KMS

AWS Systems Manager Automation provides predefined runbooks for AWS Key Management Service. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-CancelKeyDeletion \(p. 189\)](#)
- [AWSConfigRemediation-EnableKeyRotation \(p. 190\)](#)

## AWSConfigRemediation-CancelKeyDeletion

### Description

The `AWSConfigRemediation-CancelKeyDeletion` runbook cancels deletion of the AWS Key Management Service (AWS KMS) customer managed key that you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\d{4}`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `KeyId`

Type: String

Description: (Required) The ID of the customer managed key that you want to cancel deletion for.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `kms:CancelKeyDeletion`
- `kms:DescribeKey`

### Document Steps

- `aws:executeAwsApi` - Cancels deletion for the customer managed key you specify in the `KeyId` parameter.
- `aws:assertAwsResourceProperty` - Confirms key deletion is disabled on your customer managed key.

## AWSConfigRemediation-EnableKeyRotation

### Description

The AWSConfigRemediation-EnableKeyRotation runbook enables automatic key rotation for the symmetric AWS Key Management Service (AWS KMS) customer managed key.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- KeyId

Type: String

Description: (Required) The ID of the customer managed key you want to enable automatic key rotation on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `kms:EnableKeyRotation`
- `kms:GetKeyRotationStatus`

#### Document Steps

- `aws:executeAwsApi` - Enables automatic key rotation on the customer managed key you specify in the `KeyId` parameter.
- `aws:assertAwsResourceProperty` - Confirms that automatic key rotation is enabled on your customer managed key.

## Lambda

AWS Systems Manager Automation provides predefined runbooks for AWS Lambda. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

#### Topics

- [AWSConfigRemediation-ConfigureLambdaFunctionXRayTracing \(p. 191\)](#)
- [AWSConfigRemediation-DeleteLambdaFunction \(p. 192\)](#)
- [AWSConfigRemediation-EncryptLambdaEnvironmentVariablesWithCMK \(p. 193\)](#)
- [AWSConfigRemediation-MoveLambdaToVPC \(p. 194\)](#)
- [AWSSupport-RemediateLambdaS3Event \(p. 196\)](#)
- [AWSSupport-TroubleshootLambdaInternetAccess \(p. 197\)](#)
- [AWSSupport-TroubleshootLambdaS3Event \(p. 200\)](#)

## AWSConfigRemediation- ConfigureLambdaFunctionXRayTracing

#### Description

The `AWSConfigRemediation-ConfigureLambdaFunctionXRayTracing` runbook enables AWS X-Ray live tracing on the AWS Lambda function you specify in the `FunctionName` parameter.

[Run this Automation \(console\)](#)

#### Document type

Automation

#### Owner

Amazon

#### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- FunctionName

Type: String

Description: (Required) The name or ARN of the Lambda function to enable tracing on.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- lambda:UpdateFunctionConfiguration
- ssm:StartAutomationExecution
- ssm:GetAutomationExecution

### Document Steps

- aws:executeAwsApi - Enables X-Ray tracing on the Lambda function you specify in the FunctionName parameter.
- aws:assertAwsResourceProperty - Verifies that X-Ray tracing has been enabled on the Lambda function.

### Outputs

UpdateLambdaConfig.UpdateFunctionConfigurationResponse - Response from the UpdateFunctionConfiguration API call.

## AWSConfigRemediation-DeleteLambdaFunction

### Description

The AWSConfigRemediation-DeleteLambdaFunction runbook deletes the AWS Lambda function you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- LambdaFunctionName

Type: String

Description: (Required) The name of the Lambda function that you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- lambda:DeleteFunction
- lambda:GetFunction

### Document Steps

- aws:executeAwsApi - Deletes the Lambda function specified in the LambdaFunctionName parameter.
- aws:executeScript - Verifies the Lambda function has been deleted.

## AWSConfigRemediation- EncryptLambdaEnvironmentVariablesWithCMK

### Description

The AWSConfigRemediation-EncryptLambdaEnvironmentVariablesWithCMK runbook encrypts, at rest, the environment variables for the AWS Lambda (Lambda) function you specify using an AWS Key Management Service (AWS KMS) customer managed key. This runbook should only be used as a baseline to ensure that your Lambda function's environment variables are encrypted according to minimum recommended security best practices. We recommend encrypting multiple functions with different customer managed keys.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon



## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=,./-]+\]\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- FunctionName

Type: String

Description: (Required) The name or ARN of the Lambda function whose environment variables you want to encrypt.

- KMSKeyArn

Type: String

Description: (Required) The ARN of the AWS KMS customer managed key you want to use to encrypt your Lambda function's environment variables.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- lambda:GetFunctionConfiguration
- lambda:UpdateFunctionConfiguration

## Document Steps

- aws:waitForAwsResourceProperty - Waits for the LastUpdateStatus property to be Successful.
- aws:executeAwsApi - Encrypts the environment variables for the Lambda function you specify in the FunctionName parameter using the AWS KMS customer managed key you specify in the KMSKeyArn parameter.
- aws:assertAwsResourceProperty - Confirms encryption is enabled on the environment variables for your Lambda function.

# AWSConfigRemediation-MoveLambdaToVPC

## Description

The AWSConfigRemediation-MoveLambdaToVPC runbook moves an AWS Lambda (Lambda) function to an Amazon Virtual Private Cloud (Amazon VPC).

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- FunctionName

Type: String

Description: (Required) The name of the Lambda function to move to an Amazon VPC.

- SecurityGroupIds

Type: String

Description: (Required) The security group IDs you want to assign to the elastic network interfaces (ENIs) associated with your Lambda function.

- SubnetIds

Type: String

Description: (Required) The subnet IDs you want to create the elastic network interfaces (ENIs) associated with your Lambda function.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `lambda:GetFunction`
- `lambda:GetFunctionConfiguration`
- `lambda:UpdateFunctionConfiguration`

## Document Steps

- `aws:executeAwsApi` - Updates the Amazon VPC configuration for the Lambda function you specify in the `FunctionName` parameter.
- `aws:waitForAwsResourceProperty` - Waits for the Lambda function `LastUpdateStatus` to be successful.

- `aws:executeScript` - Verifies the Lambda function Amazon VPC configuration has been successfully updated.

## AWSSupport-RemediateLambdaS3Event

### Description

The `AWSSupport-TroubleshootLambdaS3Event` runbook provides an automated solution for the procedures outlined in the AWS Knowledge Center articles [Why doesn't my Amazon S3 event notification trigger my Lambda function?](#) and [Why do I get the error "Unable to validate the following destination configurations" when creating an Amazon S3 event notification to trigger my Lambda function?](#) This runbook helps you identify and remediate why an Amazon Simple Storage Service (Amazon S3) event notification failed to trigger the AWS Lambda function you specified. If the runbook output suggests validating and configuring your Lambda function concurrency, see [Asynchronous invocation](#) and [AWS Lambda Function scaling](#) .

#### Note

"Unable to validate the following destination configurations" errors can also occur due to incorrect Amazon Simple Notification Service (Amazon SNS) and Amazon Simple Queue Service (Amazon SQS) Amazon S3 event configurations. This runbook only checks Lambda function configurations. If after using the runbook, you are still receiving the "Unable to validate the following destination configurations" error, please review any existing Amazon SNS and Amazon SQS Amazon S3 event configurations.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `LambdaFunctionArn`

Type: String

Description: (Required) The ARN of the Lambda function.

- `S3BucketName`

Type: String

Description: (Required) The name of the Amazon S3 bucket whose event notifications triggers the Lambda function.

- Action

Type: String

Valid values: Troubleshoot | Remediate

Description: (Required) The action you want the runbook to perform. The `Troubleshoot` option helps identify any issues, but does not perform any mutating actions to resolve the issue. The `Remediate` option helps identify and attempts to resolve issues for you.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetDocument`
- `ssm:ListDocuments`
- `ssm:DescribeAutomationExecutions`
- `ssm:DescribeAutomationStepExecutions`
- `ssm:GetAutomationExecution`
- `lambda:GetPolicy`
- `lambda:AddPermission`
- `s3:GetBucketNotification`

### Document Steps

- `aws:branch` - Branches based on the input specified for the `Action` parameter.

If the value specified is `Troubleshoot` :

- `aws:executeAutomation` - Runs the `AWSSupport-TroubleshootLambdaS3Event` runbook.
- `aws:executeAwsApi` - Checks the output of the `AWSSupport-TroubleshootLambdaS3Event` runbook that ran in the previous step.

If the value specified is `Remediate` :

- `aws:executeScript` - Runs a script to remediate the issues outlined in the [Why doesn't my Amazon S3 event notification trigger my Lambda function?](#) and [Why do I get the error "Unable to validate the following destination configurations" when creating an Amazon S3 event notification to trigger my Lambda function?](#) Knowledge Center articles.

### Outputs

`checkoutput.Output`

`remediatelambdas3event.Output`

## AWSSupport-TroubleshootLambdaInternetAccess

### Description

The `AWSSupport-TroubleshootLambdaInternetAccess` runbook helps you troubleshoot internet access issues for a AWS Lambda function that was launched into Amazon Virtual Private Cloud (Amazon

VPC). Resources such as subnet routes, security groups rules, and network access control list (ACL) rules are reviewed to confirm outbound internet access is allowed.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- FunctionName

Type: String

Description: (Required) The name of the Lambda function you want to troubleshoot internet access for.

- destinationIp

Type: String

Description: (Required) The destination IP address you want to establish an outbound connection to.

- destinationPort

Type: String

Default: 443

Description: (Optional) The destination port you want to establish an outbound connection on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `lambda:GetFunction`
- `ec2:DescribeRouteTables`
- `ec2:DescribeNatGateways`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeNetworkAcls`

### Document Steps

- `aws:executeScript` - Verifies the configuration of various resources in your VPC where the Lambda function was launched.
- `aws:branch` - Branches based on whether the Lambda function specified is in a VPC or not.
- `aws:executeScript` - Reviews the route table routes for the subnet where the Lambda function was launched, and verifies that routes to a network address translation (NAT) gateway, and internet gateway are present. Confirms the Lambda function is not in a public subnet.
- `aws:executeScript` - Verifies the security group associated with the Lambda function allows outbound internet access based on the values specified for the `destinationIp` and `destinationPort` parameters.
- `aws:executeScript` - Verifies the ACL rules associated with the subnets of the Lambda function and the NAT gateway allow outbound internet access based on the values specified for the `destinationIp` and `destinationPort` parameters.

## Outputs

`checkVpc.vpc` - The ID of the VPC where your Lambda function was launched.

`checkVpc.subnet` - The IDs of the subnets where your Lambda function was launched.

`checkVpc.securityGroups` - Security groups associated with the Lambda function.

`checkNACL.NACL` - Analysis message with resource names. `LambdaIp` refers to the private IP address of the elastic network interface for your Lambda function. The `LambdaIpRules` object is only generated for subnets that have a route to a NAT gateway. The following content is an example of the output.

```
{
  "subnet-1234567890":{
    "NACL":"acl-1234567890",
    "destinationIp_Egress":"Allowed",
    "destinationIp_Ingress":"notAllowed",
    "Analysis":"This NACL has an allow rule for Egress traffic but there is no Ingress rule. Please allow the destination IP / destination port in Ingress rule",
    "LambdaIpRules":{
      "{LambdaIp}":{
        "Egress":"notAllowed",
        "Ingress":"notAllowed",
        "Analysis":"This is a NAT subnet NACL. It does not have ingress or egress rule allowed in it for Lambda's corresponding private ip {LambdaIp} Please allow this IP in your egress and ingress NACL rules"
      }
    }
  },
  "subnet-0987654321":{
    "NACL":"acl-0987654321",
    "destinationIp_Egress":"Allowed",
    "destinationIp_Ingress":"notAllowed",
    "Analysis":"This NACL has an allow rule for Egress traffic but there is no Ingress rule. Please allow the destination IP / destination port in Ingress rule"
  }
}
```

`checkSecurityGroups.secgrps` - Analysis for the security group associated with your Lambda function. The following content is an example of the output.

```
{
  "sg-123456789":{
    "Status":"Allowed",
    "Analysis":"This security group has allowed destination IP and port in its outbound rule."
  }
}
```

```
}  
}
```

checkSubnet.subnets - Analysis for the subnets in your VPC associated with your Lambda function. The following content is an example of the output.

```
{  
  "subnet-0c4ee6cdexample15":{  
    "Route":{  
      "DestinationCidrBlock":"8.8.8.0/26",  
      "NatGatewayId":"nat-00f0example69fdec",  
      "Origin":"CreateRoute",  
      "State":"active"  
    },  
    "Analysis":"This Route Table has an active NAT gateway path. Also, The NAT gateway is  
    launched in public subnet",  
    "RouteTable":"rtb-0b1fexample16961b"  
  }  
}
```

## AWSSupport-TroubleshootLambdaS3Event

### Description

The `AWSSupport-TroubleshootLambdaS3Event` runbook provides an automated solution for the procedures outlined in the AWS Knowledge Center articles [Why doesn't my Amazon S3 event notification trigger my Lambda function?](#) and [Why do I get the error "Unable to validate the following destination configurations" when creating an Amazon S3 event notification to trigger my Lambda function?](#) This runbook helps you identify why an Amazon Simple Storage Service (Amazon S3) event notification failed to trigger the AWS Lambda function you specified. If the runbook output suggests validating and configuring your Lambda function concurrency, see [Asynchronous invocation](#) and [AWS Lambda Function scaling](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- LambdaFunctionArn

Type: String

Description: (Required) The ARN of the Lambda function that the Amazon S3 event notification triggers.

- `S3BucketName`

Type: String

Description: (Required) The name of the Amazon S3 bucket whose event notifications triggers the Lambda function.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `lambda:GetPolicy`
- `s3:GetBucketNotification`

### Document Steps

- `aws:executeScript` - Runs the script to validate configuration settings for the Amazon S3 event notification. Validates the resource-based IAM policy for your Lambda function, and generates an AWS Command Line Interface (AWS CLI) command to add the needed permissions if the required permissions are missing from the policy. Validates other Lambda functions resource policies which are part of event notifications for the same S3 bucket and generates an AWS CLI command as output if the required permissions are missing.

### Outputs

`lambdaS3Event.output`

## Amazon RDS

AWS Systems Manager Automation provides predefined runbooks for Amazon Relational Database Service. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWS-CreateRdsSnapshot \(p. 202\)](#)
- [AWSConfigRemediation-DeleteRDSCluster \(p. 203\)](#)
- [AWSConfigRemediation-DeleteRDSClusterSnapshot \(p. 204\)](#)
- [AWSConfigRemediation-DeleteRDSInstance \(p. 205\)](#)
- [AWSConfigRemediation-DeleteRDSInstanceSnapshot \(p. 206\)](#)
- [AWSConfigRemediation-DisablePublicAccessToRDSInstance \(p. 207\)](#)
- [AWSConfigRemediation-EnableCopyTagsToSnapshotOnRDSCluster \(p. 208\)](#)
- [AWSConfigRemediation-EnableCopyTagsToSnapshotOnRDSDBInstance \(p. 209\)](#)
- [AWSConfigRemediation-EnableEnhancedMonitoringOnRDSInstance \(p. 211\)](#)
- [AWSConfigRemediation-EnableMinorVersionUpgradeOnRDS \(p. 212\)](#)
- [AWSConfigRemediation-EnableMultiAZOnRDSInstance \(p. 213\)](#)
- [AWSConfigRemediation-EnablePerformanceInsightsOnRDSInstance \(p. 214\)](#)
- [AWSConfigRemediation-EnableRDSClusterDeletionProtection \(p. 216\)](#)



- [AWSConfigRemediation-EnableRDSInstanceBackup](#) (p. 217)
- [AWSConfigRemediation-EnableRDSInstanceDeletionProtection](#) (p. 218)
- [AWSConfigRemediation-ModifyRDSInstancePortNumber](#) (p. 220)
- [AWSSupport-ModifyRDSSnapshotPermission](#) (p. 221)
- [AWSPremiumSupport-PostgreSQLWorkloadReview](#) (p. 222)
- [AWS-RebootRdsInstance](#) (p. 233)
- [AWSSupport-ShareRDSSnapshot](#) (p. 234)
- [AWS-StartRdsInstance](#) (p. 236)
- [AWSSupport-TroubleshootConnectivityToRDS](#) (p. 237)

## AWS-CreateRdsSnapshot

### Description

Create an Amazon Relational Database Service (Amazon RDS) snapshot for an Amazon RDS instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- DBInstanceIdentifier

Type: String

Description: (Required) The DBInstanceIdentifier ID of the RDS Instance to create Snapshot from.

- DBSnapshotIdentifier

Type: String

Description: (Optional) The DBSnapshotIdentifier ID of the RDS snapshot to create.

- InstanceTags

Type: String

Description: (Optional) Tags to create for instance.

- SnapshotTags

Type: String

Description: (Optional) Tags to create for snapshot.

### Document Steps

createRDSSnapshot – Creates the RDS snapshot and returns the snapshot ID.

verifyRDSSnapshot – Checks that the snapshot created in the previous step exists.

### Outputs

createRDSSnapshot.SnapshotId – The ID of the created snapshot.

## AWSConfigRemediation-DeleteRDSCluster

### Description

The `AWSConfigRemediation-DeleteRDSCluster` runbook deletes the Amazon Relational Database Service (Amazon RDS) cluster you specify. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- DBClusterId

Type: String

Description: (Required) The resource identifier for the DB cluster you want to enable deletion protection on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `rds>DeleteDBCluster`
- `rds>DeleteDBInstance`
- `rds:DescribeDBClusters`

### Document Steps

- `aws:executeScript` - Deletes the DB cluster you specify in the `DBClusterId` parameter.

## AWSConfigRemediation-DeleteRDSClusterSnapshot

### Description

The `AWSConfigRemediation-DeleteRDSClusterSnapshot` runbook deletes the given Amazon Relational Database Service (Amazon RDS) cluster snapshot.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DBClusterSnapshotId`

Type: String

Description: (Required) The Amazon RDS cluster snapshot identifier to be deleted.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`

- `ssm:GetAutomationExecution`
- `rds:DeleteDBClusterSnapshot`
- `rds:DescribeDBClusterSnapshots`

### Document Steps

- `aws:branch` - Checks if the cluster snapshot is in the `available` state. If it is not available, the flow ends.
- `aws:executeAwsApi` - Deletes the given Amazon RDS cluster snapshot using the database (DB) cluster snapshot identifier.
- `aws:executeScript` - Verifies that the given Amazon RDS cluster snapshot was deleted.

## AWSConfigRemediation-DeleteRDSInstance

### Description

The `AWSConfigRemediation-DeleteRDSInstance` runbook deletes the Amazon Relational Database Service (Amazon RDS) instance you specify. When you delete a database (DB) instance, all automated backups for that instance are deleted and can't be recovered. Manual DB snapshots are not deleted. If the DB instance you want to delete is in the `failed`, `incompatible-network`, or `incompatible-restore` state, you must set the `SkipFinalSnapshot` parameter to `true`.

#### Note

If the DB instance you want to delete is in an Amazon Aurora DB cluster, the runbook will not delete the DB instance if it is a read replica and the only instance in the DB cluster.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbiResourceId`

Type: String

Description: (Required) The resource identifier for the DB instance you want to delete.

- `SkipFinalSnapshot`

Type: Boolean

Default: false

Description: (Optional) If set to `true`, a final snapshot is not created before the DB instance is deleted.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds>DeleteDBInstance`
- `rds:DescribeDBInstances`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB instance name from the value you specify in the `DbiResourceId` parameter.
- `aws:branch` - Branches based on the value you specify in the `SkipFinalSnapshot` parameter.
- `aws:executeAwsApi` - Deletes the DB instance you specify in the `DbiResourceId` parameter.
- `aws:executeAwsApi` - Deletes the DB instance you specify in the `DbiResourceId` parameter after the final snapshot is created.
- `aws:assertAwsResourceProperty` - Verifies the DB instance was deleted.

## AWSConfigRemediation-DeleteRDSInstanceSnapshot

### Description

The `AWSConfigRemediation-DeleteRDSInstanceSnapshot` runbook deletes the Amazon Relational Database Service (Amazon RDS) instance snapshot you specify. Only snapshots in the `available` state are deleted. This runbook does not support deleting snapshots from Amazon Aurora database instances.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- DbSnapshotId

Type: String

Description: (Required) The ID of the snapshot you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds>DeleteDBSnapshot`
- `rds:DescribeDBSnapshots`

### Document Steps

- `aws:executeAwsApi` - Gathers the state of the snapshot specified in the DbSnapshotId parameter.
- `aws:assertAwsResourceProperty` - Confirms the state of the snapshot is available .
- `aws:executeAwsApi` - Deletes the snapshot specified in the DbSnapshotId parameter.
- `aws:executeScript` - Verifies the snapshot has been deleted.

## AWSConfigRemediation- DisablePublicAccessToRDSInstance

### Description

The AWSConfigRemediation-DisablePublicAccessToRDSInstance runbook disables public accessibility for the Amazon Relational Database Service (Amazon RDS) database (DB) instance that you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbiResourceId`

Type: String

Description: (Required) The resource identifier for the DB instance that you want to disable public accessibility for.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:assertAwsResourceProperty` - Verifies the DB instances is in an `AVAILABLE` state.
- `aws:executeAwsApi` - Disables public accessibility on your DB instance.
- `aws:waitForAwsResourceProperty` - Waits for the DB instance to change to a `MODIFYING` state.
- `aws:waitForAwsResourceProperty` - Waits for the DB instance to change to an `AVAILABLE` state.
- `aws:assertAwsResourceProperty` - Confirms public accessibility is disabled on the DB instance.

## AWSConfigRemediation- EnableCopyTagsToSnapshotOnRDSCluster

### Description

The `AWSConfigRemediation-EnableCopyTagsToSnapshotOnRDSCluster` runbook enables the `CopyTagsToSnapshot` setting on the Amazon Relational Database Service (Amazon RDS) cluster you specify. Enabling this setting copies all tags from the DB cluster to snapshots of the DB cluster. The default is not to copy them. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Databases

## Parameters

- `ApplyImmediately`

Type: Boolean

Default: false

Description: (Optional) If you specify `true` for this parameter, the modifications in this request and any pending modifications are asynchronously applied as soon as possible, regardless of the `PreferredMaintenanceWindow` setting for the DB cluster.

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\\w+=,./-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbClusterResourceId`

Type: String

Description: (Required) The resource identifier for the DB cluster you want to enable the `CopyTagsToSnapshot` setting on.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `rds:DescribeDBClusters`
- `rds:ModifyDBCluster`

## Document Steps

- `aws:executeAwsApi` - Gathers the DB cluster identifier from the DB cluster resource identifier.
- `aws:assertAwsResourceProperty` - Confirms the DB cluster is in an `AVAILABLE` state.
- `aws:executeAwsApi` - Enables the `CopyTagsToSnapshot` setting on your DB cluster.
- `aws:assertAwsResourceProperty` - Confirms the `CopyTagsToSnapshot` setting is enabled on your DB cluster.

# AWSConfigRemediation- EnableCopyTagsToSnapshotOnRDSDBInstance

## Description



The `AWSConfigRemediation-EnableCopyTagsToSnapshotOnRDSDBInstance` runbook enables the `CopyTagsToSnapshot` setting on the Amazon Relational Database Service (Amazon RDS) instance you specify. Enabling this setting copies all tags from the DB instance to snapshots of the DB instance. The default is not to copy them. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `ApplyImmediately`

Type: Boolean

Default: false

Description: (Optional) If you specify `true` for this parameter, the modifications in this request and any pending modifications are asynchronously applied as soon as possible, regardless of the `PreferredMaintenanceWindow` setting for the DB instance.

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]\+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbiResourceId`

Type: String

Description: (Required) The resource identifier for the DB instance you want to enable the `CopyTagsToSnapshot` setting on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:assertAwsResourceProperty` - Confirms the DB instance is in an `AVAILABLE` state.
- `aws:executeAwsApi` - Enables the `CopyTagsToSnapshot` setting on your DB instance.
- `aws:assertAwsResourceProperty` - Confirms the `CopyTagsToSnapshot` setting is enabled on your DB instance.

## AWSConfigRemediation- EnableEnhancedMonitoringOnRDSInstance

### Description

The `AWSConfigRemediation-EnableEnhancedMonitoringOnRDSInstance` runbook enables Enhanced Monitoring on the Amazon RDS database instance you specify. For information on Enhanced Monitoring, see [Enhanced Monitoring](#) in the *Amazon RDS User Guide*.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `MonitoringInterval`

Type: Integer

Valid values: `1 | 5 | 10 | 15 | 30 | 60`

Description: (Required) The interval in seconds when Enhanced Monitoring metrics are collected from the DB instance.

- `MonitoringRoleArn`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the IAM role that allows Amazon RDS to send Enhanced Monitoring metrics to Amazon CloudWatch Logs.

- `ResourceId`

Type: String

Description: (Required) The resource identifier for the DB instance you want to enable Enhanced Monitoring on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:assertAwsResourceProperty` - Confirms the DB Instance is in an `AVAILABLE` state.
- `aws:executeAwsApi` - Enables Enhanced Monitoring on your DB instance.
- `aws:executeScript` - Confirms that Enhanced Monitoring is enabled on your DB instance.

## AWSConfigRemediation- EnableMinorVersionUpgradeOnRDS

### Description

The `AWSConfigRemediation-EnableMinorVersionUpgradeOnRDS` runbook enables the `AutoMinorVersionUpgrade` setting on the Amazon RDS database instance you specify. Enabling this setting means that minor version upgrades are applied automatically to the DB instance during the maintenance window.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+\d{4}`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbiResourceId`

Type: String

Description: (Required) The resource identifier for the DB instance you want to the `AutoMinorVersionUpgrade` setting on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:assertAwsResourceProperty` - Confirms the DB Instance is in an `AVAILABLE` state.
- `aws:executeAwsApi` - Enables the `AutoMinorVersionUpgrade` setting on your DB instance.
- `aws:executeScript` - Confirms that the `AutoMinorVersionUpgrade` setting is enabled on your DB instance.

## AWSConfigRemediation- EnableMultiAZOnRDSInstance

### Description

The `AWSConfigRemediation-EnableMultiAZOnRDSInstance` runbook changes your Amazon Relational Database Service (Amazon RDS) database (DB) instance to a Multi-AZ deployment. Changing this setting doesn't result in an outage. The change is applied during the next maintenance window unless you set the `ApplyImmediately` parameter to `true`.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `ApplyImmediately`

Type: Boolean

Default: false

Description: (Optional) If you specify `true` for this parameter, the modifications in this request and any pending modifications are asynchronously applied as soon as possible, regardless of the `PreferredMaintenanceWindow` setting for the DB instance.

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\/\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbiResourceId`

Type: String

Description: (Required) The AWS Region-unique, immutable identifier for the DB instance to enable the `MultiAZ` setting.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`
- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

### Document Steps

- `aws:executeAwsApi` - Retrieves the DB instance name using the value provided in the `DBInstanceId` parameter.
- `aws:executeAwsApi` - Verifies the `DBInstanceStatus` is available .
- `aws:branch` - Checks whether the `MultiAZ` is already set to `true` on the DB instance you specify in the `DbiResourceId` parameter.
- `aws:executeAwsApi` - Changes the `MultiAZ` setting to `true` on the DB instance you specify in the `DbiResourceId` parameter.
- `aws:assertAwsResourceProperty` - Verifies the `MultiAZ` is set to `true` on the DB instance you specify in the `DbiResourceId` parameter.

## AWSConfigRemediation- EnablePerformanceInsightsOnRDSInstance

### Description

The `AWSConfigRemediation-EnablePerformanceInsightsOnRDSInstance` runbook enables Performance Insights on the Amazon RDS DB instance you specify.

[Run this Automation \(console\)](#)

### Document type

## Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- DbiResourceId

Type: String

Description: (Required) The resource identifier for the DB instance you want to enable Performance Insights on.

- PerformanceInsightsKMSKeyId

Type: String

Default: `alias/aws/rds`

Description: (Optional) The Amazon Resource Name (ARN), key ID, or the key alias of the AWS Key Management Service (AWS KMS) customer managed key you want Performance Insights to use to encrypt all potentially sensitive data. If you enter the key alias for this parameter, prefix the value with `alias/`. If you do not specify a value for this parameter, the AWS managed key is used.

- PerformanceInsightsRetentionPeriod

Type: Integer

Valid values: 7, 731

Default: 7

Description: (Optional) The number of days you want to retain Performance Insights data.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `kms:CreateGrant`
- `kms:DescribeKey`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

## Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:assertAwsResourceProperty` - Confirms the DB instance status is available .
- `aws:executeAwsApi` - Gathers the ARN of the AWS KMS customer managed key specified in the `PerformanceInsightsKMSKeyId` parameter.
- `aws:branch` - Checks whether a value is already assigned to the `PerformanceInsightsKMSKeyId` property of the DB instance.
- `aws:executeAwsApi` - Enables Performance Insights on the DB instance you specify in the `DbiResourceId` parameter.
- `aws:assertAwsResourceProperty` - Confirms the value specified for the `PerformanceInsightsKMSKeyId` parameter was used to enable encryption for Performance Insights on the DB instance.
- `aws:assertAwsResourceProperty` - Confirms Performance Insights is enabled on the DB instance.

# AWSConfigRemediation- EnableRDSClusterDeletionProtection

## Description

The `AWSConfigRemediation-EnableRDSClusterDeletionProtection` runbook enables deletion protection on the Amazon Relational Database Service (Amazon RDS) cluster you specify. AWS Config must be enabled in the AWS Region where you run this automation.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Databases

## Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([a-z0-9]+)\]`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `ClusterId`

Type: String

Description: (Required) The resource identifier for the DB cluster you want to enable deletion protection on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `config:GetResourceConfigHistory`
- `rds:DescribeDBClusters`
- `rds:ModifyDBCluster`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB cluster name from the DB cluster resource identifier.
- `aws:assertAwsResourceProperty` - Verifies the DB cluster status is available .
- `aws:executeAwsApi` - Enables deletion protection on the DB cluster you specify in the `ClusterId` parameter.
- `aws:assertAwsResourceProperty` - Verifies deletion protection has been enabled on the DB cluster.

## AWSConfigRemediation- EnableRDSInstanceBackup

### Description

The `AWSConfigRemediation-EnableRDSInstanceBackup` runbook enables backups for the Amazon Relational Database Service (Amazon RDS) database instance you specify. This runbook does not support enabling backups for Amazon Aurora database instances.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `ApplyImmediately`

Type: Boolean

Default: false

Description: (Optional) If you specify `true` for this parameter, the modifications in this request and any pending modifications are asynchronously applied as soon as possible, regardless of the `PreferredMaintenanceWindow` setting for the DB instance.

- `AutomationAssumeRole`



Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- BackupRetentionPeriod

Type: Integer

Valid values: 1-35

Description: (Required) The number of days that backups are retained.

- DbInstanceIdentifier

Type: String

Description: (Required) The resource identifier for the DB instance you want to enable backups for.

- PreferredBackupWindow

Type: String

Description: (Optional) The daily time range (in UTC) during which backups are created.

Constraints:

- Must be in the format `hh24:mi-hh24:mi`
- Must be in Coordinated Universal Time (UTC)
- Must not conflict with the preferred maintenance window
- Must be at least 30 minutes

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

### Document Steps

- `aws:executeScript` - Gathers the DB instance identifier from the DB instance resource identifier. Enables backups for your DB instance. Confirms backups are enabled on the DB instance.

## AWSConfigRemediation- EnableRDSInstanceDeletionProtection

### Description

The `AWSConfigRemediation-EnableRDSInstanceDeletionProtection` runbook enables deletion protection on the Amazon RDS database instance you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `ApplyImmediately`

Type: Boolean

Default: false

Description: (Optional) If you specify `true` for this parameter, the modifications in this request and any pending modifications are asynchronously applied as soon as possible, regardless of the `PreferredMaintenanceWindow` setting for the DB instance.

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DbInstanceResourceId`

Type: String

Description: (Required) The resource identifier for the DB instance you want to enable deletion protection on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

### Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:executeAwsApi` - Enables deletion protection on your DB instance.
- `aws:assertAwsResourceProperty` - Confirms deletion protection is enabled on the DB instance.

# AWSConfigRemediation- ModifyRDSInstancePortNumber

## Description

The `AWSConfigRemediation-ModifyRDSInstancePortNumber` runbook modifies the port number on which the Amazon Relational Database Service (Amazon RDS) instance accepts connections. Running this automation will restart the database.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Databases

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+\d{12}`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- PortNumber

Type: String

Description: (Optional) The port number you want the DB instance to accept connections on.

- RDSDBInstanceResourceID

Type: String

Description: (Required) The resource identifier for the DB instance whose inbound port number you want to modify.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:ModifyDBInstance`

## Document Steps

- `aws:executeAwsApi` - Gathers the DB instance identifier from the DB instance resource identifier.
- `aws:assertAwsResourceProperty` - Confirms the DB Instance is in an `AVAILABLE` state.
- `aws:executeAwsApi` - Modifies the inbound port number on which your DB instance accepts connections.
- `aws:waitForAwsResourceProperty` - Waits for the DB Instance to be in a `MODIFYING` state.
- `aws:waitForAwsResourceProperty` - Waits for the DB Instance to be in in an `AVAILABLE` state.

## AWSSupport-ModifyRDSSnapshotPermission

### Description

The `AWSSupport-ModifyRDSSnapshotPermission` runbook helps you modify permissions for multiple Amazon Relational Database Service (Amazon RDS) snapshots. Using this runbook, you can make snapshots `Public` or `Private` and share them with other AWS accounts. Snapshots encrypted with a default KMS key can't be shared with other accounts using this runbook.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `AccountIds`

Type: StringList

Default: none

Description: (Optional) The IDs of the accounts you want to share snapshots with. This parameter is required if you enter `No` for the value of the `Private` parameter.

- `AccountPermissionOperation`

Type: String

Valid values: `add` | `remove`

Default: none

Description: (Optional) The type of operation to perform.

- Private

Type: String

Valid values: Yes | No

Description: (Required) Enter No for the value if you want to share snapshots with specific accounts.

- SnapshotIdentifiers

Type: StringList

Description: (Required) The names of the Amazon RDS snapshots whose permission you want to modify.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `rds:DescribeDBSnapshots`
- `rds:ModifyDBSnapshotAttribute`

### Document Steps

1. `aws:executeScript` - Verifies the IDs of the snapshots provided in the `SnapshotIdentifiers` parameter. After verifying the IDs, the script checks for encrypted snapshots and outputs a list if any are found.
2. `aws:branch` - Branches the automation based on the value you enter for the `Private` parameter.
3. `aws:executeScript` - Modifies permissions of the snapshots specified to share it with the accounts specified.
4. `aws:executeScript` - Modifies permissions of the snapshots to change them from `Public` to `Private`.

### Outputs

ValidateSnapshots.EncryptedSnapshots

SharewithOtherAccounts.Result

MakePrivate.Result

MakePrivate.Commands

## AWSPremiumSupport- PostgreSQLWorkloadReview

### Description

The `AWSPremiumSupport-PostgreSQLWorkloadReview` runbook captures multiple snapshots of your Amazon Relational Database Service (Amazon RDS) PostgreSQL database usage statistics. The statistics captured are required for an AWS Support [Proactive Services](#) expert to perform an operational review. The statistics are collected using a set of custom SQL and shell scripts. These scripts are downloaded to a temporary Amazon Elastic Compute Cloud (Amazon EC2) instance in your AWS

account that is created by this runbook. The runbook requires you to provide credentials using an AWS Secrets Manager secret containing a username and password key-value pair. The username must have permissions to query the standard PostgreSQL statistics views and functions.

This runbook automatically creates the following AWS resources in your AWS account using an AWS CloudFormation stack. You can monitor the stack creation using the AWS CloudFormation console.

- A virtual private cloud (VPC) and an Amazon EC2 instance launched in a private subnet of the VPC with optional connectivity to the internet using a NAT gateway.
- An AWS Identity and Access Management (IAM) role that is attached to the temporary Amazon EC2 instance with permissions to retrieve the Secrets Manager secret value. The role also provides permissions to upload files to an Amazon Simple Storage Service (Amazon S3) bucket of your choice, and optionally to an AWS Support case.
- A VPC peering connection to allow connectivity between your DB instance and the temporary Amazon EC2 instance.
- Systems Manager, Secrets Manager, and Amazon S3 VPC endpoints that are attached to the temporary VPC.
- A maintenance window with registered tasks that periodically start and stop the temporary Amazon EC2 instance, run data collection scripts, and upload files to an Amazon S3 bucket. An IAM role is also created for the maintenance window that provides permissions to perform the registered tasks.

When the runbook completes, the AWS CloudFormation stack that is used to create the necessary AWS resources is deleted and the report is uploaded to the Amazon S3 bucket of your choice, and optionally an AWS Support case.

**Note**

By default, the root Amazon EBS volume of the temporary Amazon EC2 instance is preserved. You can override this option by setting the `EbsVolumeDeleteOnTermination` parameter to `true`.

**Prerequisites**

- **Enterprise Support subscription**

This runbook and the Proactive Services Workload Diagnostics and Reviews require an Enterprise Support Subscription. Before using this runbook, contact your Technical Account Manager (TAM) or Specialist TAM (STAM) for instructions. For more information, see [AWS Support Proactive Services](#).

- **Account and AWS Region quotas**

Be sure you have not reached the maximum number of Amazon EC2 instances or VPCs that you can create in your account and Region where you use this runbook. If you need to request a limit increase, see the [Service limit increase form](#).

- **Database configuration**

1. The database you specify in the `DatabaseName` parameter should have the `pg_stat_statements` extension configured. If you have not configured `pg_stat_statements` in `shared_preload_libraries`, then you must edit the value in the DB Parameter Group and apply the changes. Changes to the parameter `shared_preload_libraries` requires you to reboot your DB instance. For more information, see [Working with parameter groups](#). Adding `pg_stat_statements` to `shared_preload_libraries` will add some performance overhead. However, this is useful for tracking performance of individual statements. For more information about the `pg_stat_statements` extension, see the [PostgreSQL documentation](#). If you don't configure the `pg_stat_statements` extension or if the extension is not present in the database being used for statistics collection, the statement level analysis will not be presented in the operational review.
2. Make sure that `track_counts` and `track_activities` parameters are not turned off. If these parameters are turned off in the DB Parameter Group, no meaningful statistics will be available.

Changing these parameters will require you to reboot your DB instance. For more information, see [Working with parameters on your Amazon RDS for PostgreSQL DB instance](#).

3. If the `track_io_timing` parameter is turned off, the I/O level statistics will not be included in the operational review. Changing `track_io_timing` will require you to reboot your DB instance and will incur additional performance overhead depending on the DB instance workload. Despite the performance overhead for critical workloads, this parameter provides useful information related to I/O time per query.

### Billing and charges

Your AWS account will be charged for the costs associated with the temporary Amazon EC2 instance, associated Amazon EBS volume, the NAT gateway, and the data transferred while this automation is running. By default, this runbook creates a `t3.micro` Amazon Linux 2 instance to collect the statistics. The runbook starts and stops the instance between steps to reduce costs.

### Data security and governance

This runbook collects statistics by querying the [PostgreSQL statistics views and functions](#). Make sure the credentials provided in the `SecretId` parameter only allow read-only permissions to the statistics views and functions. As part of the automation, the collection scripts are uploaded to your Amazon S3 bucket and can be located in `s3://DOC-EXAMPLE-BUCKET/automation_execution_id/queries/`.

These scripts collect data that is used by an AWS Specialist to review key performance indicators at object level. The script collects information such as table name, schema name, and index name. If any of this information contains sensitive information like revenue indicators, username, email address, or any other personally identifiable information, then we recommend that you discontinue with this workload review. Contact your AWS TAM to discuss an alternative approach for the workload review.

Make sure you have the necessary approval and clearance to share the statistics and metadata collected by this automation with AWS.

### Security considerations

If you set the `UpdateRdsSecurityGroup` parameter to `yes`, the runbook updates the security group associated with your DB instance to allow inbound traffic from the temporary Amazon EC2 instance's private IP address.

If you set the `UpdateRdsRouteTable` parameter to `yes`, the runbook updates the route table associated with the subnet your DB instance is running in to allow traffic to the temporary Amazon EC2 instance through the VPC peering connection.

### User creation

To allow the collection script to connect to your Amazon RDS database, you must set up a user with permissions to read the statistic views. Then you must store the credentials in Secrets Manager. We recommend creating a new dedicated user for this automation. Creating a separate user allows you to audit and track activities performed by this automation.

1. Create a new user.

```
psql -h <database_connection_endpoint> -p <database_port> -U <admin_user> -c  
"CREATE USER <user_name> PASSWORD '<password>';"
```

2. Ensure that this user can only make read-only connections.

```
psql -h <database_connection_endpoint> -p <database_port> -U <admin_user> -c  
"ALTER USER <user_name> SET default_transaction_read_only=true;"
```

3. Set user level limits.

```
psql -h <database_connection_endpoint> -p <database_port> -U <admin_user> -c  
"ALTER USER <user_name> SET work_mem=4096;"
```

```
psql -h <database_connection_endpoint> -p <database_port> -U <admin_user> -c  
"ALTER USER <user_name> SET statement_timeout=10000;"
```

```
psql -h <database_connection_endpoint> -p <database_port> -U <admin_user> -c  
"ALTER USER <user_name> SET idle_in_transaction_session_timeout=60000;"
```

4. Grant `pg_monitor` permissions to the new user so it can access the DB statistics. (The `pg_monitor` role is a member of `pg_read_all_settings`, `pg_read_all_stats`, and `pg_stat_scan_table`.)

```
psql -h <database_connection_endpoint> -p <database_port> -U <admin_user> -c  
"GRANT pg_monitor to <user_name>;"
```

### Permissions added to the temporary Amazon EC2 instance profile by this Systems Manager Automation

The following permissions are added to the IAM role associated with the temporary Amazon EC2 instance. The `AmazonSSMManagedInstanceCore` managed policy is also associated with the IAM role to allow the Amazon EC2 instance to be managed by Systems Manager.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Action": [  
        "ec2:DescribeTags"  
      ],  
      "Resource": "*",  
      "Effect": "Allow"  
    },  
    {  
      "Action": [  
        "s3:GetBucketLocation"  
      ],  
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET",  
      "Effect": "Allow"  
    },  
    {  
      "Action": [  
        "s3:PutObject"  
      ],  
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/automation execution id/*",  
      "Effect": "Allow"  
    },  
    {  
      "Action": [  
        "secretsmanager:GetSecretValue"  
      ],  
      "Resource": "arn:aws:secretsmanager:region:account id:secret:secret id",  
      "Effect": "Allow"  
    },  
    {  
      "Action": [  
        "support:AddAttachmentsToSet",  
        "support:AddCommunicationToCase",  
        "support:DescribeCases"  
      ],  
      "Resource": "*",  
      "Effect": "Allow"  
    }  
  ]  
}
```



```
]
}
```

### Permissions added to the temporary maintenance window by this Systems Manager Automation

The following permissions are automatically added to the IAM role associated with the Maintenance Windows tasks. The Maintenance Windows tasks starts, stops, and sends commands to the temporary Amazon EC2 instance.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ssm:GetAutomationExecution",
        "ssm:ListCommands",
        "ssm:ListCommandInvocations",
        "ssm:GetCommandInvocation",
        "ssm:GetCalendarState",
        "ssm:CancelCommand",
        "ec2:DescribeInstanceStatus"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "ssm:SendCommand",
        "ec2:StartInstances",
        "ec2:StopInstances",
        "ssm:StartAutomationExecution"
      ],
      "Resource": [
        "arn:aws:ec2:region:account id:instance/temporary instance id",
        "arn:aws:ssm:*:*:document/AWS-RunShellScript",
        "arn:aws:ssm:*:*:automation-definition/AWS-StopEC2Instance:$DEFAULT",
        "arn:aws:ssm:*:*:automation-definition/AWS-StartEC2Instance:$DEFAULT"
      ],
      "Effect": "Allow"
    },
    {
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "ssm.amazonaws.com"
        }
      },
      "Action": "iam:PassRole",
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

[Run this Automation \(console\)](#)

#### Document type

Automation

#### Owner

Amazon

#### Platforms

## Databases

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- DBInstanceIdentifier

Type: String

Description: (Required) The ID of your DB instance.

- DatabaseName

Type: String

Description: (Required) The database name hosted on your DB instance.

- SecretId

Type: String

Description: (Required) The ARN of your Secrets Manager secret containing the username and password key value pair. The AWS CloudFormation stack creates an IAM policy with permissions for the `GetSecretValue` operation to this ARN. The credentials are used to allow the temporary instance to collect the database statistics. Contact your TAM or STAM to discuss the minimum required permissions.

- Acknowledge

Type: String

Description: (Required) Enter **yes** if you acknowledge that this runbook will create temporary resources in your account to collect statistics from your DB instance. We recommend contacting your TAM or STAM before running this automation.

- SupportCase

Type: String

Description: (Optional) The AWS Support case number provided by your TAM or STAM. If provided, the runbook updates the case and attaches the data collected. This option requires the temporary Amazon EC2 instance to have internet connectivity to access the AWS Support API endpoint. You must set the `AllowVpcInternetAccess` parameter to `true`. The case subject must contain the phrase `AWSPremiumSupport-PostgreSQLWorkloadReview`.

- S3BucketName

Type: String

Description: (Required) The Amazon S3 bucket name in your account where you want to upload the data collected by the automation. Verify the bucket policy does not grant any unnecessary read or write permissions to principals that do not need access to the contents of the bucket. We recommend creating a new temporary Amazon S3 bucket for the purpose of this automation. The runbook provides permissions to the `s3:PutObject` API operation to the IAM role attached to the temporary Amazon EC2 instance. The uploaded files will be located in `s3://bucket name/automation execution id/`.

- InstanceType

Type: String

Description: (Optional) The type of the temporary Amazon EC2 instance that will run the custom SQL and shell scripts.

Valid values: t2.micro | t2.small | t2.medium | t2.large | t3.micro | t3.small | t3.medium | t3.large

Default: t3.micro

- VpcCidr

Type: String

Description: (Optional) The IP address range in CIDR notation for the new VPC (for example, 172.31.0.0/16). Make sure you select a CIDR that does not overlap or match any existing VPC with connectivity to your DB instance. The smallest VPC you can create uses a /28 subnet mask, and the largest VPC uses a /16 subnet mask.

Default: 172.31.0.0/16

- StackResourcesNamePrefix

Type: String

Description: (Optional) The AWS CloudFormation stack resources name prefix and tag. The runbook creates the AWS CloudFormation stack resources using this prefix as part of the name and tag applied to the resources. The structure for the tag key-value pair is *StackResourcesNamePrefix*: `{{ automation:EXECUTION_ID }}`.

Default: AWSPostgreSQLWorkloadReview

- Schedule

Type: String

Description: (Optional) The maintenance window schedule. Specifies how often the maintenance window runs the tasks. The default value is every 1 hour.

Valid values: 15 minutes | 30 minutes | 1 hour | 2 hours | 4 hours | 6 hours | 12 hours | 1 day | 2 days | 4 days

Default: 1 hour

- Duration

Type: Integer

Description: (Optional) The maximum duration, in minutes, you want to allow the automation to run. The maximum duration supported is 8,640 minutes (6 days). The default value is 4,320 minutes (3 days).

Valid values: 30-8640

Default: 4320

- UpdateRdsRouteTable

Type: String

Description: (Optional) If set to `true`, the runbook updates the route table associated with the subnet your DB instance runs in. An IPv4 route is added to route traffic to the temporary Amazon EC2 instance private IPV4 address through the newly created VPC peering connection.

Valid values: true | false

Default: false

- AllowVpcInternetAccess

Type: String

Description: (Optional) If set to `true`, the runbook creates a NAT gateway to provide internet connectivity to the temporary Amazon EC2 instance to communicate with the AWS Support API endpoint. You can leave this parameter as `false` if you only want the runbook to upload the output to your Amazon S3 bucket.

Valid values: true | false

Default: false

- UpdateRdsSecurityGroup

Type: String

Description: (Optional) If set to `true`, the runbook updates the security group associated with your DB instance to allow traffic from the temporary instance's private IP address.

Valid values: false | true

Default: false

- EbsVolumeDeleteOnTermination

Type: String

Description: (Optional) If set to `true`, the temporary Amazon EC2 instance's root volume is deleted after the runbook completes and deletes the AWS CloudFormation stack.

Valid values: false | true

Default: false

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `cloudformation:CreateStack`
- `cloudformation>DeleteStack`
- `cloudformation:DescribeStackEvents`
- `cloudformation:DescribeStackResource`
- `cloudformation:DescribeStacks`
- `cloudformation:UpdateStack`
- `ec2:AcceptVpcPeeringConnection`
- `ec2:AllocateAddress`
- `ec2:AssociateRouteTable`
- `ec2:AssociateVpcCidrBlock`
- `ec2:AttachInternetGateway`
- `ec2:AuthorizeSecurityGroupEgress`
- `ec2:AuthorizeSecurityGroupIngress`
- `ec2:CreateEgressOnlyInternetGateway`

- ec2:CreateInternetGateway
- ec2:CreateNatGateway
- ec2:CreateRoute
- ec2:CreateRouteTable
- ec2:CreateSecurityGroup
- ec2:CreateSubnet
- ec2:CreateTags
- ec2:CreateVpc
- ec2:CreateVpcEndpoint
- ec2:CreateVpcPeeringConnection
- ec2>DeleteEgressOnlyInternetGateway
- ec2>DeleteInternetGateway
- ec2>DeleteNatGateway
- ec2>DeleteRoute
- ec2>DeleteRouteTable
- ec2>DeleteSecurityGroup
- ec2>DeleteSubnet
- ec2>DeleteTags
- ec2>DeleteVpc
- ec2>DeleteVpcEndpoints
- ec2:DescribeAddresses
- ec2:DescribeEgressOnlyInternetGateways
- ec2:DescribeImages
- ec2:DescribeInstances
- ec2:DescribeInstanceStatus
- ec2:DescribeInternetGateways
- ec2:DescribeNatGateways
- ec2:DescribeRouteTables
- ec2:DescribeSecurityGroups
- ec2:DescribeSubnets
- ec2:DescribeVpcEndpoints
- ec2:DescribeVpcPeeringConnections
- ec2:DescribeVpcs
- ec2:DetachInternetGateway
- ec2:DisassociateRouteTable
- ec2:DisassociateVpcCidrBlock
- ec2:ModifySubnetAttribute
- ec2:ModifyVpcAttribute
- ec2:RebootInstances
- ec2:ReleaseAddress
- ec2:RevokeSecurityGroupEgress
- ec2:RevokeSecurityGroupIngress
- ec2:StartInstances
- ec2:StopInstances
- ec2:RunInstances
- ec2:TerminateInstances

- iam:AddRoleToInstanceProfile
- iam:AttachRolePolicy
- iam:CreateInstanceProfile
- iam:CreateRole
- iam>DeleteInstanceProfile
- iam>DeleteRole
- iam>DeleteRolePolicy
- iam:DetachRolePolicy
- iam:GetInstanceProfile
- iam:GetRole
- iam:GetRolePolicy
- iam:PassRole
- iam:PutRolePolicy
- iam:RemoveRoleFromInstanceProfile
- iam:TagPolicy
- iam:TagRole
- rds:DescribeDBInstances
- s3:GetAccountPublicAccessBlock
- s3:GetBucketAcl
- s3:GetBucketPolicyStatus
- s3:GetBucketPublicAccessBlock
- s3:ListBucket
- ssm:AddTagsToResource
- ssm:CancelMaintenanceWindowExecution
- ssm:CreateDocument
- ssm:CreateMaintenanceWindow
- ssm>DeleteDocument
- ssm>DeleteMaintenanceWindow
- ssm:DeregisterTaskFromMaintenanceWindow
- ssm:DescribeAutomationExecutions
- ssm:DescribeDocument
- ssm:DescribeInstanceInformation
- ssm:DescribeMaintenanceWindowExecutions
- ssm:GetCalendarState
- ssm:GetDocument
- ssm:GetMaintenanceWindowExecution
- ssm:GetParameters
- ssm:ListCommandInvocations
- ssm:ListCommands
- ssm:ListTagsForResource
- ssm:RegisterTaskWithMaintenanceWindow
- ssm:RemoveTagsFromResource
- ssm:SendCommand
- support:AddAttachmentsToSet
- support:AddCommunicationToCase
- support:DescribeCases

## Document Steps

1. `aws:assertAwsResourceProperty` - Confirms the DB instance is in the available state.
2. `aws:executeAwsApi` - Gathers details about the DB instance.
3. `aws:executeScript` - Checks if the Amazon S3 bucket specified in the `S3BucketName` allows anonymous, or public read or write access permissions.
4. `aws:executeScript` - Gets the AWS CloudFormation template content from the Automation runbook attachment that is used to create the temporary AWS resources in your AWS account.
5. `aws:createStack` - Creates the AWS CloudFormation stack resources.
6. `aws:waitForAwsResourceProperty` - Waits until the Amazon EC2 instance created by the AWS CloudFormation template is running.
7. `aws:executeAwsApi` - Gets the IDs for the temporary Amazon EC2 instance and VPC peering connection created by AWS CloudFormation.
8. `aws:executeAwsApi` - Gets the IP address for the temporary Amazon EC2 instance to configure connectivity with your DB instance.
9. `aws:executeAwsApi` - Tags the Amazon EBS volume attached to the temporary Amazon EC2 instance.
10. `aws:waitForAwsResourceProperty` - Waits until the temporary Amazon EC2 instance passes status checks.
11. `aws:waitForAwsResourceProperty` - Waits until the temporary Amazon EC2 instance is managed by Systems Manager. If this step times out or fails, then the runbook reboots the instance.
  - a. `aws:executeAwsApi` - Reboots the temporary Amazon EC2 instance if the previous step failed or timed out.
  - b. `aws:waitForAwsResourceProperty` - Waits until the temporary Amazon EC2 instance is managed by Systems Manager after reboot.
12. `aws:runCommand` - Installs the metadata collector application requirements on the temporary Amazon EC2 instance.
13. `aws:runCommand` - Configures access to your DB instance by creating a configuration file on the temporary Amazon EC2 instance.
14. `aws:executeAwsApi` - Creates a maintenance window to periodically run the metadata collector application using Run Command. The maintenance window starts and stops the instance between commands.
15. `aws:waitForAwsResourceProperty` - Waits until the maintenance window created by the AWS CloudFormation template is ready.
16. `aws:executeAwsApi` - Gets the IDs for the maintenance window and change calendar created by AWS CloudFormation.
17. `aws:sleep` - Waits until the end date of the maintenance window.
18. `aws:executeAwsApi` - Turns off the maintenance window.
19. `aws:executeScript` - Gets the results of the tasks run during the maintenance window.
20. `aws:waitForAwsResourceProperty` - Waits for the maintenance window to finish the last task before continuing.
21. `aws:branch` - Branches the workflow based on whether you provided a value for the `SupportCase` parameter.
  - a. `aws:changeInstanceState` - Starts the temporary Amazon EC2 instance and waits for status checks to pass before uploading the report.
  - b. `aws:waitForAwsResourceProperty` - Waits until the temporary Amazon EC2 instance is managed by Systems Manager. If this step times out or fail, then the runbook reboots the instance.
    - i. `aws:executeAwsApi` - Reboots the temporary Amazon EC2 instance if the previous step failed or timed out.
    - ii. `aws:waitForAwsResourceProperty` - Waits until the temporary Amazon EC2 instance is managed by Systems Manager after reboot.

- c. `aws:runCommand` - Attaches the metadata report to the AWS Support case if you provided a value for the `SupportCase` parameter. The script compresses and splits the report into 5 MB files. The maximum number of files the script attaches to a AWS Support case is 12.
- 22 `aws:changeInstanceState` - Stops the temporary Amazon EC2 instance in case the AWS CloudFormation stack fails to delete.
- 23 `aws:executeAwsApi` - Describes the AWS CloudFormation stack events if the runbooks fails to create or update the AWS CloudFormation stack.
- 24 `aws:waitForAwsResourceProperty` - Waits until the AWS CloudFormation stack is in a terminal status before deleting.
- 25 `aws:executeAwsApi` - Deletes the AWS CloudFormation stack excluding the maintenance window. The root Amazon EBS volume associated with the temporary Amazon EC2 instance is preserved if the `EbsVolumeDeleteOnTermination` parameter value was set to `false`.

## AWS-RebootRdsInstance

### Description

The `AWS-RebootRdsInstance` runbook reboots an Amazon Relational Database Service (Amazon RDS) DB instance if it isn't already rebooting.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `InstanceId`

Type: String

Description: (Required) The ID of the Amazon RDS DB instance that you want to reboot.

### Document Steps

`RebootInstance` - Reboots the DB instance if it is not already rebooting.

`WaitForAvailableState` - Waits for the DB instance to complete the reboot process.

### Outputs



This automation has no outputs.

## AWSSupport-ShareRDSSnapshot

### Description

The `AWSSupport-ShareRDSSnapshot` runbook provides an automated solution for the procedure outlined in the Knowledge Center article [How can I share an encrypted Amazon RDS DB snapshot with another account?](#) If your Amazon Relational Database Service (Amazon RDS) snapshot was encrypted using the default AWS managed key, you cannot share the snapshot. In this case, you must copy the snapshot using a customer managed key, and then share the snapshot with the target account. This automation performs these steps using the value you specify in the `SnapshotName` parameter, or the latest snapshot found for the selected Amazon RDS DB instance or cluster.

#### Note

If you do not specify a value for the `KMSKey` parameter, the automation creates a new AWS KMS customer managed key in your account that is used to encrypt the snapshot.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AccountIds

Type: StringList

Description: (Required) Comma-separated list of account IDs to share the snapshot with.

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Database

Type: String

Description: (Required) The name of the Amazon RDS DB instance or cluster whose snapshot you want to share. This parameter is optional if you specify a value for the `SnapshotName` parameter.

- KMSKey

Type: String

Description: (Optional) The full Amazon Resource Name (ARN) of the AWS KMS customer managed key used to encrypt the snapshot.

- `SnapshotName`

Type: String

Description: (Optional) The ID of the DB cluster or instance snapshot that you want to use.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `rds:DescribeDBInstances`
- `rds:DescribeDBSnapshots`
- `rds:CopyDBSnapshot`
- `rds:ModifyDBSnapshotAttribute`

The `AutomationAssumeRole` requires the following actions to successfully start the runbook for a DB cluster.

- `ssm:StartAutomationExecution`
- `rds:DescribeDBClusters`
- `rds:DescribeDBClusterSnapshots`
- `rds:CopyDBClusterSnapshot`
- `rds:ModifyDBClusterSnapshotAttribute`

The IAM role used to run the automation must be added as a key user to use the KMS key specified in the `ARNKmsKey` parameter. For information about adding key users to a KMS key, see [Changing a key policy](#) in the *AWS Key Management Service Developer Guide*.

The `AutomationAssumeRole` requires the following additional actions to successfully start the runbook if you do not specify a value for the `KMSKey` parameter.

- `kms:CreateKey`
- `kms:ScheduleKeyDeletion`

### Document Steps

1. `aws:executeScript` - Checks whether a value was provided for the `KMSKey` parameter, and creates a AWS KMS customer managed key if no value is found.
2. `aws:branch` - Checks whether a value was provided for the `SnapshotName` parameter, and branches accordingly.
3. `aws:executeAwsApi` - Checks whether the snapshot provided is from a DB instance.
4. `aws:executeScript` - Formats the `SnapshotName` parameter replacing colons with a hyphen.
5. `aws:executeAwsApi` - Copies the snapshot using the specified `KMSKey`.
6. `aws:waitForAwsResourceProperty` - Waits for the copy snapshot operation to complete.
7. `aws:executeAwsApi` - Shares the new snapshot with the `AccountIds` specified.
8. `aws:executeAwsApi` - Checks whether the snapshot provided is from a DB cluster.
9. `aws:executeScript` - Formats the `SnapshotName` parameter replacing colons with a hyphen.
10. `aws:executeAwsApi` - Copies the snapshot using the specified `KMSKey`.
11. `aws:waitForAwsResourceProperty` - Waits for the copy snapshot operation to complete.
12. `aws:executeAwsApi` - Shares the new snapshot with the `AccountIds` specified.

- 13aws:executeAwsApi - Checks whether the value provided for the Database parameter is a DB instance.
- 14aws:executeAwsApi - Checks whether the value provided for the Database parameter is a DB cluster.
- 15aws:executeAwsApi - Retrieves a list of snapshots for the specified Database .
- 16aws:executeScript - Determines the latest snapshot available from the list assembled in the previous step.
- 17aws:executeAwsApi - Copies the DB instance snapshot using the specified KMSKey .
- 18aws:waitForAwsResourceProperty - Waits for the copy snapshot operation to complete.
- 19aws:executeAwsApi - Shares the new snapshot with the AccountIds specified.
- 20aws:executeAwsApi - Retrieves a list of snapshots for the specified Database .
- 21aws:executeScript - Determines the latest snapshot available from the list assembled in the previous step.
- 22aws:executeAwsApi - Copies the DB instance snapshot using the specified KMSKey .
- 23aws:waitForAwsResourceProperty - Waits for the copy snapshot operation to complete.
- 24aws:executeAwsApi - Shares the new snapshot with the AccountIds specified.
- 25aws:executeScript - Deletes the AWS KMS customer managed key created by the automation if you did not specify a value for the KMSKey parameter and the automation fails.

## AWS-StartRdsInstance

### Description

Start an Amazon Relational Database Service (Amazon RDS) instance.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) ID of the Amazon RDS instance to start.

## AWSsupport-TroubleshootConnectivityToRDS

### Description

The AWSsupport-TroubleshootConnectivityToRDS runbook diagnoses connectivity issues between an EC2 instance and an Amazon Relational Database Service instance. The automation ensures the DB instance is available, and then checks the associated security group rules, network access control lists (network ACLs), and route tables for potential connectivity issues.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- DBInstanceIdentifier

Type: String

Description: (Required) The DB instance ID to test connectivity to.

- SourceInstance

Type: String

Allowed pattern: `^i-[a-z0-9]{8,17}$`

Description: (Required) The ID of the EC2 instance to test connectivity from.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ec2:DescribeInstances
- ec2:DescribeNetworkAcls
- ec2:DescribeRouteTables
- ec2:DescribeSecurityGroups
- ec2:DescribeSubnets
- rds:DescribeDBInstances

### Document Steps

- `aws:assertAwsResourceProperty` - Confirms the DB instance status is available .
- `aws:executeAwsApi` - Gets information about the DB instance.
- `aws:executeAwsApi` - Gets information about the DB instance network ACLs.
- `aws:executeAwsApi` - Gets the DB instance subnet CIDR.
- `aws:executeAwsApi` - Gets information about the EC2 instance.
- `aws:executeAwsApi` - Gets information about the EC2 instance network ACLs.
- `aws:executeAwsApi` - Gets information about the security groups associated with the EC2 instance.
- `aws:executeAwsApi` - Gets information about the security groups associated with the DB instance.
- `aws:executeAwsApi` - Gets information about the route tables associated with the EC2 instance.
- `aws:executeAwsApi` - Gets information about the main route table associated with the Amazon VPC for the EC2 instance.
- `aws:executeAwsApi` - Gets information about the route tables associated with the DB instance.
- `aws:executeAwsApi` - Gets information about the main route table associated with the Amazon VPC for the DB instance.
- `aws:executeScript` - Evaluates security group rules.
- `aws:executeScript` - Evaluates network ACLs.
- `aws:executeScript` - Evaluates route tables.
- `aws:sleep` - Ends the automation.

### Outputs

`getRDSInstanceProperties.DBInstanceIdentifier` - The DB instance used in the automation.

`getRDSInstanceProperties.DBInstanceStatus` - The current status of the DBInstance.

`evalSecurityGroupRules.SecurityGroupEvaluation` - Results from comparing the `SourceInstance` security group rules to the DB instance security group rules.

`evalNetworkAclRules.NetworkAclEvaluation` - Results from comparing the `SourceInstance` network ACLs to the DB instance network ACLs.

`evalRouteTableEntries.RouteTableEvaluation` - Results from comparing the `SourceInstance` route table to the DB instance routes.

## Amazon Redshift

AWS Systems Manager Automation provides predefined runbooks for Amazon Redshift. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-DeleteRedshiftCluster \(p. 239\)](#)
- [AWSConfigRemediation-DisablePublicAccessToRedshiftCluster \(p. 240\)](#)
- [AWSConfigRemediation-EnableRedshiftClusterAuditLogging \(p. 241\)](#)
- [AWSConfigRemediation-EnableRedshiftClusterAutomatedSnapshot \(p. 242\)](#)
- [AWSConfigRemediation-EnableRedshiftClusterEncryption \(p. 243\)](#)
- [AWSConfigRemediation-EnableRedshiftClusterEnhancedVPCRouting \(p. 244\)](#)
- [AWSConfigRemediation-EnforceSSLOnlyConnectionsToRedshiftCluster \(p. 245\)](#)

- [AWSConfigRemediation-ModifyRedshiftClusterMaintenanceSettings](#) (p. 246)
- [AWSConfigRemediation-ModifyRedshiftClusterNodeType](#) (p. 248)

## AWSConfigRemediation-DeleteRedshiftCluster

### Description

The `AWSConfigRemediation-DeleteRedshiftCluster` runbook deletes the Amazon Redshift cluster you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-]+`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `ClusterIdentifier`

Type: String

Description: (Required) The ID of the Amazon Redshift cluster that you want to delete.

- `SkipFinalClusterSnapshot`

Type: Boolean

Default: `false`

Description: (Optional) If set to `false`, the automation creates a snapshot before deleting the Amazon Redshift cluster. If set to `true`, a final cluster snapshot is not created.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `redshift>DeleteCluster`

- `redshift:DescribeClusters`

### Document Steps

- `aws:branch` - Branches based on the value you specify for the `SkipFinalClusterSnapshot` parameter.
- `aws:executeAwsApi` - Deletes the Amazon Redshift cluster specified in the `ClusterIdentifier` parameter.
- `aws:assertAwsResourceProperty` - Verifies the Amazon Redshift cluster has been deleted.

## AWSConfigRemediation- DisablePublicAccessToRedshiftCluster

### Description

The `AWSConfigRemediation-DisablePublicAccessToRedshiftCluster` runbook disables public accessibility for the Amazon Redshift cluster that you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `ClusterIdentifier`

Type: String

Description: (Required) The unique identifier of the cluster that you want to disable public accessibility for.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

- `redshift:DescribeClusters`
- `redshift:ModifyCluster`

### Document Steps

- `aws:executeAwsApi` - Disables public accessibility for the cluster specified in the `ClusterIdentifier` parameter.
- `aws:waitForAwsResourceProperty` - Waits for the state of the cluster to change to `available`.
- `aws:assertAwsResourceProperty` - Confirms the public accessibility setting is disabled on the cluster.

## AWSConfigRemediation- EnableRedshiftClusterAuditLogging

### Description

The `AWSConfigRemediation-EnableRedshiftClusterAuditLogging` runbook enables audit logging for the Amazon Redshift cluster you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `BucketName`

Type: String

Description: (Required) The name of the Amazon Simple Storage Service (Amazon S3) bucket you want to upload logs to.

- `ClusterIdentifier`

Type: String

Description: (Required) The unique identifier of the cluster you want to enable audit logging on.

- `S3KeyPrefix`



Type: String

Description: (Optional) The Amazon S3 key prefix (subfolder) you want to upload logs to.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `redshift:DescribeLoggingStatus`
- `redshift:EnableLogging`
- `s3:GetBucketAcl`
- `s3:PutObject`

### Document Steps

- `aws:branch` - Branches based on whether a value was specified for the `S3KeyPrefix` parameter.
- `aws:executeAwsApi` - Enables audit logging on the cluster specified in the `ClusterIdentifier` parameter.
- `aws:assertAwsResourceProperty` - Verifies audit logging was enabled on the cluster.

## AWSConfigRemediation- EnableRedshiftClusterAutomatedSnapshot

### Description

The `AWSConfigRemediation-EnableRedshiftClusterAutomatedSnapshot` runbook enables automated snapshots for the Amazon Redshift cluster you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- AutomatedSnapshotRetentionPeriod

Type: Integer

Valid values: 1-35

Description: (Required) The number of days that automated snapshots are retained.

- ClusterIdentifier

Type: String

Description: (Required) The unique identifier of the cluster you want to enable automated snapshots on.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `redshift:DescribeClusters`
- `redshift:ModifyCluster`

### Document Steps

- `aws:executeAwsApi` - Enables automation snapshots on the cluster specified in the `ClusterIdentifier` parameter.
- `aws:waitForAwsResourceProperty` - Waits for the state of the cluster to change to `available`.
- `aws:executeScript` - Confirms automated snapshots were enabled on the cluster.

## AWSConfigRemediation- EnableRedshiftClusterEncryption

### Description

The `AWSConfigRemediation-EnableRedshiftClusterEncryption` runbook enables encryption on the Amazon Redshift cluster you specify using an AWS Key Management Service (AWS KMS) customer managed key. This runbook should only be used as a baseline to ensure that your Amazon Redshift clusters are encrypted according to minimum recommended security best practices. We recommend encrypting multiple clusters with different customer managed keys. This runbook cannot change the AWS KMS customer managed key used on an already encrypted cluster. To change the AWS KMS customer managed key used to encrypt a cluster, you must first disable encryption on the cluster.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

## Databases

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- ClusterIdentifier

Type: String

Description: (Required) The unique identifier of the cluster you want to enable encryption on.

- KMSKeyARN

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the AWS KMS customer managed key you want to use to encrypt the cluster's data.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- redshift:DescribeClusters
- redshift:ModifyCluster

### Document Steps

- aws:executeAwsApi - Enables encryption on the Amazon Redshift cluster specified in the ClusterIdentifier parameter.
- aws:assertAwsResourceProperty - Verifies encryption has been enabled on the cluster.

## AWSConfigRemediation- EnableRedshiftClusterEnhancedVPCRouting

### Description

The AWSConfigRemediation-EnableRedshiftClusterEnhancedVPCRouting runbook enables enhanced virtual private cloud (VPC) routing for the Amazon Redshift cluster you specify. For information about enhanced VPC routing, see [Amazon Redshift enhanced VPC routing](#) in the *Amazon Redshift Management Guide* .

[Run this Automation \(console\)](#)

### Document type

Automation

**Owner**

Amazon

**Platforms**

Databases

**Parameters**

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([w+=,./-])+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- ClusterIdentifier

Type: String

Description: (Required) The unique identifier of the cluster you want to enable enhanced VPC routing on.

**Required IAM permissions**

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- redshift:DescribeClusters
- redshift:ModifyCluster

**Document Steps**

- aws:executeAwsApi - Enables enhanced VPC routing on the cluster specified in the ClusterIdentifier parameter.
- assertAwsResourceProperty - Confirms enhanced VPC routing was enabled on the cluster.

## AWSConfigRemediation- EnforceSSLOnlyConnectionsToRedshiftCluster

**Description**

The AWSConfigRemediation-EnforceSSLOnlyConnectionsToRedshiftCluster runbook requires incoming connections to use SSL for the Amazon Redshift cluster you specify.

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

## Platforms

Databases

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[w+=,./-]+\d{12}

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- ClusterIdentifier

Type: String

Description: (Required) The unique identifier of the cluster you want to enable enhanced VPC routing on.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- redshift:DescribeClusters
- redshift:DescribeClusterParameters
- redshift:ModifyClusterParameterGroup

## Document Steps

- aws:executeAwsApi - Gathers parameter details from the cluster specified in the ClusterIdentifier parameter.
- aws:executeAwsApi - Enables the require\_ssl setting on the cluster specified in the ClusterIdentifier parameter.
- aws:assertAwsResourceProperty - Confirms the require\_ssl setting was enabled on the cluster.
- aws:executeScript - Verifies the require\_ssl setting for the cluster.

# AWSConfigRemediation- ModifyRedshiftClusterMaintenanceSettings

## Description

The AWSConfigRemediation-ModifyRedshiftClusterMaintenanceSettings runbook modifies the maintenance settings for the Amazon Redshift cluster you specify.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Databases

## Parameters

- AllowVersionUpgrade

Type: Boolean

Description: (Required) If set to `true`, major version upgrades are applied automatically to the cluster during the maintenance window.

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- AutomatedSnapshotRetentionPeriod

Type: Integer

Valid values: 1-35

Description: (Required) The number of days automated snapshots are retained.

- ClusterIdentifier

Type: String

Description: (Required) The unique identifier of the cluster you want to enable enhanced VPC routing on.

- PreferredMaintenanceWindow

Type: String

Description: (Required) The weekly time range (in UTC) during which system maintenance can occur.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `redshift:DescribeClusters`
- `redshift:ModifyCluster`

## Document Steps

- `aws:executeAwsApi` - Modifies the maintenance settings for the cluster specified in the `ClusterIdentifier` parameter.
- `aws:assertAwsResourceProperty` - Confirms the modified maintenance settings were configured for the cluster.

## AWSConfigRemediation- ModifyRedshiftClusterNodeType

### Description

The `AWSConfigRemediation-ModifyRedshiftClusterNodeType` runbook modifies the node type and number of nodes for the Amazon Redshift cluster you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Databases

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[w+=,./-]+\s$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `Classic`

Type: Boolean

Description: (Optional) If set to `true`, the resize operation uses the classic resize process.

- `ClusterIdentifier`

Type: String

Description: (Required) The unique identifier of the cluster whose node type you want to modify.

- `ClusterType`

Type: String

Valid values: `single-node | multi-node`

Description: (Required) The type of cluster you want to assign to your cluster.

- `NodeType`

Type: String

Valid values: ds2.xlarge | ds2.8xlarge | dc1.large | dc1.8xlarge | dc2.large | dc2.8xlarge | ra3.4xlarge | ra3.16xlarge

Description: (Required) The type of node you want to assign to your cluster.

- NumberOfNodes

Type: Integer

Valid values: 2-100

Description: (Optional) The number of nodes you want to assign to your cluster. If your cluster is a `single-node` type, do not specify a value for this parameter.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `redshift:DescribeClusters`
- `redshift:ResizeCluster`

### Document Steps

- `aws:executeScript` - Modifies the node type and number of nodes for the cluster specified in the `ClusterIdentifier` parameter.

## Amazon S3

AWS Systems Manager Automation provides predefined runbooks for Amazon Simple Storage Service. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWS-ConfigureS3BucketLogging \(p. 249\)](#)
- [AWS-ConfigureS3BucketVersioning \(p. 251\)](#)
- [AWSConfigRemediation-ConfigureS3BucketPublicAccessBlock \(p. 252\)](#)
- [AWSConfigRemediation-ConfigureS3PublicAccessBlock \(p. 253\)](#)
- [AWS-DisableS3BucketPublicReadWrite \(p. 255\)](#)
- [AWS-EnableS3BucketEncryption \(p. 255\)](#)
- [AWSConfigRemediation-RemovePrincipalStarFromS3BucketPolicy \(p. 256\)](#)
- [AWSConfigRemediation-RestrictBucketSSLRequestsOnly \(p. 257\)](#)
- [AWSSupport-TroubleshootS3PublicRead \(p. 258\)](#)

## AWS-ConfigureS3BucketLogging

### Description

Enable logging on an Amazon Simple Storage Service (Amazon S3) bucket.



[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- BucketName

Type: String

Description: (Required) The name of the Amazon S3 Bucket for which you want to configure logging.

- GrantedPermission

Type: String

Valid values: FULL\_CONTROL | READ | WRITE

Description: (Required) Logging permissions assigned to the grantee for the bucket.

- GranteeEmailAddress

Type: String

(Optional) Email address of the grantee.

- GranteeId

Type: String

Description: (Optional) The canonical user ID of the grantee.

- GranteeType

Type: String

Valid values: CanonicalUser | AmazonCustomerByEmail | Group

Description: (Required) Type of grantee.

- GranteeUri

Type: String

Description: (Optional) URI of the grantee group.

- TargetBucket

Type: String

Description: (Required) Specifies the bucket where you want Amazon S3 to store server access logs. You can have your logs delivered to any bucket that you own. You can also configure multiple buckets to deliver their logs to the same target bucket. In this case you should choose a different TargetPrefix for each source bucket so that the delivered log files can be distinguished by key.

- TargetPrefix

Type: String

Default: /

Description: (Optional) Specifies a prefix for the keys under which the log files will be stored.

## AWS-ConfigureS3BucketVersioning

### Description

Configure versioning for an Amazon Simple Storage Service (Amazon S3) bucket.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- BucketName

Type: String

Description: (Required) The name of the Amazon S3 bucket you want to configure versioning for.

- VersioningState

Type: String

Valid values: Enabled | Suspended

Default: Enabled

Description: (Optional) Applied to the VersioningConfiguration.Status. When set to 'Enabled', this process enables versioning for the objects in the bucket, all objects added to the bucket receive a

unique version ID. When set to `Suspended`, this process disables versioning for the objects in the bucket. All objects added to the bucket receive the version ID `null`.

## AWSConfigRemediation- ConfigureS3BucketPublicAccessBlock

### Description

The `AWSConfigRemediation-ConfigureS3BucketPublicAccessBlock` runbook configures the Amazon Simple Storage Service (Amazon S3) public access block settings for an Amazon S3 bucket based on the values you specify in the runbook parameters.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- BlockPublicAcls

Type: Boolean

Default: `true`

Description: (Optional) If set to `true`, Amazon S3 blocks public access control lists (ACLs) for the S3 bucket, and objects stored in the S3 bucket you specify in the `BucketName` parameter.

- BlockPublicPolicy

Type: Boolean

Default: `true`

Description: (Optional) If set to `true`, Amazon S3 blocks public bucket policies for the S3 bucket you specify in the `BucketName` parameter.

- BucketName

Type: String

Description: (Required) The name of the S3 bucket you want to configure.

- `IgnorePublicAcls`

Type: Boolean

Default: true

Description: (Optional) If set to `true`, Amazon S3 ignores all public ACLs for the S3 bucket you specify in the `BucketName` parameter.

- `RestrictPublicBuckets`

Type: Boolean

Default: true

Description: (Optional) If set to `true`, Amazon S3 restricts public bucket policies for the S3 bucket you specify in the `BucketName` parameter.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `s3:GetAccountPublicAccessBlock`
- `s3:PutAccountPublicAccessBlock`
- `s3:GetBucketPublicAccessBlock`
- `s3:PutBucketPublicAccessBlock`

### Document Steps

- `aws:executeAwsApi` - Creates or modifies the `PublicAccessBlock` configuration for the S3 bucket specified in the `BucketName` parameter.
- `aws:executeScript` - Returns the `PublicAccessBlock` configuration for the S3 bucket specified in the `BucketName` parameter, and verifies the changes were successfully made based on the values specified in the runbook parameters.

## AWSConfigRemediation- ConfigureS3PublicAccessBlock

### Description

The `AWSConfigRemediation-ConfigureS3PublicAccessBlock` runbook configures an AWS account's Amazon Simple Storage Service (Amazon S3) public access block settings based on the values you specify in the runbook parameters.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- `AccountId`

Type: String

Description: (Required) The ID of the AWS account that owns the S3 bucket you are configuring.

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `BlockPublicAcls`

Type: Boolean

Default: `true`

Description: (Optional) If set to `true`, Amazon S3 blocks public access control lists (ACLs) for S3 buckets owned by the AWS account you specify in the `AccountId` parameter.

- `BlockPublicPolicy`

Type: Boolean

Default: `true`

Description: (Optional) If set to `true`, Amazon S3 blocks public bucket policies for S3 buckets owned by the AWS account you specify in the `AccountId` parameter.

- `IgnorePublicAcls`

Type: Boolean

Default: `true`

Description: (Optional) If set to `true`, Amazon S3 ignores all public ACLs for S3 buckets owned by the AWS account you specify in the `AccountId` parameter.

- `RestrictPublicBuckets`

Type: Boolean

Default: `true`

Description: (Optional) If set to `true`, Amazon S3 restricts public bucket policies for S3 buckets owned by the AWS account you specify in the `AccountId` parameter.

## Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

- `s3:GetAccountPublicAccessBlock`
- `s3:PutAccountPublicAccessBlock`

### Document Steps

- `aws:executeAwsApi` - Creates or modifies the `PublicAccessBlock` configuration for the AWS account specified in the `AccountId` parameter.
- `aws:executeScript` - Returns the `PublicAccessBlock` configuration for the AWS account specified in the `AccountId` parameter, and verifies the changes were successfully made based on the values specified in the runbook parameters.

## AWS-DisableS3BucketPublicReadWrite

### Description

Use Amazon Simple Storage Service (Amazon S3) `Block Public Access` to disable read and write access for a public S3 bucket. For more information, see [Using Amazon S3 Block Public Access](#) in the *Amazon Simple Storage Service User Guide* .

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `S3BucketName`

Type: String

Description: (Required) S3 bucket on which you want to restrict access.

## AWS-EnableS3BucketEncryption

### Description

Configures default encryption for an Amazon Simple Storage Service (Amazon S3) bucket.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- BucketName

Type: String

Description: (Required) The name of the S3 bucket where you want to encrypt the contents.

- SSEAlgorithm

Type: String

Default: AES256

Description: (Optional) Server-side encryption algorithm to use for the default encryption.

## AWSConfigRemediation- RemovePrincipalStarFromS3BucketPolicy

### Description

The AWSConfigRemediation-RemovePrincipalStarFromS3BucketPolicy runbook removes principal policy statements that have wildcards (`Principal: *` or `Principal: "AWS": *`) for `Allow` actions from your Amazon Simple Storage Service (Amazon S3) bucket policy. Policy statements with conditions are also removed.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- BucketName

Type: String

Description: (Required) The name of the Amazon S3 bucket whose policy you want to modify.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `s3:DeleteBucketPolicy`
- `s3:GetBucketPolicy`
- `s3:PutBucketPolicy`

### Document Steps

- `aws:executeScript` - Modifies the bucket policy and verifies principal policy statements with wildcards have been removed from the Amazon S3 bucket you specify in the `BucketName` parameter.

## AWSConfigRemediation- RestrictBucketSSLRequestsOnly

### Description

The `AWSConfigRemediation-RestrictBucketSSLRequestsOnly` runbook creates an Amazon Simple Storage Service (Amazon S3) bucket policy statement that explicitly denies HTTP requests to the Amazon S3 bucket you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters



- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- BucketName

Type: String

Description: (Required) The name of the S3 bucket that you want to deny HTTP requests.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `s3:DeleteBucketPolicy`
- `s3:GetBucketPolicy`
- `s3:PutEncryptionConfiguration`
- `s3:PutBucketPolicy`

### Document Steps

- `aws:executeScript` - Creates a bucket policy for the S3 bucket specified in the `BucketName` parameter that explicitly denies HTTP requests.

## AWSSupport-TroubleshootS3PublicRead

### Description

The `AWSSupport-TroubleshootS3PublicRead` runbook diagnoses issues reading objects from the public Amazon Simple Storage Service (Amazon S3) bucket you specify in the `S3BucketName` parameter. A subset of settings are also analyzed for objects in the S3 bucket.

[Run this Automation \(console\)](#)

### Limitations

- This automation does not check for access points that allow public access to objects.
- This automation does not evaluate condition keys in the S3 bucket policy.
- If you're using AWS Organizations, this automation does not evaluate service control policies to confirm that access to Amazon S3 is allowed.

### Document type

Automation

### Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- CloudWatchLogGroupName

Type: String

Description: (Optional) The Amazon CloudWatch Logs log group where you want to send the automation output. If a log group is not found that matches the value you specify, the automation will create a log group using this parameter value. The retention period for the log group created by this automation is 14 days.

- CloudWatchLogStreamName

Type: String

Description: (Optional) The CloudWatch Logs log stream where you want to send the automation output. If a log stream is not found that matches the value you specify, the automation will create a log stream using this parameter value. If you do not specify a value for this parameter, the automation will use the `ExecutionId` for the name of the log stream.

- HttpGet

Type: Boolean

Valid values: true | false

Default: true

Description: (Optional) If this parameter is set to `true`, the automation makes a partial HTTP request to the objects in the `S3BucketName` you specify. Only the first byte of the object is returned using the Range HTTP header.

- IgnoreBlockPublicAccess

Type: Boolean

Valid values: true | false

Default: false

Description: (Optional) If this parameter is set to `true`, the automation ignores the public access block settings of the S3 bucket you specify in the `S3BucketName` parameter. Changing this parameter from the default value is not recommended.

- MaxObjects

Type: Integer

Valid values: 1-25

Default: 5

Description: (Optional) The number of objects to analyze in the S3 bucket you specify in the `S3BucketName` parameter.

- `S3BucketName`

Type: String

Description: (Required) The name of the S3 bucket to troubleshoot.

- `S3PrefixName`

Type: String

Description: (Optional) The key name prefix of the objects you want to analyze in your S3 bucket. For more information, see [Object keys](#) in the *Amazon Simple Storage Service User Guide* .

- `StartAfter`

Type: String

Description: (Optional) The object key name where you want the automation to begin analyzing objects in your S3 bucket.

- `ResourcePartition`

Type: String

Valid values: `aws` | `aws-us-gov` | `aws-cn`

Default: `aws`

Description: (Required) The partition where your S3 bucket is located.

- `Verbose`

Type: Boolean

Valid values: `true` | `false`

Default: `false`

Description: (Optional) To return more detailed information during the automation, set this parameter to `true` . Only warning and error messages will be returned if the parameter is set to `false` .

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

The `logs:CreateLogGroup` , `logs:CreateLogStream` , and `logs:PutLogEvents` permissions are only required if you want the automation to send log data to CloudWatch Logs.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iam:SimulateCustomPolicy",
        "iam:GetContextKeysForCustomPolicy",
        "s3:ListAllMyBuckets",
        "logs:CreateLogGroup",
        "logs:CreateLogStream",

```

```
        "logs:PutLogEvents",
        "logs:PutRetentionPolicy",
        "s3:GetAccountPublicAccessBlock"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:GetObject",
      "s3:GetObjectAcl",
      "s3:GetObjectTagging"
    ],
    "Resource": "arn:aws:s3:::awsexamplebucket1/*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:ListBucket",
      "s3:GetBucketLocation",
      "s3:GetBucketPublicAccessBlock",
      "s3:GetBucketRequestPayment",
      "s3:GetBucketPolicyStatus",
      "s3:GetBucketPolicy",
      "s3:GetBucketAcl"
    ],
    "Resource": "arn:aws:s3:::awsexamplebucket1",
    "Effect": "Allow"
  }
]
}
```

## Document Steps

- `aws:assertAwsResourceProperty` - Confirms the S3 bucket exists, and is accessible.
- `aws:executeScript` - Returns the S3 bucket location and your canonical user ID.
- `aws:executeScript` - Returns the public access block settings for your account and the S3 bucket.
- `aws:assertAwsResourceProperty` - Confirms the S3 bucket payer is set to `BucketOwner`. If `Requester Pays` is enabled on the S3 bucket, the automation ends.
- `aws:executeScript` - Returns the S3 bucket policy status and determines whether it is considered public. For more information about public S3 buckets, see [The meaning of "public" in the Amazon Simple Storage Service User Guide](#).
- `aws:executeAwsApi` - Returns the S3 bucket policy.
- `aws:executeAwsApi` - Returns all context keys found in the S3 bucket policy.
- `aws:assertAwsResourceProperty` - Confirms whether there is an explicit deny in the S3 bucket policy for the `GetObject` API action.
- `aws:executeAwsApi` - Returns the access control list (ACL) for the S3 bucket.
- `aws:executeScript` - Creates a CloudWatch Logs log group and log stream if you specify a value for the `CloudWatchLogGroupName` parameter.
- `aws:executeScript` - Based on the values you specify in the runbook input parameters, evaluates whether any of the S3 bucket settings gathered during the automation are preventing objects from being accessed by the public. This script performs the following functions:
  - Evaluates public access block settings
  - Returns objects from your S3 bucket based on the values you specify in the `MaxObjects`, `S3PrefixName`, and `StartAfter` parameters.
  - Returns the S3 bucket policy to simulate a custom IAM policy for the objects returned from your S3 bucket.

- Performs a partial HTTP request to the returned objects if the `HttpGet` parameter is set to `true`. Only the first byte of the object is returned using the `Range` HTTP header.
- Checks the returned object's key name to confirm whether it ends with one or two periods. Object key names that end in periods can't be downloaded from the Amazon S3 console.
- Checks whether the returned object's owner matches the owner of the S3 bucket.
- Checks whether the object's ACL grants `READ` or `FULL_CONTROL` permissions to anonymous users.
- Returns tags associated with the object.
- Uses the simulated IAM policy to confirm whether there is an explicit deny for this object in the S3 bucket policy for the `GetObject` API action.
- Returns the object's metadata to confirm that the storage class is supported.
- Checks the object's server-side encryption settings to confirm whether the object is encrypted using a AWS Key Management Service (AWS KMS) customer managed key.

### Outputs

AnalyzeObjects.bucket

AnalyzeObjects.object

## Secrets Manager

AWS Systems Manager Automation provides predefined runbooks for AWS Secrets Manager. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWSConfigRemediation-DeleteSecret \(p. 262\)](#)
- [AWSConfigRemediation-RotateSecret \(p. 263\)](#)

## AWSConfigRemediation-DeleteSecret

### Description

The `AWSConfigRemediation-RotateSecret` runbook deletes a secret and all of the versions stored in AWS Secrets Manager. You can optionally specify the recovery window during which you can restore the secret. If you don't specify a value for the `RecoveryWindowInDays` parameter, the operation defaults to 30 days.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=, @/ -]+\]\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- RecoveryWindowInDays

Type: Integer

Valid values: 7-30

Default: 30

Description: (Optional) The number of days which you can restore the secret.

- SecretId

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the secret you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `secretsmanager:DeleteSecret`
- `secretsmanager:DescribeSecret`

### Document Steps

- `aws:executeAwsApi` - Deletes the secret you specify in the SecretId parameter.
- `aws:executeScript` - Verifies the secret has been scheduled for deletion.

## AWSConfigRemediation-RotateSecret

### Description

The AWSConfigRemediation-RotateSecret runbook rotates a secret stored in AWS Secrets Manager.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

#### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- RotationInterval

Type: Interval

Valid values: 1-365

Description: (Required) The number of days between rotations of the secret.

- RotationLambdaArn

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the AWS Lambda function that can rotate the secret.

- SecretId

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the secret you want to rotate.

#### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `lambda:InvokeFunction`
- `secretsmanager:DescribeSecret`
- `secretsmanager:RotateSecret`

#### Document Steps

- `aws:executeAwsApi` - Rotates the secret you specify in the `SecretId` parameter.
- `aws:executeScript` - Verifies rotation has been enabled on the secret.

## Security Hub

AWS Systems Manager Automation provides predefined runbooks for AWS Security Hub. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

#### Topics

- [AWSConfigRemediation-EnableSecurityHub \(p. 265\)](#)

## AWSConfigRemediation-EnableSecurityHub

### Description

The `AWSConfigRemediation-EnableSecurityHub` runbook enables AWS Security Hub (Security Hub) for the AWS account and AWS Region where you run the automation. For information about Security Hub, see [What is AWS Security Hub?](#) in the *AWS Security Hub User Guide*.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `EnableDefaultStandards`

Type: Boolean

Default: `true`

Description: (Required) If set to `true`, the default security standards designated by Security Hub are enabled.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `securityhub:DescribeHub`
- `securityhub:EnableSecurityHub`
- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

### Document Steps

- `aws:executeAwsApi` - Enables Security Hub in the current account and Region.



- `aws:executeAwsApi` - Verifies that Security Hub has been enabled.

## Amazon SNS

AWS Systems Manager Automation provides predefined runbooks for Amazon Simple Notification Service. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-EncryptSNSTopic \(p. 266\)](#)
- [AWS-PublishSNSNotification \(p. 267\)](#)

## AWSConfigRemediation-EncryptSNSTopic

### Description

The `AWSConfigRemediation-EncryptSNSTopic` runbook enables encryption on the Amazon Simple Notification Service (Amazon SNS) topic you specify using an AWS Key Management Service (AWS KMS) customer managed key. This runbook should only be used as a baseline to ensure that your Amazon SNS topics are encrypted according to minimum recommended security best practices. We recommend encrypting multiple topics with different customer managed keys.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `KmsKeyArn`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the AWS KMS customer managed key you want to use to encrypt the Amazon SNS topic.

- `TopicArn`

Type: String

Description: (Required) The ARN of the Amazon SNS topic you want to encrypt.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `sns:GetTopicAttributes`
- `sns:SetTopicAttributes`

### Document Steps

- `aws:executeAwsApi` - Encrypts the Amazon SNS topic you specify in the `TopicArn` parameter.
- `aws:assertAwsResourceProperty` - Confirms encryption is enabled on the Amazon SNS topic.

## AWS-PublishSNSNotification

### Description

Publish a notification to Amazon SNS.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `Message`

Type: String

Description: (Required) The message to include in the SNS notification.

- `TopicArn`

Type: String

Description: (Required) The ARN of the SNS topic to publish the notification to.

## Systems Manager

AWS Systems Manager Automation provides predefined runbooks for Systems Manager. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWS-BulkEditOpsItems \(p. 268\)](#)
- [AWS-BulkResolveOpsItems \(p. 270\)](#)
- [AWS-CreateManagedLinuxInstance \(p. 272\)](#)
- [AWS-CreateManagedWindowsInstance \(p. 273\)](#)
- [AWSConfigRemediation-EnableCWLoggingForSessionManager \(p. 275\)](#)
- [AWS-ExportOpsDataToS3 \(p. 276\)](#)
- [AWS-ExportPatchReportToS3 \(p. 277\)](#)
- [AWS-SetupInventory \(p. 279\)](#)
- [AWS-SetupManagedInstance \(p. 281\)](#)
- [AWS-SetupManagedRoleOnEC2Instance \(p. 282\)](#)
- [AWSSupport-TroubleshootManagedInstance \(p. 283\)](#)

## AWS-BulkEditOpsItems

### Description

The `AWS-BulkEditOpsItems` runbook helps you edit the status, severity, category, or priority of AWS Systems Manager OpsItems. This automation can edit a maximum of 50 OpsItems at a time.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your

behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Category

Type: String

Valid values:

- Availability
- Cost
- No change
- Performance
- Recovery
- Security

Default: No change

Description: (Optional) The new category you want to specify for the edited OpsItems.

- OpsItemIds

Type: StringList

Description: (Required) A comma-separated list of OpsItems IDs you want to edit (for example, oi-XXXXXXXXXXXX,oi-XXXXXXXXXXXX).

- Priority

Type: String

Valid values:

- No change
- 1
- 2
- 3
- 4
- 5

Default: No change

Description: (Optional) The importance of the edited OpsItems in relation to other OpsItems in the system.

- Severity

Type: String

Valid values:

- No change
- 1
- 2
- 3
- 4

Default: No change

Description: (Optional) The severity of the edited OpsItems.

- WaitTimeBetweenEditsInSecs

Type: String

Valid values: 0.0-2.0

Default: 0.8

Description: (Optional) The time the automation waits between calling the `UpdateOpsItems` operation.

- Status

Type: String

Valid values:

- InProgress
- No change
- Open
- Resolved

Default: No change

Description: (Optional) The new status of the edited OpsItems.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:GetAutomationExecution`
- `ssm:StartAutomationExecution`
- `ssm:UpdateOpsItem`

### Document Steps

- `aws:executeScript` - Edits the OpsItems you specified in the `OpsItemIds` parameter based on the values you specify for the `Category`, `Priority`, `Severity`, and `Status` parameters.

## AWS-BulkResolveOpsItems

### Description

The `AWS-BulkResolveOpsItems` runbook resolves AWS Systems Manager OpsItems that match the filter you specify. You can also specify an `OpsItemId` to add to the resolved OpsItems using the `OpsInsightsId` parameter. If you specify a value for the `S3BucketName` parameter, a result summary is sent to the Amazon Simple Storage Service (Amazon S3) bucket. To receive a notification once the result summary has been sent to the Amazon S3 bucket, specify a value for the `SnsTopicArn` parameter. This automation will resolve a maximum of 1,000 OpsItems at a time.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Filters

Type: String

Description: (Required) The key-value pairs of filters to return the OpsItems you want to resolve. For example, [{"Key": "Status", "Values": ["Open"], "Operator": "Equals"}]. To learn more about the options available for filtering OpsItems responses, see [OpsItemFilters](#) in the *AWS Systems Manager API Reference*.

- OpsInsightId

Type: String

Description: (Optional) The related resource identifier you want to add to resolved OpsItems.

- S3BucketName

Type: String

Description: (Optional) The name of the Amazon S3 bucket you want to send the result summary to.

- SnsMessage

Type: String

Description: (Optional) The notification you want Amazon Simple Notification Service (Amazon SNS) to send when the automation completes.

- SnsTopicArn

Type: String

Description: (Optional) The ARN of the Amazon SNS topic you want to notify when the result summary has been sent to Amazon S3.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- s3:GetBucketAcl
- s3:PutObject
- sns:Publish
- ssm:DescribeOpsItems
- ssm:GetAutomationExecution
- ssm:StartAutomationExecution
- ssm:UpdateOpsItem

### Document Steps

- `aws:executeScript` - Gathers and resolves the OpsItems based on the filters you specify. If you specified a value for the `OpsInsightId` parameter, the value is added as a related resource.
- `aws:executeScript` - If you specified a value for the `S3BucketName` parameter, a result summary is then sent to the Amazon S3 bucket.
- `aws:executeScript` - If you specified a value for the `SnsTopicArn` parameter, a notification is sent to the Amazon SNS topic after the result summary has been sent to Amazon S3 including the `SnsMessage` parameter value if specified.

## AWS-CreateManagedLinuxInstance

### Description

Create an EC2 instance for Linux that is configured for Systems Manager.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux

### Parameters

- `Amild`

Type: String

Description: (Required) AMI ID to use for launching the instance.

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `GroupName`

Type: String

Default: `SSMSecurityGroupForLinuxInstances`

Description: (Required) Security group name to create.

- `InstanceType`

Type: String

Default: `t2.medium`

Description: (Required) Type of instance to launch. Default is t2.medium.

- KeyPairName

Type: String

Description: (Required) Key pair to use when creating instance.

- RemoteAccessCidr

Type: String

Default: 0.0.0.0/0

Description: (Required) Creates Security group with port for SSH(Port range 22) open to IPs specified by CIDR (default is 0.0.0.0/0). If the security group already exists it will not be modified and rules will not be changed.

- RoleName

Type: String

Default: SSManagedInstanceProfileRole

Description: (Required) Role name to create.

- StackName

Type: String

Default: CreateManagedInstanceStack{{automation:EXECUTION\_ID}}

Description: (Optional) Specify stack name used by this runbook

- SubnetId

Type: String

Default: Default

Description: (Required) New instance will be deployed into this subnet or in the default subnet if not specified.

- VpcId

Type: String

Default: Default

Description: (Required) New instance will be deployed into this Amazon Virtual Private Cloud (Amazon VPC) or in the default Amazon VPC if not specified.

## AWS-CreateManagedWindowsInstance

### Description

Create an EC2 instance for a Windows Server that is configured for Systems Manager.

[Run this Automation \(console\)](#)

### Document type

Automation



**Owner**

Amazon

**Platforms**

Windows

**Parameters**

**Parameters**

- Amild

Type: String

Default: `{{ssm:/aws/service/ami-windows-latest/Windows_Server-2016-English-Full-Base}}`

Description: (Required) AMI ID to use for launching the instance.

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- GroupName

Type: String

Default: `SSMSecurityGroupForLinuxInstances`

Description: (Required) Security group name to create.

- InstanceType

Type: String

Default: `t2.medium`

Description: (Required) Type of instance to launch. Default is `t2.medium`.

- KeyPairName

Type: String

Description: (Required) Key pair to use when creating instance.

- RemoteAccessCidr

Type: String

Default: `0.0.0.0/0`

Description: (Required) Creates security group with port for RDP (Port range 3389) open to IPs specified by CIDR (default is `0.0.0.0/0`). If the security group already exists it will not be modified and rules will not be changed.

- RoleName

Type: String

Default: SSMMangedInstanceProfileRole

Description: (Required) Role name to create.

- StackName

Type: String

Default: CreateManagedInstanceStack{{automation:EXECUTION\_ID}}

Description: (Optional) Specify stack name used by this runbook

- SubnetId

Type: String

Default: Default

Description: (Required) New instance will be deployed into this subnet or in the default subnet if not specified.

- VpcId

Type: String

Default: Default

Description: (Required) New instance will be deployed into this Amazon Virtual Private Cloud (Amazon VPC) or in the default Amazon VPC if not specified.

## AWSConfigRemediation- EnableCWLoggingForSessionManager

### Description

The `AWSConfigRemediation-EnableCWLoggingForSessionManager` runbook enables AWS Systems Manager Session Manager (Session Manager) sessions to store output logs to an Amazon CloudWatch (CloudWatch) log group.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- DestinationLogGroup

Type: String

Description: (Required) The name of the CloudWatch log group.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ssm:GetDocument`
- `ssm:UpdateDocument`
- `ssm:CreateDocument`
- `ssm:UpdateDefaultDocumentVersion`
- `ssm:DescribeDocument`

### Document Steps

- `aws:executeScript` - Accepts the CloudWatch log group to update the document which stores Session Manager session output logs preferences, or creates one if it doesn't exist.

## AWS-ExportOpsDataToS3

### Description

This runbook retrieves a list of OpsData summaries in AWS Systems Manager Explorer and exports them to an object in a specified Amazon Simple Storage Service (Amazon S3) bucket.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your

behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- columnFields

Type: StringList

Description: (Required) Column fields to write to the output file.

- filters

Type: String

Description: (Optional) Filters for the getOpsSummary request.

- resultAttribute

Type: String

Description: (Optional) The result attribute for getOpsSummary request.

- s3BucketName

Type: String

Description: (Required) S3 bucket where you want to download the output file.

- snsSuccessMessage

Type: String

Description: (Optional) Message to send when runbook finishes.

- snsTopicArn

Type: String

Description: (Required) Amazon Simple Notification Service (Amazon SNS) topic ARN to notify when the download completes.

- syncName

Type: String

Description: (Optional) The name of the resource data sync.

### Document Steps

getOpsSummaryStep – Retrieves up to 5,000 ops summaries to export in a CSV file now.

### Outputs

OpsData object – If the runbook runs successfully, you will find the exported OpsData object in your target S3 bucket.

## AWS-ExportPatchReportToS3

### Description

This runbook retrieves lists of patch summary data and patch details in AWS Systems Manager Patch Manager and exports them to .csv files in a specified Amazon Simple Storage Service (Amazon S3) bucket.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- `assumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that runs this document.

- `s3BucketName`

Type: String

Description: (Required) The S3 bucket where you want to download the output file.

- `snsTopicArn`

Type: String

Description: (Optional) The Amazon Simple Notification Service (Amazon SNS) topic Amazon Resource Name (ARN) to notify when the download completes.

- `snsSuccessMessage`

Type: String

Description: (Optional) Text of the message to send when the runbook finishes.

- `targets`

Type: String

Description: (Required) The instance ID or a wildcard character (\*) to indicate whether to report patch data for a specific instance or for all instances.

## Document Steps

`ExportReportStep` – The action for this step depends on the value of the `targets` parameter. If `targets` is in the format of `instanceids=*`, the step retrieves up to 10,000 patch summaries for instances in your account and exports the data to a .csv file.

If `targets` is in the format `instanceids=<instance-id>`, the step retrieves both the patch summary and all the patches for the specified instance in your account and exports them to a .csv file.

## Outputs

`PatchSummary/Patches` object – If the runbook runs successfully, the exported patch report object is downloaded to your target S3 bucket.

## AWS-SetupInventory

### Description

Create a Systems Manager Inventory association for one or more managed instances. The system collects metadata from your instances according to the schedule in the association. For more information, see [AWS Systems Manager Inventory](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- Applications
  - Type: String
  - Default: Enabled
  - Description: (Optional) Collect metadata about installed applications.
- AssociatedDocName
  - Type: String
  - Default: AWS-GatherSoftwareInventory
  - Description: (Optional) The name of the runbook used to collect Inventory from the managed instance.
- AssociationName
  - Type: String
  - Description: (Optional) A name for the Inventory association that will be assigned to the instance.
- AssocWaitTime
  - Type: String
  - Default: PT5M
  - Description: (Optional) Amount of time that Inventory collection should pause when the Inventory association start time is reached. The time uses ISO 8601 format.
- AutomationAssumeRole
  - Type: String
  - Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.
- AwsComponents

Type: String

Default: Enabled

Description: (Optional) Collect metadata for AWS Components like amazon-ssm-agent.

- CustomInventory

Type: String

Default: Enabled

Description: (Optional) Collect custom inventory metadata.

- Files

Type: String

Description: (Optional) Collect metadata about files on your instances. For more information about how to collect this type of Inventory data, see [Working with file and Windows registry inventory](#) . Requires SSMAgent version 2.2.64.0 or later. Linux example: [{"Path":"/usr/bin", "Pattern":["aws\*", "\*ssm\*"],"Recursive":false}, {"Path":"/var/log", "Pattern":["amazon\*.\*"], "Recursive":true, "DirScanLimit":1000}] Windows example: [{"Path": "%PROGRAMFILES%", "Pattern": ["\*.exe"], "Recursive": true}]

- InstanceDetailedInformation

Type: String

Default: Enabled

Description: (Optional) Collect additional information about the instance, including the CPU model, speed, and the number of cores, to name a few.

- InstanceIds

Type: String

Default: \*

Description: (Required) EC2 instances that you want to inventory.

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

- NetworkConfig

Type: String

Default: Enabled

Description: (Optional) Collect metadata about network configurations.

- OutputS3BucketName

Type: String

Description: (Optional) Name of an Amazon S3 bucket where you want to write Inventory log data.

- OutputS3KeyPrefix

Type: String

Description: (Optional) An Amazon S3 key prefix (subfolder) where you want to write Inventory log data.

- OutputS3Region

Type: String

Description: (Optional) The name of the AWS Region where the Amazon S3 exists.

- Schedule

Type: String

Default: cron(0 \*/30 \* \* \* ? \*)

Description: (Optional) A cron expression for the Inventory association schedule. The default is every 30 minutes.

- Services

Type: String

Default: Enabled

Description: (Optional, Windows OS only, requires SSMAgent version 2.2.64.0 and above) Collect data for service configurations.

- WindowsRegistry

Type: String

Description: (Optional) Collect metadata about Microsoft Windows Registry keys. For more information about how to collect this type of Inventory data, see [Working with file and Windows registry inventory](#) . Requires SSM Agent version 2.2.64.0 or later. Example: [ {"Path":"HKEY\_CURRENT\_CONFIG\System","Recursive":true}, {"Path":"HKEY\_LOCAL\_MACHINE\SOFTWARE\Amazon\MachineImage","ValueNames":["AMIName"]} ]

- WindowsRoles

Type: String

Default: Enabled

Description: (Optional) Collect information about Windows roles on the instance. Applies to Windows operating systems only. Requires SSMAgent version 2.2.64.0 or later.

- WindowsUpdates

Type: String

Default: Enabled

Description: (Optional) Collect data about all Windows Updates on the instance.

## AWS-SetupManagedInstance

### Description

Configure an instance with an AWS Identity and Access Management (IAM) role for Systems Manager access.

[Run this Automation \(console\)](#)



### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) ID of the EC2 instance to configure

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

- RoleName

Type: String

Default: SSMRoleForManagedInstance

Description: (Optional) The name of the IAM role for the EC2 instance. If this role does not exist, it will be created. When specifying this value, verify that the role contains the **AmazonSSMManagedInstanceCore** Managed Policy.

## AWS-SetupManagedRoleOnEC2Instance

### Description

Configure an instance with the SSMRoleForManagedInstance managed IAM role for Systems Manager access.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) ID of the EC2 instance to configure

- LambdaAssumeRole

Type: String

Description: (Optional) The ARN of the role that allows Lambda created by Automation to perform the actions on your behalf. If not specified a transient role will be created to run the Lambda function.

- RoleName

Type: String

Default: SSMRoleForManagedInstance

Description: (Optional) The name of the IAM role for the EC2 instance. If this role does not exist, it will be created. When specifying this value, verify that the role contains the **AmazonSSMManagedInstanceCore** Managed Policy.

# AWSSupport-TroubleshootManagedInstance

## Description

The `AWSSupport-TroubleshootManagedInstance` runbook helps you to determine why an Amazon Elastic Compute Cloud (Amazon EC2) instance does not report as managed by AWS Systems Manager. This runbook reviews the VPC configuration for the instance including security group rules, VPC endpoints, network access control list (ACL) rules, and route tables. It also confirms an AWS Identity and Access Management (IAM) instance profile that contains the required permissions is attached to the instance.

[Run this Automation \(console\)](#)

## Document type

Automation

## Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The ID of the Amazon EC2 instance that is not reporting as managed by Systems Manager.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:DescribeAutomationExecutions
- ssm:DescribeAutomationStepExecutions
- ssm:DescribeAutomationStepExecutions
- ssm:DescribeInstanceInformation
- ssm:DescribeInstanceProperties
- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ssm:GetDocument
- ssm:ListDocuments
- ssm:StartAutomationExecution
- iam:ListRoles
- iam:GetInstanceProfile
- iam:ListAttachedRolePolicies
- ec2:DescribeInstances
- ec2:DescribeNetworkAcls
- ec2:DescribeRouteTables
- ec2:DescribeSecurityGroups
- ec2:DescribeVpcEndpoints

## Document Steps

- aws:executeScript - Gathers the PingStatus of the instance.
- aws:branch - Branches based on whether the instance is already reporting as managed by Systems Manager.
- aws:executeAwsApi - Gathers details about the instance including the VPC configuration.
- aws:executeScript - If applicable, gathers additional details related to VPC endpoints that have been deployed to use with Systems Manager, and confirms the security groups attached to the VPC endpoint allow inbound traffic on TCP port 443 from the instance.
- aws:executeScript - Checks whether the route table allows traffic to the VPC endpoint or public Systems Manager endpoints.

- `aws:executeScript` - Checks whether the network ACL rules allow traffic to the VPC endpoint or public Systems Manager endpoints.
- `aws:executeScript` - Checks whether outbound traffic to the VPC endpoint or public Systems Manager endpoints is allowed by the security group associated with the instance.
- `aws:executeScript` - Checks if the instance profile attached to the instance includes a managed policy that provides the required permissions.
- `aws:branch` - Branches based on the operating system of the instance.
- `aws:executeScript` - Provides reference to `ssmagent-toolkit-linux` shell script.
- `aws:executeScript` - Provides reference to `ssmagent-toolkit-windows` PowerShell script.
- `aws:executeScript` - Generates final output for the automation.
- `aws:executeScript` - If the `PingStatus` of the instance is `Online`, returns that the instance is already managed by Systems Manager.

## Third-party

AWS Systems Manager Automation provides predefined runbooks for third-party products and services. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWS-CreateJiraIssue \(p. 285\)](#)
- [AWS-CreateServiceNowIncident \(p. 287\)](#)
- [AWS-RunPacker \(p. 288\)](#)

## AWS-CreateJiraIssue

### Description

Create an issue in Jira.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AssigneeName`

Type: String

Description: (Optional) The username of the person the issue should be assigned to.

- `DueDate`

Type: String

Description: (Optional) The due date for the issue in yyyy-mm-dd format.

- IssueDescription

Type: String

Description: (Required) A detailed description of the issue.

- IssueSummary

Type: String

Description: (Required) A brief summary of the issue.

- IssueTypeName

Type: String

Description: (Required) The name of the type of issue you want to create (for example, Task, Sub-task, Bug, etc.).

- JiraURL

Type: String

Description: (Required) The url of the Jira instance.

- JiraUsername

Type: String

Description: (Required) The name of the user the issue will be created with.

- PriorityName

Type: String

Description: (Optional) The name of the priority of the issue.

- ProjectKey

Type: String

Description: (Required) The key of the project the issue should be created in.

- SSMPparameterName

Type: String

Description: (Required) The name of an encrypted SSM Parameter containing the API key or password for the Jira user.

### Document Steps

`aws:createStack` - Create CloudFormation stack to create Lambda IAM role and function.

`aws:invokeLambdaFunction` - Invoke Lambda function to create the Jira issue

`aws:deleteStack` - Delete the CloudFormation stack created.

### Outputs

Issued: ID of the newly created Jira issue

## AWS-CreateServiceNowIncident

### Description

Create an incident in the ServiceNow incident table.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Category

Type: String

Description: (Optional) The category of the incident.

Valid values: None | Inquiry/Help | Software | Hardware | Network | Database

Default Value: None

- Description

Type: String

Description: (Required) A detailed explanation on the incident.

- Impact

Type: String

Description: (Optional) The effect an incident has on business.

Valid values: High | Medium | Low

Default Value: Low

- ServiceNowInstanceUsername

Type: String

Description: (Required) The name of the user the incident will be created with.

- ServiceNowInstancePassword

Type: String

Description: (Required) The name of an encrypted SSM Parameter containing the password for the ServiceNow user.

- ServiceNowInstanceURL

Type: String

Description: (Required) The URL of the ServiceNow instance

- ShortDescription

Type: String

Description: (Required) A brief description of the incident.

- Subcategory

Type: String

Description: (Optional) The subcategory of the incident.

Valid values: None | Antivirus | Email | Internal Application | Operating System | CPU | Disk | Keyboard | Hardware | Memory | Monitor | Mouse | DHCP | DNS | IP Address | VPN | Wireless | DB2 | MS SQL Server | Oracle

Default Value: None

### Document Steps

Push\_incident – Pushes the incident information to ServiceNow.

### Outputs

Push\_incident.incidentID – The created incident ID.

## AWS-RunPacker

### Description

This runbook uses the HashiCorp [Packer](#) tool to validate, fix, or build packer templates that are used to create machine images. This runbook uses Packer v1.7.2.

#### Note

If you specify a `vpc_id` value, you must also specify the `subnet_id` value of a public subnet. Unless you modify your subnet's IPv4 public addressing attribute, you must also set `associate_public_ip_address` to `true`.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Force

Type: Boolean

Description: A Packer option to force a builder to run when artifacts from a previous build otherwise prevent a build from running.

- Mode

Type: String

Description: The mode, or command, in which to use Packer when validating against the template. Options include `Build`, `Validate`, and `Fix`.

- TemplateFileName

Type: String

Description: The name, or key, of the template file in the S3 bucket.

- TemplateS3BucketName

Type: String

Description: The name of the S3 bucket containing the packer template.

### Document Steps

RunPackerProcessTemplate – Runs the selected mode against the template using the Packer tool.

### Outputs

RunPackerProcessTemplate.output – The stdout from the Packer tool.

RunPackerProcessTemplate.fixed\_template\_key – The name of the template stored in an S3 bucket to use only when running in "Fix" mode.

RunPackerProcessTemplate.s3\_bucket – The name of the S3 bucket that contains the fixed template to use only when running in "Fix" mode.

## Amazon VPC

AWS Systems Manager Automation provides predefined runbooks for Amazon Virtual Private Cloud. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics



- [AWSSupport-ConfigureDNSQueryLogging](#) (p. 290)
- [AWSSupport-ConfigureTrafficMirroring](#) (p. 292)
- [AWSSupport-ConnectivityTroubleshooter](#) (p. 294)
- [AWSConfigRemediation-DeleteEgressOnlyInternetGateway](#) (p. 296)
- [AWSConfigRemediation-DeleteUnusedENI](#) (p. 297)
- [AWSConfigRemediation-DeleteUnusedSecurityGroup](#) (p. 298)
- [AWSConfigRemediation-DeleteUnusedVPCNetworkACL](#) (p. 299)
- [AWSConfigRemediation-DeleteVPCFlowLog](#) (p. 300)
- [AWSConfigRemediation-DetachAndDeleteInternetGateway](#) (p. 301)
- [AWSConfigRemediation-DetachAndDeleteVirtualPrivateGateway](#) (p. 302)
- [AWS-DisablePublicAccessForSecurityGroup](#) (p. 304)
- [AWSConfigRemediation-DisableSubnetAutoAssignPublicIP](#) (p. 305)
- [AWSSupport-EnableVPCFlowLogs](#) (p. 306)
- [AWSConfigRemediation-EnableVPCFlowLogsToCloudWatch](#) (p. 308)
- [AWSConfigRemediation-EnableVPCFlowLogsToS3Bucket](#) (p. 310)
- [AWS-ReleaseElasticIP](#) (p. 311)
- [AWSConfigRemediation-RemoveUnrestrictedSourceIngressRules](#) (p. 312)
- [AWSConfigRemediation-RemoveVPCDefaultSecurityGroupRules](#) (p. 313)
- [AWSSupport-SetupIPMonitoringFromVPC](#) (p. 314)
- [AWSSupport-TerminateIPMonitoringFromVPC](#) (p. 321)

## AWSSupport-ConfigureDNSQueryLogging

### Description

The `AWSSupport-ConfigureDNSQueryLogging` runbook configures logging for DNS queries that originate in your virtual private cloud (VPC) or for Amazon Route 53 hosted zones. You can choose to publish query logs to Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3), or Amazon Kinesis Data Firehose. For more information about query logging and resolver query logs, see [Public DNS query logging](#) and [Resolver query logging](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your

behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `LogDestinationArn`

Type: String

Description: (Optional) The ARN of the CloudWatch Logs group, Amazon S3 bucket or Kinesis Data Firehose stream you want to send query logs to. Note that Route 53 public DNS query logging only supports CloudWatch Logs groups. If you do not specify a value for this parameter, the automation creates a CloudWatch Logs group with the format `AWSSupport-ConfigureDNSQueryLogging-{automation: EXECUTION_ID }`, and an IAM resource policy to publish the query logs. The CloudWatch Logs group created by the automation has a retention period of 14 days.

- `QueryLogType`

Type: String

Description: (Optional) The types of queries you want to log.

Valid values: `Public` | `Resolver/Private`

Default: `Public`

- `ResourceId`

Type: String

Description: (Required) The ID of the resource whose queries you want to log. If you specify `Public` for the `QueryLogType` parameter, the resource must be the ID of a Route 53 private hosted zone. If you specify `Resolver/Private` for the `QueryLogType` parameter, the resource must be the ID of a VPC.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ec2:DescribeVpcs`
- `firehose:ListTagsForDeliveryStream`
- `firehose:PutRecord`
- `firehose:PutRecordBatch`
- `firehose:TagDeliveryStream`
- `iam:AttachRolePolicy`
- `iam:CreatePolicy`
- `iam:CreateRole`
- `iam:CreateServiceLinkedRole`
- `iam>DeletePolicy`
- `iam>DeleteRole`
- `iam>DeleteRolePolicy`
- `iam:GetPolicy`
- `iam:GetRole`
- `iam:PassRole`
- `iam:PutRolePolicy`
- `iam:TagRole`
- `iam:UpdateRole`
- `logs:CreateLogDelivery`

- logs:CreateLogGroup
- logs>DeleteLogDelivery
- logs>DeleteLogGroup
- logs:DescribeLogGroups
- logs:DescribeLogStreams
- logs:DescribeResourcePolicies
- logs>ListLogDeliveries
- logs:PutResourcePolicy
- logs:PutRetentionPolicy
- logs:UpdateLogDelivery
- route53:CreateQueryLoggingConfig
- route53>DeleteQueryLoggingConfig
- route53:GetHostedZone
- route53resolver:AssociateResolverQueryLogConfig
- route53resolver:CreateResolverQueryLogConfig
- route53resolver>DeleteResolverQueryLogConfig
- s3:GetBucketAcl

### Document Steps

- aws:executeScript - Verifies the resource you specify for the ResourceId parameter exists, and checks whether the resource type matches the required QueryLogType option.
- aws:executeScript - Verifies that the value you specify for the LogDestinationArn parameter matches the required QueryLogType .
- aws:executeScript - Verifies the required permissions for Route 53 to publish logs to the CloudWatch Logs log group, and creates the required IAM resource policy if it doesn't exist.
- aws:executeScript - Enables the DNS query logging on the selected destination.

## AWSSupport-ConfigureTrafficMirroring

### Description

The AWSSupport-ConfigureTrafficMirroring runbook configures traffic mirroring to help you troubleshoot connectivity issues between a load balancer and Amazon Elastic Compute Cloud (Amazon EC2) instances. Traffic mirroring copies inbound and outbound traffic from the network interfaces that are attached to your instances. To configure traffic mirroring, this runbook creates the required targets, filters, and sessions. By default, the runbook configures mirroring for all inbound and outbound traffic for all protocols except Amazon DNS. If you want to mirror traffic from specific sources and destinations, you can modify the inbound and outbound rules after the automation completes.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([w+=,.-]+)\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- SourceENI

Type: String

Description: (Required) The elastic network interface you want to configure traffic mirroring for.

- Target

Type: String

Description: (Required) The destination for the mirrored traffic. You must specify the ID of a network interface, a Network Load Balancer, or a Gateway Load Balancer endpoint. If you specify a Network Load Balancer, there must be UDP listeners on port 4789.

- SessionNumber

Type: String

Valid values: 1-32766

Description: (Required) The number of the mirror session you want to use.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ec2:CreateTrafficMirrorTarget
- ec2:CreateTrafficMirrorFilter
- ec2:CreateTrafficMirrorFilterRule
- ec2:CreateTrafficMirrorSession
- ec2>DeleteTrafficMirrorSession
- ec2>DeleteTrafficMirrorFilter
- ec2>DeleteTrafficMirrorSession
- ec2>DeleteTrafficMirrorFilterRule
- iam:ListRoles
- ssm:GetAutomationExecution
- ssm:StartAutomationExecution

### Document Steps

- aws:executeScript - Runs a script to create a target.
- aws:executeAwsApi - Creates a filter rule.
- aws:executeAwsApi - Creates a mirror filter rule for all inbound traffic.

- `aws:executeAwsApi` - Creates a mirror filter rule for all outbound traffic.
- `aws:executeAwsApi` - Creates a traffic mirror session.
- `aws:executeAwsApi` - Deletes the filter if filter or session creation fails.
- `aws:executeAwsApi` - Deletes the target if filter or session creation fails.

### Outputs

CreateFilter.FilterId

CreateSession.SessionId

CreateTarget.TargetIDOutput

## AWSSupport-ConnectivityTroubleshooter

### Description

The `AWSSupport-ConnectivityTroubleshooter` runbook diagnoses connectivity issues between the following:

- AWS resources within an Amazon Virtual Private Cloud (Amazon VPC)
- AWS resources in different Amazon VPCs within the same AWS Region that are connected using VPC peering
- AWS resources in an Amazon VPC and an internet resource using an internet gateway
- AWS resources in an Amazon VPC and an internet resource using a network address translation (NAT) gateway

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- DestinationIP

Type: String

Description: (Required) The IPv4 address of the resource you want to connect to.

- **DestinationPort**  
Type: String  
Default: true  
Description: (Required) The port number you want to connect to on the destination resource.
- **DestinationVpc**  
Type: String  
Default: All  
Description: (Optional) The ID of the Amazon VPC you want to test connectivity to.
- **SourceIP**  
Type: String  
Description: (Required) The private IPv4 address of the AWS resource in your Amazon VPC you want to test connectivity from.
- **SourcePortRange**  
Type: String  
Description: (Optional) The port range used by the AWS resource in your Amazon VPC you want to test connectivity from.
- **SourceVpc**  
Type: String  
Default: All  
Description: (Optional) The ID of the Amazon VPC you want to test connectivity from.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ec2:DescribeNatGateways`
- `ec2:DescribeNetworkAcls`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeVpcPeeringConnections`

### Document Steps

- `aws:executeScript` - Gathers details about the AWS resource you specify in the `SourceIP` parameter.
- `aws:executeScript` - Determines the destination of network traffic from the AWS resource using the routes gathered from the previous step.
- `aws:branch` - Branches based on the destination of the network traffic.
- `aws:executeAwsApi` - Gathers details about the destination resource.
- `aws:executeScript` - Confirms that the ID returned for the destination Amazon VPC matches the value specified, if any, in the `DestinationVpc` parameter.

- `aws:executeAwsApi` - Gathers the security group rules for the source and destination resources.
- `aws:executeScript` - Confirms whether the security group rules allow the needed traffic between the source and destination resources.
- `aws:executeAwsApi` - Gathers the network access control lists (NACLs) associated with the subnets for the source and destination resources.
- `aws:executeScript` - Confirms whether the NACLs allow the needed traffic between the source and destination resources.
- `aws:executeScript` - Confirms whether the source has a public IP address associated with the resource, if the route destination is an internet gateway.
- `aws:executeAwsApi` - Gathers the security group rules for the source resource.
- `aws:executeScript` - Confirms whether the security group rules allow the needed traffic from the source to the destination resource.
- `aws:executeAwsApi` - Gathers the NACLs associated with the subnet for the source resource.
- `aws:executeScript` - Confirms whether the NACLs allow the needed traffic from the source resource.
- `aws:executeAwsApi` - Gathers details about the NAT gateway.
- `aws:executeAwsApi` - Gathers the NACLs associated with the subnet for the NAT gateway.
- `aws:executeScript` - Confirms whether the NACLs allow the needed traffic from the subnet for the NAT gateway.
- `aws:executeScript` - Gathers the routes associated with the subnet for the NAT gateway.
- `aws:executeScript` - Confirms whether the NAT gateway has a route to an internet gateway.
- `aws:executeAwsApi` - Gathers details about the VPC peering connection.
- `aws:executeScript` - Confirms both VPCs are in the same Region and that the ID returned for the destination VPC matches the value specified, if any, in the `DestinationVpc` parameter.
- `aws:executeAwsApi` - Returns the subnet of the destination resource.
- `aws:executeScript` - Gathers the routes associated with the subnet for the peered VPC.
- `aws:executeScript` - Confirms whether the peered VPC has a route to the peering connection.
- `aws:executeScript` - Confirms whether traffic is allowed from the source resource if the destination is not supported by the automation.

## AWSConfigRemediation-DeleteEgressOnlyInternetGateway

### Description

The `AWSConfigRemediation-DeleteEgressOnlyInternetGateway` runbook deletes the egress-only internet gateway you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[w+=, @/ -\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- EgressOnlyInternetGatewayId

Type: String

Description: (Required) The ID of the egress-only internet gateway that you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ec2:DeleteEgressOnlyInternetGateway
- ec2:DescribeEgressOnlyInternetGateways

### Document Steps

- aws:executeScript - Deletes the egress-only internet gateway specified in the EgressOnlyInternetGatewayId parameter.
- aws:executeScript - Verifies the egress-only internet gateway has been deleted.

## AWSConfigRemediation-DeleteUnusedENI

### Description

The AWSConfigRemediation-DeleteUnusedENI runbook deletes an elastic network interface (ENI) that has an attachment status of detached .

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole



Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- NetworkInterfaceId

Type: String

Description: (Required) The ID of the ENI that you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2>DeleteNetworkInterface`
- `ec2:DescribeNetworkInterfaces`

### Document Steps

- `aws:executeAwsApi` - Deletes the ENI you specify in the `NetworkInterfaceId` parameter.
- `aws:executeScript` - Verifies the ENI has been deleted.

## AWSConfigRemediation-DeleteUnusedSecurityGroup

### Description

The `AWSConfigRemediation-DeleteUnusedSecurityGroup` runbook deletes the security group you specify in the `GroupId` parameter. If you attempt to delete a security group that is associated with an Amazon Elastic Compute Cloud (Amazon EC2) instance, or is referenced by another security group, the automation fails. This automation does not delete a default security group.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `GroupId`

Type: String

Description: (Required) The ID of the security group that you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeSecurityGroups`
- `ec2>DeleteSecurityGroup`

### Document Steps

- `aws:executeAwsApi` - Returns the security group name using the value you provide in the `GroupId` parameter.
- `aws:branch` - Confirms that the group name is not "default".
- `aws:executeAwsApi` - Deletes the security group specified in the `GroupId` parameter.
- `aws:executeScript` - Confirms the security group was deleted.

## AWSConfigRemediation-DeleteUnusedVPCNetworkACL

### Description

The `AWSConfigRemediation-DeleteUnusedVPCNetworkACL` runbook deletes a network access control list (ACL) that is not associated with a subnet.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\\w+=,./-]+\\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- NetworkAclId

Type: String

Description: (Required) The ID of the network ACL that you want to delete.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ec2>DeleteNetworkAcl
- ec2:DescribeNetworkAcls

### Document Steps

- aws:executeAwsApi - Deletes the network ACL specified in the NetworkAclId parameter.
- aws:executeScript - Confirms the network ACL specified in the NetworkAclId parameter was deleted.

## AWSConfigRemediation-DeleteVPCFlowLog

### Description

The AWSConfigRemediation-DeleteVPCFlowLog runbook deletes the virtual private cloud (VPC) flow log you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- FlowLogId

Type: String

Description: (Required) The ID of the flow log that you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2>DeleteFlowLogs`
- `ec2:DescribeFlowLogs`

### Document Steps

- `aws:executeAwsApi` - Deletes the flow log you specify in the `FlowLogId` parameter.
- `aws:executeScript` - Verifies the flow log has been deleted.

## AWSConfigRemediation- DetachAndDeleteInternetGateway

### Description

The `AWSConfigRemediation-DetachAndDeleteInternetGateway` runbook detaches and deletes the internet gateway you specify. If any Amazon EC2 instances in your virtual private cloud (VPC) have elastic IP addresses or public IPv4 addresses associated with them, the runbook fails.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `InternetGatewayId`

Type: String

Description: (Required) The ID of the internet gateway that you want to delete.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DeleteInternetGateway`
- `ec2:DescribeInternetGateways`
- `ec2:DetachInternetGateway`

### Document Steps

- `aws:waitForAwsResourceProperty` - Accepts the ID of the virtual private gateway and waits until the virtual private gateway's state property changes to `available` or times out.
- `aws:executeAwsApi` - Retrieves a specified virtual private gateway configuration.
- `aws:branch` - Branches based on the `VpcAttachments.state` parameter value.
  
- `aws:waitForAwsResourceProperty` - Accepts the ID of the virtual private gateway and waits until the virtual private gateway's `VpcAttachments.state`'s property changes to `attached` or times out.
- `aws:executeAwsApi` - Accepts the ID of the virtual private gateway and the ID of the Amazon VPC as input, and detaches the virtual private gateway from the Amazon VPC.
- `aws:waitForAwsResourceProperty` - Accepts the ID of the virtual private gateway and waits until the virtual private gateway's `VpcAttachments.state`'s property changes to `detached` or times out.
  
- `aws:executeAwsApi` - Accepts the ID of the virtual private gateway as input and deletes it.
  
- `aws:waitForAwsResourceProperty` - Accepts the ID of the virtual private gateway as input and verifies its deletion.
  
- `aws:executeAwsApi` - Gathers the VPC ID from the internet gateway ID.
- `aws:executeAwsApi` - Detaches the internet gateway ID from the VPC.
- `aws:executeAwsApi` - Deletes the internet gateway.

## AWSConfigRemediation- DetachAndDeleteVirtualPrivateGateway

### Description

The AWSConfigRemediation-DetachAndDeleteVirtualPrivateGateway runbook detaches and deletes a given Amazon Elastic Compute Cloud (Amazon EC2) virtual private gateway attached to a virtual private cloud (VPC) created with Amazon Virtual Private Cloud (Amazon VPC).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[[\w+=, @/ -]+\]\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- VpnGatewayId

Type: String

Description: (Required) The ID of the virtual private gateway to be deleted.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ec2>DeleteVpnGateway
- ec2:DetachVpnGateway
- ec2:DescribeVpnGateways

### Document Steps

- aws:waitForAwsResourceProperty - Accepts the ID of the virtual private gateway and waits until the virtual private gateway's state property changes to available or times out.
- aws:executeAwsApi - Retrieves a specified virtual private gateway configuration.
- aws:branch - Branches based on the VpcAttachments.state parameter value.
- aws:waitForAwsResourceProperty - Accepts the ID of the virtual private gateway and waits until the virtual private gateway's VpcAttachments.state's property changes to attached or times out.
- aws:executeAwsApi - Accepts the ID of the virtual private gateway and the ID of the Amazon VPC as input, and detaches the virtual private gateway from the Amazon VPC.

- `aws:waitForAwsResourceProperty` - Accepts the ID of the virtual private gateway and waits until the virtual private gateway's `VpcAttachments.state`'s property changes to `detached` or times out.
- `aws:executeAwsApi` - Accepts the ID of the virtual private gateway as input and deletes it.
- `aws:waitForAwsResourceProperty` - Accepts the ID of the virtual private gateway as input and verifies its deletion.

## AWS-DisablePublicAccessForSecurityGroup

### Description

This runbook disables default SSH and RDP ports that are opened to all IP addresses.

#### Important

This runbook fails with an "InvalidPermission.NotFound" error for security groups that meet both of the following criteria: 1) The security group is located in a non-default VPC; and 2) The inbound rules for the security group don't specify open ports using all four of the following patterns:

- `0.0.0.0/0`
- `::/0`
- `SSH or RDP port + 0.0.0.0/0`
- `SSH or RDP port + ::/0`

#### Note

This runbook is not available in the AWS Regions located within China.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `GroupId`

Type: String

Description: (Required) The ID of the security group for which the ports should be disabled.

- IpAddressToBlock

Type: String

Description: (Optional) Additional IPv4 addresses from which access should be blocked, in the format 1.2.3.4/32.

### Outputs

None

## AWSConfigRemediation- DisableSubnetAutoAssignPublicIP

### Description

The AWSConfigRemediation-DisableSubnetAutoAssignPublicIP runbook disables the IPv4 public addressing attribute for the subnet you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[w+=,./-]+\\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- SubnetId

Type: String

Description: (Required) The ID of the subnet that you want to disable the auto-assign public IPv4 address attribute on.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.



- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeSubnets`
- `ec2:ModifySubnetAttribute`

### Document Steps

- `aws:executeAwsApi` - Disables the auto-assign public IPv4 address attribute for the subnet you specified in the `SubnetId` parameter.
- `aws:assertAwsResourceProperty` - Verifies the attribute has been disabled.

## AWSSupport-EnableVPCFlowLogs

### Description

The `AWSSupport-EnableVPCFlowLogs` runbook creates Amazon Virtual Private Cloud (Amazon VPC) Flow Logs for subnets, network interfaces, and VPCs in your AWS account. If you create a flow log for a subnet or VPC, each elastic network interface in that subnet or Amazon VPC is monitored. Flow log data is published to the Amazon CloudWatch Logs log group or the Amazon Simple Storage Service (Amazon S3) bucket you specify. For more information about flow logs, see [VPC Flow Logs](#) in the *Amazon VPC User Guide*.

#### Important

Data ingestion and archival charges for vendored logs apply when you publish flow logs to CloudWatch Logs or to Amazon S3. For more information, see [Flow Logs pricing](#)

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- `DeliverLogsPermissionArn`

Type: String

Description: (Optional) The ARN for the IAM role that permits Amazon Elastic Compute Cloud (Amazon EC2) to publish flow logs to the CloudWatch Logs log group in your account. If you specify `s3` for the

`LogDestinationType` parameter, do not provide a value for this parameter. For more information, see [Publish flow logs to CloudWatch Logs](#) in the *Amazon VPC User Guide* .

- `LogDestinationARN`

Type: String

Description: (Optional) The ARN of the resource to which the flow log data is published. If `cloud-watch-logs` is specified for the `LogDestinationType` parameter, provide the ARN of the CloudWatch Logs log group you want to publish flow log data to. Alternatively, use `LogGroupName` instead. If `s3` is specified for the `LogDestinationType` parameter, you must specify the ARN of the Amazon S3 bucket you want to publish flow log data to for this parameter. You can also specify a folder in the bucket.

- `LogDestinationType`

Type: String

Valid values: `cloud-watch-logs` | `s3`

Description: (Required) Determines where flow log data is published. If you specify `LogDestinationType` as `s3` , do not specify `DeliverLogsPermissionArn` or `LogGroupName` .

- `LogFormat`

Type: String

Description: (Optional) The fields to include in the flow log, and the order in which they should appear in the record. For a list of available fields, see [Flow log records](#) in the *Amazon VPC User Guide* . If you do not provide a value for this parameter, the flow log is created using the default format. If you specify this parameter, you must specify at least one field.

- `LogGroupName`

Type: String

Description: (Optional) The name of the CloudWatch Logs log group where flow log data is published. If you specify `s3` for the `LogDestinationType` parameter, do not provide a value for this parameter.

- `ResourceIds`

Type: StringList

Description: (Required) A comma-separated list of the IDs for the subnets, elastic network interfaces, or VPC for which you want to create a flow log.

- `TrafficType`

Type: String

Valid values: `ACCEPT` | `REJECT` | `ALL`

Description: (Required) The type of traffic to log. You can log traffic that the resource accepts or rejects, or all traffic.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:CreateFlowLogs`
- `ec2>DeleteFlowLogs`

- `ec2:DescribeFlowLogs`
- `iam:AttachRolePolicy`
- `iam:CreateRole`
- `iam:CreatePolicy`
- `iam>DeletePolicy`
- `iam>DeleteRole`
- `iam>DeleteRolePolicy`
- `iam:GetPolicy`
- `iam:GetRole`
- `iam:TagRole`
- `iam:PassRole`
- `iam:PutRolePolicy`
- `iam:UpdateRole`
- `logs:CreateLogDelivery`
- `logs:CreateLogGroup`
- `logs>DeleteLogDelivery`
- `logs>DeleteLogGroup`
- `logs:DescribeLogGroups`
- `logs:DescribeLogStreams`
- `s3:GetBucketAcl`

#### Document Steps

- `aws:branch` - Branches based on the value specified for the `LogDestinationType` parameter.
- `aws:executeScript` - Creates a log group if no value is specified for the `LogDestinationARN` parameter, and `cloud-watch-logs` is specified for the `LogDestinationType` parameter.
- `aws:executeScript` - Creates flow logs based on the values specified in the runbook parameters.

## AWSConfigRemediation- EnableVPCFlowLogsToCloudWatch

#### Description

The `AWSConfigRemediation-EnableVPCFlowLogsToCloudWatch` runbook replaces an existing Amazon VPC flow log that publishes flow log data to Amazon Simple Storage Service (Amazon S3) with a flow log that publishes flow log data to the Amazon CloudWatch Logs (CloudWatch Logs) log group you specify.

[Run this Automation \(console\)](#)

#### Document type

Automation

#### Owner

Amazon

## Platforms

Linux, macOS, Windows

## Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\/[\w+=,./-]+\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- DestinationLogGroup

Type: String

Description: (Required) The name of the CloudWatch Logs log group you want to publish flow log data to.

- DeliverLogsPermissionArn

Type: String

Description: (Required) The ARN of the AWS Identity and Access Management (IAM) role you want to use that provides Amazon Elastic Compute Cloud (Amazon EC2) the requisite permissions to publish flow log data to CloudWatch Logs.

- FlowLogId

Type: String

Description: (Required) The ID of the flow log that publishes to Amazon S3 you want to replace.

- MaxAggregationInterval

Type: Integer

Valid values: 60 | 600

Description: (Optional) The maximum interval of time, in seconds, during which a flow of packets is captured and aggregated into a flow log record.

- TrafficType

Type: String

Valid values: ACCEPT | REJECT | ALL

Description: (Required) The type of flow log data you want to record and publish.

## Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

- ssm:StartAutomationExecution
- ssm:GetAutomationExecution
- ec2:CreateFlowLogs
- ec2>DeleteFlowLogs

- `ec2:DescribeFlowLogs`

### Document Steps

- `aws:executeAwsApi` - Gathers details about your VPC from the value you specify in the `FlowLogId` parameter.
- `aws:executeAwsApi` - Creates a flow log based on the values you specify for the runbook parameters.
- `aws:assertAwsResourceProperty` - Verifies the newly created flow log publishes to CloudWatch Logs.
- `aws:executeAwsApi` - Deletes the flow log that publishes to Amazon S3.
- `aws:executeScript` - Confirms the flow log that published to Amazon S3 was deleted.

## AWSConfigRemediation- EnableVPCFlowLogsToS3Bucket

### Description

The `AWSConfigRemediation-EnableVPCFlowLogsToS3Bucket` runbook *replaces* an existing Amazon VPC flow log that publishes flow log data to Amazon CloudWatch Logs (CloudWatch Logs) with a flow log that publishes flow log data to the Amazon Simple Storage Service (Amazon S3) bucket you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `DestinationS3BucketArn`

Type: String

Description: (Required) The ARN of the Amazon S3 bucket you want to publish flow log data to.

- `FlowLogId`

Type: String

Description: (Required) The ID of the flow log that publishes to CloudWatch Logs you want to replace.

- MaxAggregationInterval

Type: Integer

Valid values: 60 | 600

Description: (Optional) The maximum interval of time, in seconds, during which a flow of packets is captured and aggregated into a flow log record.

- TrafficType

Type: String

Valid values: ACCEPT | REJECT | ALL

Description: (Required) The type of flow log data you want to record and publish.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:CreateFlowLogs`
- `ec2>DeleteFlowLogs`
- `ec2:DescribeFlowLogs`

### Document Steps

- `aws:executeAwsApi` - Gathers details about your VPC from the value you specify in the `FlowLogId` parameter.
- `aws:executeAwsApi` - Creates a flow log based on the values you specify for the runbook parameters.
- `aws:assertAwsResourceProperty` - Verifies the newly created flow log publishes to Amazon S3.
- `aws:executeAwsApi` - Deletes the flow log that publishes to CloudWatch Logs.
- `aws:executeScript` - Confirms the flow log that published to CloudWatch Logs was deleted.

## AWS-ReleaseElasticIP

### Description

Release the specified Elastic IP address using the allocation ID.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- AllocationId

Type: String

Description: (Required) The Allocation ID of the Elastic IP address.

## AWSConfigRemediation- RemoveUnrestrictedSourceIngressRules

### Description

The AWSConfigRemediation-RemoveUnrestrictedSourceIngressRules runbook removes all ingress rules from the security group you specify that allow traffic from all source addresses.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: ^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[([w+=, @/ -]+)\$

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- SecurityGroupId

Type: String

Description: (Required) The ID of the security group that you want to remove ingress rules that allow traffic from all source addresses from.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeSecurityGroups`
- `ec2:RevokeSecurityGroupIngress`

### Document Steps

- `aws:executeScript` - Removes all ingress rules that allow traffic from all source addresses from the security group you specified in the `SecurityGroupId` parameter.

## AWSConfigRemediation- RemoveVPCDefaultSecurityGroupRules

### Description

The `AWSConfigRemediation-RemoveVPCDefaultSecurityGroupRules` runbook removes all rules from the default security group of the virtual private cloud (VPC) you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-\/\]+$`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `GroupId`

Type: String

Description: (Required) The ID of the security group that you want to remove all rules from.



### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `ec2:DescribeSecurityGroups`
- `ec2:RevokeSecurityGroupEgress`
- `ec2:RevokeSecurityGroupIngress`

### Document Steps

- `aws:assertAwsResourceProperty` - Confirms the security group you specified in the `GroupId` parameter is named `default`.
- `aws:executeScript` - Removes all rules from the security group you specified in the `GroupId` parameter.

## AWSSupport-SetupIPMonitoringFromVPC

### Description

`AWSSupport-SetupIPMonitoringFromVPC` creates an Amazon Elastic Compute Cloud (Amazon EC2) instance in the specified subnet and monitors selected target IPs (IPv4 or IPv6) by continuously running ping, MTR, traceroute and tracetcp tests. The results are stored in Amazon CloudWatch Logs logs, and metric filters are applied to quickly visualize latency and packet loss statistics in a CloudWatch dashboard.

### Additional Information

The CloudWatch Logs data can be used for network troubleshooting and analysis of pattern/trends. Additionally, you can configure CloudWatch alarms with Amazon SNS notifications when packet loss and/or latency reach a threshold. The data can also be used when opening a case with AWS Support, to help isolate an issue quickly and reduce time to resolution when investigating a network issue.

#### Note

To clean up resources created by `AWSSupport-SetupIPMonitoringFromVPC`, you can use the runbook `AWSSupport-TerminateIPMonitoringFromVPC`. For more information, see [AWSSupport-TerminateIPMonitoringFromVPC \(p. 321\)](#).

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- CloudWatchLogGroupNamePrefix

Type: String

Default: / AWSSupport-SetupIPMonitoringFromVPC

Description: (Optional) Prefix used for each CloudWatch log group created for the test results.

- CloudWatchLogGroupRetentionInDays

Type: String

Valid values: 1 | 3 | 5 | 7 | 14 | 30 | 60 | 90 | 120 | 150 | 180 | 365 | 400 | 545 | 731 | 1827 | 3653

Default: 7

Description: (Optional) Number of days you want to keep the network monitoring results for.

- InstanceType

Type: String

Valid values: t2.micro | t2.small | t2.medium | t2.large

Default: t2.micro

Description: (Optional) The EC2 instance type for the EC2Rescue instance. Recommended size: t2.micro.

- SubnetId

Type: String

Description: (Required) The subnet ID for the monitor instance. Be aware that if you specify a private subnet, then you must make sure there is Internet access to allow the monitor instance to setup the test (meaning, install the CloudWatch Logs agent, interact with Systems Manager and CloudWatch).

- TargetIPs

Type: String

Description: (Required) Comma separated list of IPv4s and/or IPv6s to monitor. No spaces allowed. Maximum size is 255 characters. Be aware that if you provide an invalid IP, then the automation will fail and rollback the test setup.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

It is recommended that the user who runs the automation have the **AmazonSSMAutomationRole** IAM managed policy attached. In addition, the user must have the following policy attached to their user account, group, or role:

```
{  
  "Version": "2012-10-17",
```

```
"Statement": [
  {
    "Action": [
      "iam:CreateRole",
      "iam:CreateInstanceProfile",
      "iam:GetRole",
      "iam:GetInstanceProfile",
      "iam:DetachRolePolicy",
      "iam:AttachRolePolicy",
      "iam:PassRole",
      "iam:AddRoleToInstanceProfile",
      "iam:RemoveRoleFromInstanceProfile",
      "iam>DeleteRole",
      "iam>DeleteInstanceProfile",
      "iam:PutRolePolicy",
      "iam>DeleteRolePolicy"
    ],
    "Resource": [
      "arn:aws:iam::
      AWS_account_ID
      :role/AWSSupport/SetupIPMonitoringFromVPC_*",
      "arn:aws:iam::
      AWS_account_ID
      :instance-profile/AWSSupport/SetupIPMonitoringFromVPC_*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "iam:DetachRolePolicy",
      "iam:AttachRolePolicy"
    ],
    "Resource": [
      "arn:aws:iam::aws:policy/service-role/AmazonSSMManagedInstanceCore"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudwatch:DeleteDashboards"
    ],
    "Resource": [
      "*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "ec2:AuthorizeSecurityGroupEgress",
      "ec2:CreateSecurityGroup",
      "ec2>DeleteSecurityGroup",
      "ec2:DescribeSubnets"
    ],
    "Resource": [
      "*"
    ],
    "Effect": "Allow"
  }
]
```

## Document Steps

1. **aws:executeAwsApi** - describe the provided subnet.

2. **aws:branch** - evaluate the TargetIPs input.

(IPv6) If TargetIPs contains an IPv6:

**aws:assertAwsResourceProperty** - check the provided subnet has an IPv6 pool associated

3. **aws:executeAwsApi** - get the latest Amazon Linux 2 AMI from Parameter Store.

4. **aws:executeAwsApi** - create a security group for the test in the subnet's VPC.

(Cleanup) If the security group creation fails:

**aws:executeAwsApi** - delete the security group created by the automation, if it exists.

5. **aws:executeAwsApi** - allow all outbound traffic in the test security group.

(Cleanup) If the security group egress rule creation fails:

**aws:executeAwsApi** - delete the security group created by the automation, if it exists.

6. **aws:executeAwsApi** - create an IAM role for the test EC2 instance

(Cleanup) If the role creation fails:

a. **aws:executeAwsApi** - delete the IAM role created by the automation, if it exists.

b. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

7. **aws:executeAwsApi** - attach the AmazonSSMManagedInstanceCore managed policy

(Cleanup) If the policy attachment fails:

a. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation, if attached.

b. **aws:executeAwsApi** - delete the IAM role created by the automation.

c. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

8. **aws:executeAwsApi** - attach an inline policy to allow setting CloudWatch log group retentions and creating a CloudWatch dashboard

(Cleanup) If the inline policy attachment fails:

a. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation, if created.

b. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.

c. **aws:executeAwsApi** - delete the IAM role created by the automation.

d. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

9. **aws:executeAwsApi** - create an IAM instance profile.

(Cleanup) If the instance profile creation fails:

a. **aws:executeAwsApi** - delete the IAM instance profile created by the automation, if it exists.

b. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.

c. **aws:executeAwsApi** - delete the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.

d. **aws:executeAwsApi** - delete the IAM role created by the automation.

e. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

10. **aws:executeAwsApi** - associate the IAM instance profile to the IAM role.

(Cleanup) If the instance profile and role association fails:

a. **aws:executeAwsApi** - remove the IAM instance profile from the role, if associated.

b. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.

- c. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - d. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - e. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - f. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 11**aws:sleep** - wait for the instance profile to become available.
- 12**aws:runInstances** - create the test instance in the specified subnet, and with the instance profile created earlier attached.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 13**aws:branch** - evaluate the TargetIPs input.

(IPv6) If TargetIPs contains an IPv6:

- aws:executeAwsApi** - assign an IPv6 to the test instance.
- 14**aws:waitForAwsResourceProperty** - wait for the test instance to become a managed instance.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 15**aws:runCommand** - install test pre-requisites:

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 16**aws:runCommand** - validate the provided IPs are syntactically correct IPv4 and/or IPv6 addresses:

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 17**aws:runCommand** - define the MTR test for each of the provided IPs.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 18**aws:runCommand** - define the first ping test for each of the provided IPs.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 19**aws:runCommand** - define the second ping test for each of the provided IPs.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 20**aws:runCommand** - define the tracepath test for each of the provided IPs.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
- b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
- c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
- d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
- e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
- f. **aws:executeAwsApi** - delete the IAM role created by the automation.
- g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

21**aws:runCommand** - define the traceroute test for each of the provided IPs.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
- b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
- c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
- d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
- e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
- f. **aws:executeAwsApi** - delete the IAM role created by the automation.
- g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

22**aws:runCommand** - configure CloudWatch logs.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
- b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
- c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
- d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
- e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
- f. **aws:executeAwsApi** - delete the IAM role created by the automation.
- g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

23**aws:runCommand** - schedule cronjobs to run each test every minute.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
- b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
- c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
- d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
- e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
- f. **aws:executeAwsApi** - delete the IAM role created by the automation.
- g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

24**aws:sleep** - wait for the tests to generate some data.

25**aws:runCommand** - set the desired CloudWatch log group retentions.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 26aws:runCommand** - set the CloudWatch log group metric filters.

(Cleanup) If the step fails:

- a. **aws:changeInstanceState** - terminate the test instance.
  - b. **aws:executeAwsApi** - remove the IAM instance profile from the role.
  - c. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
  - d. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
  - e. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
  - f. **aws:executeAwsApi** - delete the IAM role created by the automation.
  - g. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.
- 27aws:runCommand** - create the CloudWatch dashboard.

(Cleanup) If the step fails:

- a. **aws:executeAwsApi** - delete the CloudWatch dashboard, if it exists.
- b. **aws:changeInstanceState** - terminate the test instance.
- c. **aws:executeAwsApi** - remove the IAM instance profile from the role.
- d. **aws:executeAwsApi** - delete the IAM instance profile created by the automation.
- e. **aws:executeAwsApi** - delete the CloudWatch inline policy from the role created by the automation.
- f. **aws:executeAwsApi** - detach the AmazonSSMManagedInstanceCore managed policy from the role created by the automation.
- g. **aws:executeAwsApi** - delete the IAM role created by the automation.
- h. **aws:executeAwsApi** - delete the security group created by the automation, if it exists.

## Outputs

**createCloudWatchDashboards.Output** - the URL of the CloudWatch dashboard.

**createManagedInstance.InstanceId** - the test instance ID.

# AWSSupport-TerminateIPMonitoringFromVPC

## Description

AWSSupport-TerminateIPMonitoringFromVPC terminates an IP monitoring test previously started by AWSSupport-SetupIPMonitoringFromVPC. Data related to the specified test ID will be deleted.

[Run this Automation \(console\)](#)

## Document type



## Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- InstanceId

Type: String

Description: (Required) The instance ID for the monitor instance.

- SubnetId

Type: String

Description: (Required) The subnet ID for the monitor instance.

### Required IAM permissions

The AutomationAssumeRole parameter requires the following actions to successfully use the runbook.

It is recommended that the user who runs the automation have the **AmazonSSMAutomationRole** IAM managed policy attached. In addition, the user must have the following policy attached to their user account, group, or role:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iam:DetachRolePolicy",
        "iam:RemoveRoleFromInstanceProfile",
        "iam>DeleteRole",
        "iam>DeleteInstanceProfile",
        "iam>DeleteRolePolicy"
      ],
      "Resource": [
        "arn:aws:iam::An-AWS-Account-ID:role/AWSSupport/SetupIPMonitoringFromVPC_*",
        "arn:aws:iam::An-AWS-Account-ID:instance-profile/AWSSupport/SetupIPMonitoringFromVPC_*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "iam:DetachRolePolicy"
      ],
      "Resource": [
```

```
        "arn:aws:iam::aws:policy/service-role/AmazonSSMManagedInstanceCore"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudwatch:DeleteDashboards"
    ],
    "Resource": [
      "*"
    ],
    "Effect": "Allow"
  }
]
```

### Document Steps

1. `aws:assertAwsResourceProperty` - check `AutomationExecutionId` and `InstanceId` are related to the same test.
2. `aws:assertAwsResourceProperty` - check `SubnetId` and `InstanceId` are related to the same test.
3. `aws:executeAwsApi` - retrieve the test security group.
4. `aws:executeAwsApi` - delete the CloudWatch dashboard.
5. `aws:changeInstanceState` - terminate the test instance.
6. `aws:executeAwsApi` - remove the IAM instance profile from the role.
7. `aws:executeAwsApi` - delete the IAM instance profile created by the automation.
8. `aws:executeAwsApi` - delete the CloudWatch inline policy from the role created by the automation.
9. `aws:executeAwsApi` - detach the **AmazonSSMManagedInstanceCore** managed policy from the role created by the automation.
10. `aws:executeAwsApi` - delete the IAM role created by the automation.
11. `aws:executeAwsApi` - delete the security group created by the automation, if it exists.

### Outputs

None

## AWS WAF

AWS Systems Manager Automation provides predefined runbooks for AWS WAF. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

### Topics

- [AWSConfigRemediation-EnableWAFClassicLogging \(p. 323\)](#)
- [AWSConfigRemediation-EnableWAFClassicRegionalLogging \(p. 325\)](#)
- [AWSConfigRemediation-EnableWAFV2Logging \(p. 326\)](#)

## AWSConfigRemediation-EnableWAFClassicLogging

### Description

The `AWSConfigRemediation-EnableWAFClassicLogging` runbook enables logging to Amazon Kinesis Data Firehose (Kinesis Data Firehose) for the AWS WAF web access control list (web ACL) you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- AutomationAssumeRole

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=, @\/-\]+`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- DeliveryStreamName

Type: String

Description: (Required) The name of the Kinesis Data Firehose delivery stream that you want to send logs to.

- WebACLId

Type: String

Description: (Required) The ID of the AWS WAF web ACL that you want to enable logging on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `iam:CreateServiceLinkedRole`
- `waf:GetLoggingConfiguration`
- `waf:GetWebAcl`
- `waf:PutLoggingConfiguration`

### Document Steps

- `aws:executeAwsApi` - Confirms the delivery stream you specify in the `DeliveryStreamName` exists.
- `aws:executeAwsApi` - Gathers the ARN of the AWS WAF web ACL specified in the `WebACLId` parameter.

- `aws:executeAwsApi` - Enables logging for the web ACL.
- `aws:assertAwsResourceProperty` - Verifies logging has been enabled on the AWS WAF web ACL.

## AWSConfigRemediation- EnableWAFClassicRegionalLogging

### Description

The `AWSConfigRemediation-EnableWAFClassicRegionalLogging` runbook enables logging to Amazon Kinesis Data Firehose (Kinesis Data Firehose) for the AWS WAF web access control list (ACL) you specify.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LogDestinationConfigs`

Type: String

Description: (Required) The Amazon Resource Name (ARN) of the Kinesis Data Firehose delivery stream that you want to send logs to.

- `WebACLId`

Type: String

Description: (Required) The ID of the AWS WAF web ACL that you want to enable logging on.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`

- `iam:CreateServiceLinkedRole`
- `waf-regional:GetLoggingConfiguration`
- `waf-regional:GetWebAcl`
- `waf-regional:PutLoggingConfiguration`

### Document Steps

- `aws:executeAwsApi` - Gathers the ARN of the AWS WAF web ACL specified in the `webACLId` parameter.
- `aws:executeAwsApi` - Enables logging for the web ACL.
- `aws:assertAwsResourceProperty` - Verifies logging has been enabled on the AWS WAF web ACL.

## AWSConfigRemediation-EnableWAFV2Logging

### Description

The `AWSConfigRemediation-EnableWAFV2Logging` runbook enables logging for an AWS WAF (AWS WAFV2) web access control list (web ACL) with the specified Amazon Kinesis Data Firehose (Kinesis Data Firehose) delivery stream.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,.\@\/-]+\`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `LogDestinationConfigs`

Type: String

Description: (Required) The Kinesis Data Firehose delivery stream ARN that you want to associate with the web ACL.

### Note

The Kinesis Data Firehose delivery stream ARN must begin with the prefix `aws-waf-logs-`. For example, `aws-waf-logs-us-east-2-analytics`. For more information, see [Amazon Kinesis Data Firehose](#).

- `WebAclArn`

Type: String

Description: (Required) ARN of the web ACL for which logging will be enabled.

#### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `firehose:DescribeDeliveryStream`
- `wafv2:PutLoggingConfiguration`
  
- `wafv2:GetLoggingConfiguration`

#### Document Steps

- `aws:executeScript` - Enables logging for the AWS WAFV2 web ACL and verifies that the logging has the specified configuration.

## Amazon WorkSpaces

AWS Systems Manager Automation provides predefined runbooks for Amazon WorkSpaces. For more information about runbooks, see [Working with runbooks](#) . For information about how to view runbook content, see [View runbook content \(p. 2\)](#) .

#### Topics

- [AWSSupport-RecoverWorkSpace \(p. 327\)](#)

## AWSSupport-RecoverWorkSpace

#### Description

The `AWSSupport-RecoverWorkSpace` runbook performs recovery steps on the Amazon WorkSpaces virtual desktop, known as a `WorkSpace`, you specify. The runbook reboots the `WorkSpace`, and if the state is still `UNHEALTHY` , restores or rebuilds the `WorkSpace` based on the values you specify for the input parameters. Before using this runbook we recommend reviewing [Troubleshooting WorkSpaces Issues](#) in the *Amazon WorkSpaces Administration Guide* .

#### Important

Restoring or rebuilding a `WorkSpace` is a potentially destructive action that can result in the loss of data. This is because the `WorkSpace` is restored from the last available snapshot and data recovered from snapshots can be as old as 12 hours.

The restore option recreates both the root volume and user volume based on the most recent snapshots. The rebuild option recreates the user volume from the most recent snapshot and recreates the `WorkSpace` from the image associated with the bundle the `WorkSpace` was created from. Applications that were installed or system settings that were changed after the `WorkSpace` was created are lost. For more information about restoring and rebuilding `WorkSpaces`, see [Restore a WorkSpace](#) and [Rebuild a WorkSpace](#) in the *Amazon WorkSpaces Administration Guide* .

[Run this Automation \(console\)](#)

**Document type**

Automation

**Owner**

Amazon

**Platforms**

Linux, macOS, Windows

**Parameters**

- AutomationAssumeRole

Type: String

Description: (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf. If no role is specified, Systems Manager Automation uses the permissions of the user that starts this runbook.

- Acknowledge

Type: String

Valid values: Yes

Description: (Required) Entering yes means that you understand the restore and rebuild actions will try to recover the WorkSpace from the most recent snapshot, and that data restored from these snapshots can be as old as 12 hours.

- Reboot

Type: String

Valid values: Yes | No

Default: Yes

Description: (Required) Determines whether the WorkSpace is rebooted.

- Rebuild

Type: String

Valid values: Yes | No

Default: No

Description: (Required) Determines whether the WorkSpace is rebuilt.

- Restore

Type: String

Valid values: Yes | No

Default: No

Description: (Required) Determines whether the WorkSpace is restored.

- `WorkspaceId`

Type: String

Description: (Required) The ID of the Workspace you want to recover.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `workspaces:DescribeWorkspaces`
- `workspaces:DescribeWorkspaceSnapshots`
- `workspaces:RebootWorkspaces`
- `workspaces:RebuildWorkspaces`
- `workspaces:RestoreWorkspace`
- `workspaces:StartWorkspaces`

### Document Steps

- `aws:executeAwsApi` - Gathers the state of the Workspace you specify in the `WorkspaceId` parameter.
- `aws:assertAwsResourceProperty` - Verifies the state of the Workspace is `AVAILABLE`, `ERROR`, `IMPAIRED`, `STOPPED`, or `UNHEALTHY`.
- `aws:branch` - Branches based on the state of the Workspace.
- `aws:executeAwsApi` - Starts the Workspace.
- `aws:branch` - Branches based on the value you specify for the `Action` parameter.
- `aws:waitForAwsResourceProperty` - Waits for the Workspace status after being started.
- `aws:waitForAwsResourceProperty` - Waits for the Workspace state to change to `AVAILABLE`, `ERROR`, `IMPAIRED`, or `UNHEALTHY` after being started.
- `aws:executeAwsApi` - Gathers the state of the Workspace after being started.
- `aws:branch` - Branches based on the state of the Workspace after being started.
- `aws:executeAwsApi` - Gathers the available snapshots for restoring or rebuilding the Workspace.
- `aws:branch` - Branches based on the value you specify for the `Reboot` parameter.
- `aws:executeAwsApi` - Reboots the Workspace.
- `aws:executeAwsApi` - Gathers the state of the Workspace after being started.
- `aws:waitForAwsResourceProperty` - Waits for the state of the Workspace to change to `REBOOTING`.
- `aws:waitForAwsResourceProperty` - Waits for the Workspace state to change to `AVAILABLE`, `ERROR`, or `UNHEALTHY` after being rebooted.
- `aws:executeAwsApi` - Gathers the state of the Workspace after being rebooted.
- `aws:branch` - Branches based on the state of the Workspace after rebooting.
- `aws:branch` - Branches based on the value you specify for the `Restore` parameter.
- `aws:executeAwsApi` - Restores the Workspace. If the restore fails, the runbook tries to rebuild the Workspace.
- `aws:waitForAwsResourceProperty` - Waits for the state of the Workspace to change to `RESTORING`.
- `aws:waitForAwsResourceProperty` - Waits for the Workspace state to change to `AVAILABLE`, `ERROR`, or `UNHEALTHY` after being restored.



- `aws:executeAwsApi` - Gathers the state of the WorkSpace after being restored.
- `aws:branch` - Branches based on the state of the WorkSpace after restoring.
- `aws:branch` - Branches based on the value you specify for the `Rebuild` parameter.
- `aws:executeAwsApi` - Rebuilds the WorkSpace.
- `aws:waitForAwsResourceProperty` - Waits for the state of the WorkSpace to change to `REBUILDING`.
- `aws:waitForAwsResourceProperty` - Waits for the WorkSpace state to change to `AVAILABLE`, `ERROR`, or `UNHEALTHY` after being rebuilt.
- `aws:executeAwsApi` - Gathers the state of the WorkSpace after being rebuilt.
- `aws:assertAwsResourceProperty` - Confirms the state of the WorkSpace is `AVAILABLE`.

## X-Ray

AWS Systems Manager Automation provides predefined runbooks for AWS X-Ray. For more information about runbooks, see [Working with runbooks](#). For information about how to view runbook content, see [View runbook content \(p. 2\)](#).

### Topics

- [AWSConfigRemediation-UpdateXRayKMSKey \(p. 330\)](#)

## AWSConfigRemediation-UpdateXRayKMSKey

### Description

The `AWSConfigRemediation-UpdateXRayKMSKey` runbook enables encryption on your AWS X-Ray data using an AWS Key Management Service (AWS KMS) key. This runbook should only be used as a baseline to ensure that your AWS X-Ray data is encrypted according to minimum recommended security best practices. We recommend encrypting multiple sets of data with different KMS keys.

[Run this Automation \(console\)](#)

### Document type

Automation

### Owner

Amazon

### Platforms

Linux, macOS, Windows

### Parameters

- `AutomationAssumeRole`

Type: String

Allowed values: `^arn:(?:aws|aws-us-gov|aws-cn):iam::\d{12}:role\[\/\w+=,\/-]+`

Description: (Required) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that allows Systems Manager Automation to perform the actions on your behalf.

- `KeyId`

Type: String

Description: (Required) The Amazon Resource Name (ARN), key ID, or the key alias of the KMS key you want AWS X-Ray to use to encrypt data.

### Required IAM permissions

The `AutomationAssumeRole` parameter requires the following actions to successfully use the runbook.

- `ssm:StartAutomationExecution`
- `ssm:GetAutomationExecution`
- `kms:DescribeKey`
- `xray:GetEncryptionConfig`
- `xray:PutEncryptionConfig`

### Document Steps

- `aws:executeAwsApi` - Enables encryption on your X-Ray data using the KMS key you specify in the `KeyId` parameter.
- `aws:waitForAwsResourceProperty` - Waits for the encryption configuration status of your X-Ray to be `ACTIVE`.
- `aws:executeAwsApi` - Gathers the ARN of the key you specify in the `KeyId` parameter.
- `aws:assertAwsResourceProperty` - Verifies encryption has been enabled on your X-Ray.