

Compliance Automation Framework using AWS Config

Objective:

To develop an end-to-end compliance solution that

- Detects non-compliance using AWS Config.
- Automatically remediates using SSM Automation Documents or lambda
- Sends alerts via SNS and stores logs in S3/CloudWatch.

AWS Services Used:

- AWS Config
- SSM Automation Documents (SSM Docs)
- Amazon SNS
- AWS CloudTrail
- IAM
- Amazon S3
- Lambda
- Event bridge

Scenarios to Simulate

Scenario 1: Open Security Group (Port 22 exposed to 0.0.0.0/0)

Step 1: Create the IAM Role for Remediation

Our automation needs permission to perform actions (like changing a security group). We'll create a special role for this.

Create a IAM role for Remediation

Screenshot of the AWS IAM 'Create role' wizard.

Step 1: Name, review, and create

- Name, review, and create**
 - Allow AWS services like EC2, Lambda, or others to perform actions in this account.**
 - Web identity**

Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
 - SAML 2.0 federation**

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
 - Custom trust policy**

Create a custom trust policy to enable others to perform actions in this account.

Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case
Systems Manager

Choose a use case for the specified service.
Use case

- Systems Manager**

Allows SSM to call AWS services on your behalf
- Systems Manager - Inventory and Maintenance Windows**

Allow AWS Systems Manager to call AWS resources on your behalf.

Next Step

Screenshot of the AWS IAM 'SsmSecurityGroupRemediationPolicy' details page.

Identity and Access Management (IAM)

SsmSecurityGroupRemediationPolicy Info

Allows SSM to call AWS services on your behalf

Summary

Creation date: July 24, 2025, 11:34 (UTC-04:00) **ARN**: arn:aws:iam::185187793433:role/SsmSecurityGroupRemediationPolicy

Last activity: - **Maximum session duration**: 1 hour

Permissions **Trust relationships** **Tags** **Last Accessed** **Revoke sessions**

Permissions policies (1) Info

You can attach up to 10 managed policies.

Filter by Type

Policy name	Type	Attached entities
SsmSecurityGroupRe...	Customer inline	0

Permissions boundary (not set)

Create another role assume role

The screenshot shows the AWS IAM Role Details page for 'AutomationServiceRole'. The role was created on July 24, 2025, at 13:23 UTC. It has a maximum session duration of 1 hour. The ARN of the role is copied to the clipboard. The 'Permissions' tab is selected, showing two managed policies attached: 'AmazonEC2FullAccess' and 'AmazonSSMAutomationRole'. There is also a section for generating a policy based on CloudTrail events.

Step 2: Create a SNS notification topic This is how we'll get email alerts.

The screenshot shows the 'Create topic' page for Amazon SNS. The 'Standard' message delivery mode is selected. The topic name is 'SecurityOps-Alerts', and the display name is 'My Topic'. An optional encryption section is shown at the bottom.

The screenshot shows the AWS SNS Topics page. On the left, there's a sidebar with 'Amazon SNS' navigation: Dashboard, Topics (which is selected and highlighted in blue), Subscriptions, and Mobile (Push notifications, Text messaging (SMS)). The main area has a title 'Topics (1)'. A search bar at the top right contains 'Search'. Below it is a table with two columns: 'Name' and 'Type'. There is one entry: 'SecurityOps-Alerts' (Type: Standard). To the right of the table are buttons for 'Edit', 'Delete', 'Publish message', and 'Create topic'. At the bottom right of the table are navigation arrows and a refresh icon.

Create a subscription For Protocol, select Email

The screenshot shows the 'Create subscription' page for the 'SecurityOps-Alerts' topic. The top navigation bar includes 'Amazon SNS > Subscriptions > Create subscription'. The main form is titled 'Create subscription'. It has three sections: 'Details', 'Subscription filter policy - optional', and 'Redrive policy (dead-letter queue) - optional'.
Details: Contains fields for 'Topic ARN' (selected value: 'arn:aws:sns:us-east-1:185187793433:SecurityOps-Alerts'), 'Protocol' (selected value: 'Email'), and 'Endpoint' (selected value: 'sailakshmi0819@gmail.com'). A note below the endpoint says: 'After your subscription is created, you must confirm it.'
Subscription filter policy - optional: A note states: 'This policy filters the messages that a subscriber receives.'
Redrive policy (dead-letter queue) - optional: A note states: 'Send undeliverable messages to a dead-letter queue.'



The screenshot shows the AWS SNS console for the "SecurityOps-Alerts" topic. The topic details are displayed, including Name (SecurityOps-Alerts), ARN (arn:aws:sns:us-east-1:185187793433:SecurityOps-Alerts), Type (Standard), and Topic owner (185187793433). The "Subscriptions" tab is selected, showing one subscription:

ID	Endpoint	Status	Protocol
7cd939d9-e931-4bba-b83c-...	sailakshmi0819@gmail.com	Confirmed	EMAIL

Step 3: Enable AWS Config

This rule will detect when a security group has Port 22 open to the world (0.0.0.0/0).

Screenshot of the AWS Config console showing the "Edit data and delivery channel settings" page. The "Data retention period" section is selected, showing options to retain AWS Config data for 7 years or set a custom retention period. The "Delivery channel" section shows options for an Amazon S3 bucket, including creating a new bucket or choosing an existing one. An S3 Bucket name is specified as "config-bucket-185187793433". The "Amazon SNS topic" section includes a checkbox for streaming configuration changes to an SNS topic. A "Save" button is visible at the bottom right.

Screenshot of the AWS Config console showing the "Edit recorder settings" page. The "Enable recording" checkbox is checked. The "Recording method" section shows two options: "All resource types with customizable overrides" (selected) and "Specific resource types". The "Default settings" section shows recording frequency options: "Continuous recording" (selected) and "Daily recording". The "Override settings" section allows specifying resource types to override recording frequency. A "Save" button is visible at the bottom right.

Step 4: Create the SSM Automation Document

us-east-1.console.aws.amazon.com/systems-manager/documents/create-document?region=us-east-1#documentType=Automation

aws Search [Alt+S] United States (N. Virginia) sailakshmi @ 1851-8779-5433

Remediate-OpenSSH-Port

Design Code

Cancel YAML Actions Create runbook

Undo Redo Format Copy Commands

Zoom in Zoom out Center

```

1 description: 'Remediates a Security Group by revoking ingress for Port 22'
2 schemaVersion: '0.3'
3 assumeRole: '{{ AutomationAssumeRole }}'
4 parameters:
5   AutomationAssumeRole:
6     type: String
7     description: (Required) The ARN of the role that allows Automation to assume it.
8   SecurityGroupId:
9     type: String
10    description: (Required) The ID of the security group to remediate.
11 mainSteps:
12  - name: RevokePublicSshRule
13    action: 'aws:executeAwsApi'
14    inputs:
15      Service: 'ec2'
16      Api: 'RevokeSecurityGroupIngress'
17      GroupId: '{{ SecurityGroupId }}'
18      IpPermissions:
19        - IpProtocol: 'tcp'
20          FromPort: 22
21          ToPort: 22
22          IpRanges:
23            - CidrIp: '0.0.0.0/0'

```

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/systems-manager/documents/?region=us-east-1

aws Search [Alt+S] United States (N. Virginia) sailakshmi @ 1851-8779-5433

Your document was successfully created

AWS Systems Manager > Documents

Owned by Amazon Owned by me Shared with me Favorites - new All documents

No categories supported for this owner type.

Documents Preferences Actions Create document

Search by keyword or filter by tag or attributes

Remediate-OpenSSH-Port

Document type Owner
Automation 185187793433

Platform types Windows, Linux, MacOS

Default version 1

CloudShell Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 5: Create the AWS Config Rule (The "Detection Rule")

- Go to the AWS Config service.
- Ensure the recorder is ON.
- On the left menu, click Rules.
- Click Add rule.

- Select Add AWS managed rule.
- In the search box, type restricted-ssh and select the rule named restricted-ssh.
- Click Next, then Add rule.

Specify rule type

Add rules to help you manage the ideal configuration settings of your AWS resources. You can add any of the following predefined, customizable AWS Config Managed rules, or you can create your own AWS Config Custom rule using AWS Lambda functions or Guard Custom policy.

Select rule type				
<input checked="" type="radio"/> Add AWS managed rule Deploy the following managed rules in their default state or customize to suit your needs.	<input type="radio"/> Create custom Lambda rule Use a Lambda function with your custom code to evaluate whether your AWS resources comply with the rule.	<input type="radio"/> Create custom rule using Guard Use Guard Custom policy that you write to evaluate whether your AWS resources comply with the rule.		

AWS Managed Rules (668)

Name	Resource types	Trigger type	Description	Supports evaluation mode
restricted-ssh	AWS::EC2::SecurityGroup	HYBRID	Checks whether security groups that are in use disallow unrestricted incoming SSH traffic.	DELETE

Configure rule

Customize any of the following fields

Details

Name
A unique name for the rule. 128 characters max. No special characters or spaces.

Description - optional
Describe what the rule evaluates and how to fix resources that don't comply.

Managed rule name

Evaluation mode

Turn on proactive evaluation

Remediation rule

The screenshot shows the AWS Config Remediation rule configuration interface. The top navigation bar includes links for AWS Config, Rules, restricted-ssh, and Manage remediation. The main title is "Edit: Remediation action".

Select remediation method:

- Automatic remediation: The remediation action gets triggered automatically when the resources in scope become noncompliant.
- Manual remediation: The selected remediation action must be triggered manually by you in order to remediate the noncompliant resources in scope.

If a resource is still noncompliant after auto-remediation, you can specify the maximum number of retry attempts and the maximum amount of time in seconds before auto-remediation stops and AWS Config places a remediation exception.

Note: there are costs associated with running a remediation action.

Maximum retry attempts: 5
Retry time window in seconds: 60
Min: 1, Max: 25
Min: 1, Max: 2678000

Remediation action details:

Remediation actions are run using AWS Systems Manager Automation.

Choose remediation action: Remediate-OpenSSH-Port

Resource ID parameter:

Using the dropdown list, you can pass the resource ID of noncompliant resources to a parameter of the remediation action. The parameters available in the dropdown list depend on the selected remediation action.

SecurityGroupId

Parameters:

Parameters allow you to pass specific information to your remediation action, such as resource IDs or configuration settings. Each remediation action has its own set of parameters. Valid values include StringList and String. Custom SSM documents for remediation with other data types are not supported.

For StringLists, enter values as an array of strings (value 1, value 2, value 3). For Strings, enter the value as a single string (value).

AutomationAssumeRole: arn:aws:iam::185187793433:role/SsmSecurityGrn
SecurityGroupId: (required)

Buttons: Cancel, Save changes

Step 6: Connect Everything with EventBridge

This rule listens for the "Non-compliant" alert from Config and triggers our SSM document.

Create an event bridge

- Define rule detail: rule type give rule with an event pattern click on next
- Build event pattern:
- We want to detect whenever someone adds a security group inbound rule, specifically port 22.
- Select Event source: AWS events or AWS services
- AWS service: EC2
- Event type: AWS API Call via CloudTrail
- Any specific operation Yes
 - Type: AuthorizeSecurityGroupIngress
- This will catch events like adding an inbound SSH rule.
- Select target :
- target type –ssm automation
- Document –select your ssm document
- Configure Input Transformer Click “Create a new input transformer”
- Give input paths

```
{
  "groupId": "$.detail.requestParameters.groupId"
}
```

- Input template as

```
{
  "AutomationAssumeRole": "arn:aws:iam::123456789012:role/SSMAutomationRemediationRole",
  "SecurityGroupId": "<groupId>"
}
```

- Review and Create

us-east-1.console.aws.amazon.com/events/home?region=us-east-1#/rules/create

Amazon EventBridge > Rules > Create rule

Step 1 Define rule detail

Rule detail

Name: Trigger-SSM-Remediation-for-OpenSSH

Description - optional: Enter description

Event bus: Info: default

Rule type: Rule with an event pattern

Schedule: A rule that runs on a schedule

Cancel **Next**

Amazon EventBridge sidebar:

- Developer resources: Learn, Sandbox, Quick starts
- Buses: Event buses Updated, Rules, Global endpoints, Archives, Replays
- Pipes: Pipes
- Scheduler: Schedules, Schedule groups
- Integration: Partner event sources, API destinations, Connections
- Schema registry: Schemas

Documentation

us-east-1.console.aws.amazon.com/events/home?region=us-east-1#/rules/create

Amazon EventBridge > Rules > Create rule

Step 2 Build event pattern

Events: You don't have to select or enter a sample event, but it's recommended so you can reference it when writing and testing the event pattern, or filter criteria.

Event source: Select the event source from which events are sent.

- AWS events or EventBridge partner events**: Events sent from AWS services or EventBridge partners.
- Other**: Custom events or events sent from more than one source, e.g., events from AWS services and partners.
- All events**: All events sent to your account.

Sample event - optional: You don't have to select or enter a sample event, but it's recommended so you can reference it when writing and testing the event pattern, or filter criteria.

Event pattern

Creation method

- Use schema**: Use an Amazon EventBridge schema to generate the event pattern.
- Use pattern form**: Use a template provided by EventBridge to create an event pattern.
- Custom pattern (JSON editor)**: Write an event pattern in JSON.

Event source: AWS service or EventBridge partner as source

AWS service: AWS services

AWS service: The name of the AWS service as the event source

EC2

Event type: The type of events as the source of the matching pattern

AWS API Call via CloudTrail

Event pattern: Event pattern, or filter to match the events

```

1 {
2   "source": ["aws.ec2"],
3   "detail-type": ["AWS API Call via CloudTrail"],
4   "detail": {
5     "eventSource": ["ec2.amazonaws.com"],
6     "eventName": ["AuthorizeSecurityGroupIngress"]
7   }
8 }
  
```

Copy **Test pattern** **Edit pattern**

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Amazon EventBridge sidebar:

- Developer resources: Learn, Sandbox, Quick starts
- Buses: Event buses Updated, Rules, Global endpoints, Archives, Replays
- Pipes: Pipes
- Scheduler: Schedules, Schedule groups
- Integration: Partner event sources, API destinations, Connections
- Schema registry: Schemas

Documentation

us-east-1.console.aws.amazon.com/events/home?region=us-east-1#/rules/create

Amazon EventBridge > Rules > Create rule

Event pattern

Creation method

- Use schema: Use an Amazon EventBridge schema to generate the event pattern.
- Use pattern form: Use a template provided by EventBridge to create an event pattern.
- Custom pattern (JSON editor): Write an event pattern in JSON.

Event source: AWS service or EventBridge partner as source

AWS services: EC2

Event type: AWS API Call via CloudTrail

Event Type Specification 1

- Any operation
- Specific operation(s)
- Specific operation(s): AuthorizeSecurityGroupIngress

Event pattern

```

1 {
  "source": ["aws.ec2"],
  "detail-type": ["AWS API Call via CloudTrail"],
  "detail": {
    "eventSource": ["ec2.amazonaws.com"],
    "eventName": ["AuthorizeSecurityGroupIngress"]
  }
}

```

Buttons: Copy, Test pattern, Edit pattern

Buttons at bottom: Cancel, Previous, Next

us-east-1.console.aws.amazon.com/events/home?region=us-east-1#/rules/create

Amazon EventBridge > Rules > Create rule

Step 5: Review and create

Target types

- EventBridge event bus
- EventBridge API destination (SaaS partner)
- AWS service

Select a target: Systems Manager Automation

Document: Remediate-OpenSSH-Port

Configure automation parameter(s)

Input path

```

1 {
  "groupId": "$.detail.requestParameters.groupId"
}

```

JSON is valid

Buttons: Copy, Prettify

Template

```

1 {
  "AutomationServiceRoleArn": "arn:aws:iam:185187793433:role/AutomationServiceRole",
  "SecurityGroupId": "group1"
}

```

Buttons at bottom: CloudShell, Feedback, © 2025, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, Cookie preferences

Step 7: Testing the Automation setup

- Create a new security group Add an Inbound rule:

- Type: SSH
- Source: Anywhere-IPv4 (0.0.0.0/0)
- Save the rule.
- Now, wait. AWS Config can take a few minutes to evaluate the change.
- Check the result:
 - You should receive an email notification from SNS.
 - Go back to the Security Group in the EC2 console. Refresh the page. The rule you just added should be gone!
 - Go to AWS Config > Rules. The restricted-ssh rule might briefly show "Non-compliant" before changing back to "Compliant".
 - Go to Systems Manager > Automations. You will see an execution of your Remediate document with a "Success" status.

The screenshot shows the 'Create security group' wizard in the AWS EC2 console. The 'Inbound rules' section contains one rule: 'Allows SSH access to developers'. The 'Source' is set to 'Anywhere...' and the 'Destination' is '0.0.0.0/0'. A warning message below states: 'Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' The 'Outbound rules' section shows a single rule for 'All traffic' to 'Custom' destination, also with '0.0.0.0/0'. A similar warning message is present here: 'Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.' The 'Tags - optional' section is empty. At the bottom right, there are 'Cancel' and 'Create security group' buttons.

Got email from SNS

2 of 601 < > ⌂ ⌂

[AWS Config:us-east-1] AWS::EC2::SecurityGroup sg-0b62619d87fedf997 is NON_COMPLIANT with restric... Compose Print

Inbox ×

AWS Notifications <no-reply@sns.amazonaws.com> to me ▾

View the Timeline for this Resource in AWS Config Management Console:
<https://console.aws.amazon.com/config/home?region=us-east-1#/timeline/AWS::EC2::SecurityGroup/sg-0b62619d87fedf997?time=2025-07-24T18:12:37.084Z>

New Compliance Change Record:

```
-----  
{  
    "awsAccountId": "185187793433",  
    "configRuleName": "restricted-ssh",  
    "configRuleARN": "arn:aws:config:us-east-1:185187793433:config-rule/config-rule/bycacb",  
    "resourceType": "AWS::EC2::SecurityGroup",  
    "resourceId": "sg-0b62619d87fedf997",  
    "awsRegion": "us-east-1",  
    "newEvaluationResult": {  
        "evaluationResultIdentifier": {  
            "evaluationResultQualifier": {  
                "configRuleName": "restricted-ssh",  
                "resourceType": "AWS::EC2::SecurityGroup",  
                "resourceId": "sg-0b62619d87fedf997",  
                "evaluationMode": "DETECTIVE"  
            },  
            "resourceEvaluationId": null,  
            "orderingTimestamp": "2025-07-24T18:12:37.084Z"  
        },  
        "complianceType": "NON_COMPLIANT"  
    }  
}
```

System manager with a "Success" status

← → ⓘ us-east-1.console.aws.amazon.com/systems-manager/automation/executions?region=us-east-1

aws Search [Alt+S] New Chrome available

AWS Systems Manager > Automation

Executions Integrations Preferences

Automation executions

View all automation executions that you have permission to view, including executions completed in the past 30 days.

Filter executions Show child automations

Execution ID Runbook name Status Start time End time Executed by

0dc40180-d51c-4cbe-a39a-52c5afdf93a50	Remediate-OpenSSH-Port	Success	Thu, 24 Jul 2025 18:08:56 GMT	Thu, 24 Jul 2025 18:08:57 GMT	arn:awssts:185187793433:assumed-role/AWSServiceRoleForConfigRemediation/aws
---------------------------------------	------------------------	---------	-------------------------------	-------------------------------	---

AWS config show "Non-compliant"

The screenshot shows the AWS Config Rules page. On the left, there's a navigation sidebar with options like Dashboard, Conformance packs, Rules, Resources, Aggregators, and more. The main content area is titled 'Rules' and contains a table with one row. The table columns are Name, Remediation action, Type, Enabled evaluation mode, and Detective compliance. The single row shows 'restricted-ssh' as the Name, 'Remediate-OpenSSH-Port' as the Remediation action, 'AWS managed' as the Type, 'DETECTIVE' as the Enabled evaluation mode, and '1 Noncompliant resource(s)' as the Detective compliance status.

Security group rule deleted

The screenshot shows the AWS EC2 Security Groups page. The left sidebar includes sections for Instances, Images, and Network & Security. The main content shows a security group named 'sg-0b62619d87fedf997 - ssh'. Under the 'Inbound rules' tab, there is a table with one entry: 'restrict ssh' with port range '22'. The table has columns for Name, Security group rule ID, IP version, Type, Protocol, Port range, Source, and Description. A note at the bottom says 'No security group rules found'.

Scenario 2: Missing Required Tags on EC2

Step 1: Create AWS Config Rule to Detect Missing Tags

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules/add

Specify rule type

Add rules to help you manage the ideal configuration settings of your AWS resources. You can add any of the following predefined, customizable AWS Config Managed rules, or you can create your own AWS Config Custom rule using AWS Lambda functions or Guard Custom policy.

Select rule type

Add AWS managed rule
Deploy the following managed rules in their default state or customize to suit your needs.

Create custom Lambda rule
Use a Lambda function with your custom code to evaluate whether your AWS resources comply with the rule.

Create custom rule using Guard
Use Guard Custom policy that you write to evaluate whether your AWS resources comply with the rule.

AWS Managed Rules (668)

Find Rules 1 match

Name = required-tags Clear filters

Name	Resource types	Trigger type	Description	Supported evaluation mode
required-tags	AWS:ACM::Certificate, AWS:AutoScaling::AutoScalingGroup, AWS:CloudFormation::Stack, AWS:CodeBuild::Project, AWS:DynamoDB::Table, AWS:EC2::CustomerGateway, AWS:EC2::Instance, AWS:EC2::InternetGateway, AWS:EC2::NetworkAcl, AWS:EC2::NetworkInterface, AWS:EC2::RouteTable, AWS:EC2::SecurityGroup, AWS:EC2::Subnet, AWS:EC2::Volume, AWS:EC2::VPC, AWS:EC2::VPNConnection, AWS:EC2::VPNGateway, AWS:ElasticLoadBalancing::LoadBalancer, AWS:ElasticLoadBalancingV2::LoadBalancer, AWS:RDS::DBInstance, AWS:RDS::DBSecurityGroup, AWS:RDS::EventSubscription, AWS:RDS::DBSubnetGroup, AWS:RDS::EventSubscription, AWS:Redshift::Cluster, AWS:Redshift::ClusterParameterGroup, AWS:Redshift::ClusterSecurityGroup, AWS:Redshift::ClusterSnapshot, AWS:Redshift::ClusterSubnetGroup, AWS:S3::Bucket	CHANGE-TRIGGERED	Checks whether your resources have the tags that you specify.	DETECTIVE AI

CloudShell Feedback Outlook (new) Cancel Next © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Trigger type

When configuration changes Runs when there are changes to your specified AWS resources Periodic Runs on the frequency that you choose

Scope of changes

Choose when evaluations will occur.

All changes When any resource recorded by AWS Config is created, changed, or deleted Resources When any resource that matches the specified type, or the type plus identifier, is created, changed, or deleted Tags When any resource with the specified tag is created, changed, or deleted

Resources

This rule can be triggered only when the recorded resources are created, edited, or deleted. Specify the resources to record by editing the Settings page.

Resource category

All resource categories Multiple selected

AWS EC2 Instance

Resource identifier - optional

Enter resource identifier

Parameters

Rule parameters define attributes that your resources must adhere to for compliance with the rule. Example attributes include a required tag or a specified S3 bucket. Optional parameters that are not valid, such as missing a key or a value, will not be saved.

Key	Value
tag1Key	CostCenter <input type="button"/> Remove
tag1Value	(optional) <input type="button"/> Remove
Environment	(optional) <input type="button"/> Remove
Owner	(optional) <input type="button"/> Remove
tag3Key	(optional) <input type="button"/> Remove
tag3Value	(optional) <input type="button"/> Remove

The screenshot shows the AWS Config Rules page. A green banner at the top states: "The rule: required-tags has been added to your account." Below this, the "Rules" section is displayed with the following details:

Name	Remediation action	Type	Enabled evaluation mode
required-tags	Not set	AWS managed	DETECTIVE

The left sidebar includes links for Dashboard, Conformance packs, Rules, Resources, Aggregators, Compliance Dashboard, Conformance packs, Rules, Inventory Dashboard, Resources, Authorizations, Advanced queries (Preview), Settings, and What's new. Documentation, Partners, FAQs, and Pricing links are also present.

Step 2: SSM Automation Document for EC2 Tag Remediation

Create SSM Document: Remediate-Missing-EC2-Tags

The screenshot shows the AWS Systems Manager Documents page. A green banner at the top states: "Your document was successfully created." Below this, the "Documents" section is displayed with the following details:

Document type	Owner
Automation	185187793433

The left sidebar includes links for Owned by Amazon, Owned by me, Shared with me, Favorites - new, and All documents. A message indicates "No categories supported for this owner type." The right sidebar includes Preferences, Actions, and Create document buttons.

Step 3: IAM Role for EC2 Tag Remediation

Policy: AmazonEC2FullAccess

Attach to role: SSM-EC2-TagRemediation-Role

The screenshot shows the AWS IAM Roles details page for the role 'SSM-EC2-TagRemediation-Role'. The left sidebar shows navigation options like Identity and Access Management (IAM), Access management, and Access reports. The main content area displays the role's summary, including its ARN (arn:aws:iam::185187793433:role/SSM-EC2-TagRemediation-Role) which is copied to the clipboard. It also shows the maximum session duration as 1 hour. Below this, the 'Permissions' tab is selected, showing two managed policies attached: 'AmazonEC2FullAccess' and 'AWSConfigUserAccess'. There are tabs for Trust relationships, Tags, Last Accessed, and Revoke sessions. A 'Permissions boundary' section indicates it is not set. A 'Generate policy based on CloudTrail events' section shows no requests in the past 7 days.

Step 4: AWS Config Rule Remediation

- Rule: required-tags
- Remediation action: Remediate-Missing-EC2-Tags
- Parameter: InstanceId = resourceId
- IAM Role: SSM-EC2-TagRemediation-Role

The screenshot shows the AWS Config Remediation rule configuration page. It includes sections for Select remediation method (Automatic remediation selected), Remediation action details (Remediate-Missing-EC2-Tags selected), Rate Limits (Concurrent Execution Rate: 2, Error Rate: 5), and Resource ID parameter (Instanceld selected). The parameters section shows RESOURCE_ID and AutomationAssumeRole defined.

Step 5: Testing the Automation setup

- launch a new EC2 instance with **no tags**.
- AWS Config detects it's **NON_COMPLIANT** with the required-tags rule.
- SSM automatically **adds the missing Environment and Owner tags** with default values.
- You should receive an email notification from SNS.

Ec2 instance with no tags

The screenshot shows the AWS EC2 Instances details page for an instance with ID i-062af2ec306f85dd7. The instance has the following details:

- Hostname type:** IP name: ip-172-31-21-229.ec2.internal
- Answer private resource DNS name (IPv4 (A)):** Private IP DNS name (IPv4 only) ip-172-31-21-229.ec2.internal
- Auto-assigned IP address:** 18.209.61.77 [Public IP]
- VPC ID:** vpc-05f0e32fcf899db09
- IAM Role:** -
- Subnet ID:** subnet-01d02e41622b80257
- Instance ARN:** arnaws:ec2:us-east-1:185187793433:instance/i-062af2ec306f85dd7
- Elastic IP addresses:** -
- AWS Compute Optimizer finding:** Opt-in to AWS Compute Optimizer for recommendations.
- Auto Scaling Group name:** -
- Managed:** false

The "Tags" tab is selected in the navigation bar. There are no tags listed under "Tags".

AWS Config detects it's NON_COMPLIANT

The screenshot shows the AWS Config Rules page for the us-east-1 region. A single rule is listed:

Name	Remediation action	Type	Enabled evaluation mode	Detective compliance
required-tags	Remediate-Missing-EC2-Tags	AWS managed	DETECTIVE	1 Noncompliant resource(s)

The "Detective compliance" column indicates 1 Noncompliant resource(s).

SSM automatically added the missing Tags

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like Dashboard, EC2 Global View, Events, Instances (selected), Instances Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, and Load Balancing. The main area displays a table for 'Instances (1/1) Info'. The table has columns for Name (testtags), Instance ID (i-0e037177cd99d04ee), Instance state (Running), Instance type (t2.micro), Status check (2/2 checks passed), Alarm status (green), Availability Zone (us-east-1e), Public IPv4 DNS (ec2-54-236-103-46.co...), and Public IPv4 IP (54.236.103.46). Below the table, a detailed view for instance i-0e037177cd99d04ee (testtags) is shown, specifically the 'Tags' tab. It lists three tags: Owner (DefaultOwner), Environment (DefaultEnv), and Name (testtags). There's also a 'Manage tags' button.

Remediate document with a "Success" status

The screenshot shows the AWS Systems Manager Automation page. The top navigation bar includes links for us-east-1.console.aws.amazon.com, Automation, Executions (selected), Integrations, and Preferences. The main content area is titled 'Automation executions' and shows a table of completed executions. The table has columns for Execution ID (0c3f7e9e-cd6d-4c0a-b1d3-fbd17f28ed52), Runbook name (Remediate-Missing-EC2-Tags), Status (Success), Start time (Thu, 24 Jul 2025 23:23:09 GMT), End time (Thu, 24 Jul 2025 23:23:10 GMT), and Executed by (arn:aws:iam::18518777). There are buttons for 'Create runbook' and 'Execute runbook'.

Received an email notification from SNS

Search mail

[AWS Config:us-east-1] AWS::EC2::Instance i-04ab3b9f12814ebd0 is NON_COMPLIANT with required-tags...

AWS Notifications <no-reply@sns.amazonaws.com> to me 3:24 PM (0 minutes ago)

View the Timeline for this Resource in AWS Config Management Console: <https://console.aws.amazon.com/config/home?region=us-east-1#/timeline/AWS::EC2::Instance/i-04ab3b9f12814ebd0?time=2025-07-24T20:05:43.764Z>

New Compliance Change Record:

```
{
  "awsAccountId": "185187793433",
  "configRuleName": "required-tags",
  "configRuleARN": "arn:aws:config:us-east-1:185187793433:config-rule/config-rule-ptuwzy",
  "resourceType": "AWS::EC2::Instance",
  "resourceId": "i-04ab3b9f12814ebd0",
  "awsRegion": "us-east-1",
  "newEvaluationResult": {
    "evaluationResultIdentifier": {
      "evaluationResultQualifier": {
        "configRuleName": "required-tags",
        "resourceType": "AWS::EC2::Instance",
        "resourceId": "i-04ab3b9f12814ebd0",
        "evaluationMode": "DETECTIVE"
      }
    }
  }
}
```

Scenario3: Disabled cloud trail

Step1: Create a lambda function

us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/check_cLOUDTRAIL_ENABLED?newFunction=true&tab=code

Lambda > Functions > check_cLOUDTRAIL_ENABLED

Successfully created the function check_cLOUDTRAIL_ENABLED. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Code source [Info](#)

check_cLOUDTRAIL_ENABLED

```
lambda_function.py
1 import boto3
2
3 def lambda_handler(event, context):
4     client = boto3.client('cloudtrail')
5     trails = client.describe_trails(includeShadowTrails=False)['trailList']
6
7     for trail in trails:
8         name = trail['Name']
9         status = client.get_trail_status(Name=name)
10        if status['IsLogging']:
11            return {
12                'compliance_type': 'COMPLIANT',
13                'annotation': f'CloudTrail "{name}" is enabled and logging.'
14            }
15
16        return {
17            'compliance_type': 'NON_COMPLIANT',
18            'annotation': 'No CloudTrail logging is enabled in the account.'
19        }
```

[Open in Visual Studio Code](#) [Upload from](#)

DEPLOY [UNDEPLOYED CHANGES]

- You have undeployed changes.

[Deploy \(Ctrl+Shift+U\)](#) [Test \(Ctrl+Shift+I\)](#)

TEST EVENTS [NONE SELECTED]

- + Create new test event

[CloudShell](#) [Feedback](#)

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

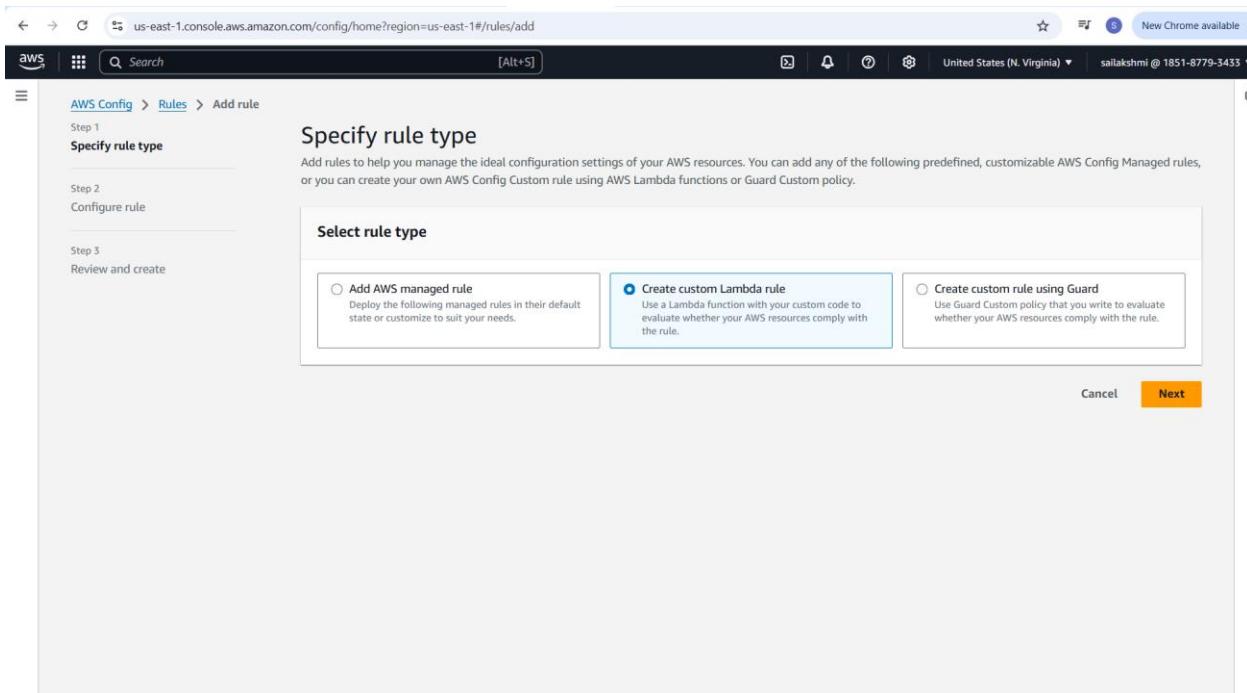
- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

[Learn more](#) [Start tutorial](#)

Step2: Create AWS Config Custom Rule with lambda

- Go to AWS Config → Rules → Add Rule
- Choose "Add custom rule"

- Enter:
 - Name: CustomCloudTrailEnabledRule
 - Resource type: Choose AWS::CloudTrail::Trail or leave blank for all resources
 - Lambda Function: Choose the Lambda you created (check_cloudtrail_enabled)
 - Execution Role: Ensure Lambda has permissions to call cloudtrail:DescribeTrails, cloudtrail:GetTrailStatus



Screenshot of the AWS Config 'Add rule' configuration page:

Configure rule

Details

- Name:** CustomCloudTrailEnabledRule
- Description - optional:** Your description can be anything you like.
- AWS Lambda function ARN:** arn:aws:lambda:us-east-1:185187793433:function:check_cldtrail_enabled

Evaluation mode

- Turn on proactive evaluation
- Turn on detective evaluation

Trigger type

Screenshot of the AWS Config 'Rules' page:

Rules

Name	Remediation action	Type	Enabled evaluation mode
CustomCloudTrailEnabledRule	Not set	Custom Lambda	DETECTIVE

Step 3: Check IAM role for lambda has permissions to call cloudtrail:DescribeTrails, cloudtrail:GetTrailStatus

Step 1
Specify permissions

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Action": [
6         "cloudtrail:DescribeTrails",
7         "cloudtrail:GetTrailStatus"
8       ],
9       "Effect": "Allow",
10      "Resource": "*"
11    }
12  ]
13 }
14

```

Edit statement

Select a statement

Select an existing statement in the policy or add a new statement.

+ Add new statement

Identity and Access Management (IAM)

Policy cloudtrail created.

Last activity - Maximum session duration 1 hour

Permissions | Trust relationships | Tags | Last Accessed | Revoke sessions

Permissions policies (2) Info

You can attach up to 10 managed policies.

Policy name	Type	Attached entities
AWSLambdaBasicExecutionRole-9b058292...	Customer managed	1
cloudtrail	Customer inline	0

Filter by Type

Permissions boundary (not set)

Generate policy based on CloudTrail events

You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

Generate policy

Step 4: Create EventBridge Rule:

- Go to **EventBridge** → **Rules** → **Create rule**
- Event Source: AWS API Call via CloudTrail
- Event Pattern:
{
 "source": ["aws.config"],

```

"detail-type": ["Config Rules Compliance Change"],
"detail": {
    "messageType": ["ComplianceChangeNotification"],
    "configRuleName": ["CustomCloudTrailEnabledRule"],
    "newEvaluationResult": {
        "complianceType": ["NON_COMPLIANT"]
    }
}
}

```

The screenshot shows the 'Create rule' wizard in the Amazon EventBridge console. The left sidebar navigation includes 'Dashboard', 'Developer resources' (with 'Learn', 'Sandbox', 'Quick starts'), 'Buses' (with 'Event buses' updated), 'Rules' (selected), 'Pipes', 'Scheduler', 'Integration', and 'Schema registry'. The main content area is titled 'Build event pattern' and shows the following steps:

- Step 1: Define rule detail**
- Step 2: Build event pattern** (selected)
- Step 3: Select target(s)**
- Step 4 - optional**
- Step 5: Configure tags**
- Step 6: Review and create**

Events: You don't have to select or enter a sample event, but it's recommended so you can reference it when writing and testing the event pattern, or filter criteria.

Event source: Select the event source from which events are sent.

- AWS events or EventBridge partner events
- Other
- All events

Sample event - optional: You don't have to select or enter a sample event, but it's recommended so you can reference it when writing and testing the event pattern, or filter criteria.

Event pattern: Write an event pattern in JSON. You can test the event pattern against the sample event. You can also go to pre-defined pattern.

- Use schema
- Use pattern form
- Custom pattern (JSON editor)

Content-based filter syntax: A dropdown menu with options like 'Prefix matching' and 'Insert'.

us-east-1.console.aws.amazon.com/events/home?region=us-east-1#/eventbus/default/rules/CloudTrail/edit

Amazon EventBridge <

Developer resources

- Learn
- Sandbox
- Quick starts
- Buses**
 - Event buses Updated
 - Rules**
 - Global endpoints
 - Archives
 - Replays
- Pipes**
 - Pipes
- Scheduler**
 - Schedules
 - Schedule groups
- Integration**
 - Partner event sources
 - API destinations
 - Connections
- Schema registry**

Create rule

Creation method

- Use schema
Use an Amazon EventBridge schema to generate the event pattern.
- Use pattern form
Use a template provided by EventBridge to create an event pattern.
- Custom pattern (JSON editor)
Write an event pattern in JSON.

Event pattern
Write an event pattern in JSON. You can test the event pattern against the sample event. You can also go to pre-defined pattern.

Prefix matching Content-based filter syntax

```

1 {
2   "source": ["aws.config"],
3   "detail-type": ["Config Rules Compliance Change"],
4   "detail": {
5     "messageType": ["ComplianceChangeNotification"],
6     "configRuleName": ["CLOUD_TRAIL_ENABLED"],
7     "newEvaluationResults": [
8       {
9         "complianceType": ["NON_COMPLIANT"]
10      }
11    ]
12  }

```

JSON is valid

us-east-1.console.aws.amazon.com/events/home?region=us-east-1#/rules/create

Amazon EventBridge <

Developer resources

- Learn
- Sandbox
- Quick starts
- Buses**
 - Event buses Updated
 - Rules**
 - Global endpoints
 - Archives
 - Replays
- Pipes**
 - Pipes
- Scheduler**
 - Schedules
 - Schedule groups
- Integration**
 - Partner event sources
 - API destinations
 - Connections
- Schema registry**

Create rule

Step 5 Review and create

Target types
Select an EventBridge event bus, EventBridge API destination (SaaS partner), or another AWS service as a target.

- EventBridge event bus
- EventBridge API destination
- AWS service

Select a target | info
Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule)

Target location

Target in this account Target in another AWS account

Function

Permissions

Use execution role (recommended)

Execution role
EventBridge needs permission to send events to the target specified above. By continuing, you are allowing us to do so. [EventBridge and AWS Identity and Access Management](#)

Create a new role for this specific resource Use existing role

Role name

Amazon_EventBridge_Invoke_Lambda_9134626

Additional settings

The screenshot shows the AWS EventBridge Rules page. On the left, there's a sidebar with sections like Developer resources, Buses, Pipes, Scheduler, Integration, and Schema registry. The main area has a green success message: "Rule CloudTrail was created successfully". Below it, a "Rules" section explains what a rule does. A "Select event bus" step is shown with a dropdown set to "default". The "Rules (1)" table lists one rule:

Name	Status	Type	ARN	Description
CloudTrail	Enabled	Standard	arn:aws:events:us-east-1:85187793433:rule/CloudTrail	-

Step 5: Create CloudWatch Log Group

The screenshot shows the AWS CloudWatch Log Groups page. The sidebar includes sections like AI Operations, Alarms, Logs, Metrics, and Application Signals. The main area shows a green success message: "Log group '/aws/lambda/EnableCloudTrailLog' has been created.". The "Log groups (2)" table lists two log groups:

Log group	Log class	Anomaly d...	Data pr...	Sensitiv...	Retention	Metric f...
/cloudtrail/logs	Standard	Configure	-	-	Never expire	-
/aws/lambda/EnableCloudTrailLog	Standard	Configure	-	-	Never expire	-

Step 6: Create a cloud trail

The screenshot shows the AWS CloudTrail Trails page. A single trail, 'Trailtest', is listed in the table. The table columns include Name, Home region, Multi-region trail, ARN, Insights, Organization trail, S3 bucket, Log file prefix, CloudWatch Logs log group, and Status. The 'Status' column shows 'Logging' with a green checkmark. The 'Log file prefix' column contains the value 'aws-cloudtrail-logs-185187793433-72727464'. The ARN column shows the full ARN: arn:aws:cloudtrail:us-east-1:185187793433:trail/Trailtest.

Test the setup

- Stop Logging: Select your trail → click Actions → Stop logging
- AWS Config detects it's **NON_COMPLIANT**
- EventBridge detects **NON_COMPLIANT** and triggers Lambda
- Lambda automatically calls StartLogging
- Go back to CloudTrail → Check Logging = ON
- You should receive an email notification from SNS.

The screenshot shows the AWS CloudTrail Trail details page for 'Trailtest'. In the 'General details' section, under 'Trail logging', there is a radio button labeled 'Logging' with a green checkmark. A 'Stop logging?' dialog box is displayed in the center. The dialog asks 'Stop logging?' and provides information: 'You will no longer log events to your S3 bucket or CloudWatch Logs log group, but your S3 bucket still stores existing log files.' and 'You can still access existing log files in your S3 bucket.' It has 'Cancel' and 'Stop logging' buttons. The 'Edit' button is located at the top right of the main page area.

AWS Config detects it's **NON_COMPLIANT**

The screenshot shows the AWS Config Rules details page for a rule named "CustomCloudTrailEnabledRule". The left sidebar includes links for Dashboard, Conformance packs, Rules (selected), Resources, Aggregators, Advanced queries, Settings, and What's new. The main content area displays the rule details, showing it is in DETECTIVE mode, last evaluated on July 25, 2025 at 12:25 PM, and triggered by oversized configuration changes. It also lists resources in scope, specifically a CloudTrail Trail named "Trailtest" which is marked as Noncompliant.

Received an email notification from SNS.

The screenshot shows an AWS SNS email notification. The subject line is "[AWS Config:us-east-1] AWS::CloudTrail::Trail Trailtest is NON_COMPLIANT with CustomCloudTrailEna...". The message body starts with "AWS Notifications <no-reply@sns.amazonaws.com> to me" and includes a link to view the timeline in the AWS Config Management Console. It then provides a "New Compliance Change Record" in JSON format:

```
{
  "awsAccountId": "185187793433",
  "configRuleName": "CustomCloudTrailEnabledRule",
  "configRuleARN": "arn:aws:config:us-east-1:185187793433:config-rule/config-rule/tl8izj",
  "resourceType": "AWS::CloudTrail::Trail",
  "resourceId": "Trailtest",
  "awsRegion": "us-east-1",
  "newEvaluationResult": {
    "evaluationResultIdentifier": {
      "evaluationResultQualifier": {
        "configRuleName": "CustomCloudTrailEnabledRule",
        "resourceType": "AWS::CloudTrail::Trail",
        "resourceId": "Trailtest",
        "evaluationMode": "DETECTIVE"
      }
    },
    "resourceEvaluationId": null,
    "orderingTimestamp": "2025-07-25T16:25:51.000Z"
  },
  "complianceType": "NON_COMPLIANT",
}
```

Check Logging = ON

The screenshot shows the AWS CloudTrail console with the URL <https://us-east-1.console.aws.amazon.com/cloudtrailv2/home?region=us-east-1#/trails/arm:aws:cloudtrail:us-east-1:185187793433:trail/Trailtest>. The page displays the 'General details' for the 'Trailtest' trail. Key information includes:

- Trail logging:** Enabled (Logging)
- Trail name:** Trailtest
- Multi-region trail:** Yes
- Log file validation:** Disabled
- SNS notification delivery:** Disabled
- Last log file delivered:** July 25, 2025, 12:26:51 (UTC-04:00)
- Last file validation delivered:** -
- Last SNS notification:** -
- Log file SSE-KMS encryption:** Not enabled
- Apply trail to my organization:** Not enabled

The left sidebar shows the navigation menu for CloudTrail, including Dashboard, Event history, Insights, Lake (Dashboards, Query, Event data stores, Integrations), Trails (selected), and Settings.

Cloud watch logs

The screenshot shows the AWS CloudWatch Logs console with the URL [https://us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#logsV2:log-groups/log-group/\\$25Faws\\$25Flambda\\$25Fcheck_cloudtrail_enabled](https://us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#logsV2:log-groups/log-group/$25Faws$25Flambda$25Fcheck_cloudtrail_enabled). The page displays the 'Log streams (46)' section. The table lists the following log streams:

Log stream	Last event time
2025/07/25/[\$LATEST]3b3c5fe4ccfd44c18b64c7a90d560fa2	2025-07-25 16:40:30 (UTC)
2025/07/25/[\$LATEST]82a3886a87d648a58d7a3e770dfbd893	2025-07-25 16:40:30 (UTC)
2025/07/25/[\$LATEST]107190b0dd1b4ec69a354a0d8f53a5d4	2025-07-25 16:40:28 (UTC)
2025/07/25/[\$LATEST]126e34cf32ed4fe2bcd3fbfaaddc5b17	2025-07-25 16:40:28 (UTC)
2025/07/25/[\$LATEST]8c503a72700d45c389d99a4e50e492ca	2025-07-25 16:40:28 (UTC)
2025/07/25/[\$LATEST]af954d4ee9bd4263b20fe4117e9fbe1d	2025-07-25 16:40:25 (UTC)
2025/07/25/[\$LATEST]90992c51ba05476380389e22aaed8a21	2025-07-25 16:40:24 (UTC)
2025/07/25/[\$LATEST]e19aa0e2d36941a68b4f2c46f1e0c168	2025-07-25 16:40:22 (UTC)
2025/07/25/[\$LATEST]b3ddb8035b14da98e4df27df9bd28	2025-07-25 16:40:22 (UTC)
2025/07/25/[\$LATEST]d737a74100e54403bdaf01a966829089	2025-07-25 16:40:22 (UTC)
2025/07/25/[\$LATEST]082821b4b6a24cd68ef07d604f93620a	2025-07-25 16:25:55 (UTC)

The left sidebar shows the navigation menu for CloudWatch, including Favorites and recents, AI Operations (New), Alarms, Logs (Log groups, Log Anomalies, Live Tail, Logs Insights, Contributor Insights), Metrics, Application Signals (New (APM)), Network Monitoring, and Insights.

The screenshot shows the AWS CloudWatch Log Events interface. The left sidebar navigation includes CloudWatch, AI Operations, Alarms, Logs (Log groups, Log Anomalies, Live Tail, Logs Insights, Contributor Insights), Metrics, Application Signals (APM), Network Monitoring, and Insights. The main content area is titled "Log events" and displays a table of log entries. The columns are "Timestamp" and "Message". The messages show CloudTrail API requests for creating a trail named "testtrail".

Timestamp	Message
2025-07-25T22:23:46.431Z	START RequestId: 1cda982d-159e-42b6-82f3-0e5116109554 Version: \$LATEST
2025-07-25T22:23:46.672Z	Detected trails: [{"Name": "testtrail", "S3BucketName": "aws-cloudtrail-logs-185187793433-ced2d6be", "IncludeGlobalServiceEvents": ...}]
2025-07-25T22:23:46.672Z	Checking trail: testtrail
2025-07-25T22:23:46.709Z	Trail status: {"IsLogging": true, "LatestDeliveryTime": datetime.datetime(2025, 7, 25, 22, 22, 30, 189000, tzinfo=tzlocal()), "Star...}
2025-07-25T22:23:46.709Z	Trail 'testtrail' is already logging.
2025-07-25T22:23:46.750Z	END RequestId: 1cda982d-159e-42b6-82f3-0e5116109554
2025-07-25T22:23:46.750Z	REPORT RequestId: 1cda982d-159e-42b6-82f3-0e5116109554 Duration: 318.05 ms Billed Duration: 319 ms Memory Size: 128 MB Max Memory U...
2025-07-25T22:23:47.969Z	START RequestId: b11b5fef-d8c1-4262-9c42-44effe94d3c0 Version: \$LATEST
2025-07-25T22:23:48.210Z	Detected trails: [{"Name": "testtrail", "S3BucketName": "aws-cloudtrail-logs-185187793433-ced2d6be", "IncludeGlobalServiceEvents": ...}]
2025-07-25T22:23:48.210Z	Checking trail: testtrail
2025-07-25T22:23:48.249Z	Trail status: {"IsLogging": true, "LatestDeliveryTime": datetime.datetime(2025, 7, 25, 22, 22, 30, 189000, tzinfo=tzlocal()), "Star...}
2025-07-25T22:23:48.249Z	Trail 'testtrail' is already logging.
2025-07-25T22:23:48.290Z	END RequestId: b11b5fef-d8c1-4262-9c42-44effe94d3c0
2025-07-25T22:23:48.290Z	REPORT RequestId: b11b5fef-d8c1-4262-9c42-44effe94d3c0 Duration: 320.46 ms Billed Duration: 321 ms Memory Size: 128 MB Max Memory U...

Scenario 4 : S3 Bucket Public Access Block Enforcement

Step 1: Enable AWS Config and Create Compliance Rule

- Go to **AWS Config** Console.
- Click **Rules > Add rule**.
- Search for and select these **managed rules**:
 - s3-bucket-public-read-prohibited
 - s3-bucket-public-write-prohibited
- Configure the rules to evaluate all S3 buckets.
- Save and enable the rules.

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules/add

Step 2 [Configure rule](#)

Step 3 [Review and create](#)

Select rule type

Add AWS managed rule
Deploy the following managed rules in their default state or customize to suit your needs.

Create custom Lambda rule
Use a Lambda function with your custom code to evaluate whether your AWS resources comply with the rule.

Create custom rule using Guard
Use Guard Custom policy that you write to evaluate whether your AWS resources comply with the rule.

AWS Managed Rules (668)

Name	Resource types	Trigger type	Description	Supported evaluation mode	Labels
s3-bucket-public-read-prohibited	AWS::S3::Bucket	HYBRID	Checks that your Amazon S3 buckets do not allow public read access. The rule checks the Block Public Access settings, the bucket policy, and the bucket access control list (ACL).	DETECTIVE	S3, Zelk
s3-bucket-public-write-prohibited	AWS::S3::Bucket	HYBRID	Checks that your Amazon S3 buckets do not allow public write access. The rule checks the Block Public Access settings, the bucket policy, and the bucket access control list (ACL).	DETECTIVE	S3, Zelk

Find Rules 2 matches

[Clear filters](#)

[Cancel](#) [Next](#)

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules

The rule: s3-bucket-public-read-prohibited has been added to your account.

AWS Config > Rules

Rules

A rule is a compliance check that helps you manage your ideal configuration settings. AWS Config evaluates whether your resource configurations comply with relevant rules and displays the compliance results.

Name	Remediation action	Type	Enabled evaluation mode
s3-bucket-public-read-prohibited	Not set	AWS managed	DETECTIVE

View details Edit rule Actions Add rule

Filter by compliance status All

Documentation

The screenshot shows the AWS Config Rules page. A green banner at the top indicates that the rule 's3-bucket-public-write-prohibited' has been added. The main content area displays the 'Rules' section, explaining what a rule is and how it helps manage configuration settings. Below this is a table titled 'Rules' showing two entries:

Name	Remediation action	Type	Enabled evaluation mode
s3-bucket-public-read-prohibited	Not set	AWS managed	DETECTIVE
s3-bucket-public-write-prohibited	Not set	AWS managed	DETECTIVE

Step 2: Create an IAM Role for Lambda

Create a role named S3PublicAccessBlockLambdaRole

The screenshot shows the 'Specify permissions' step of creating an IAM role. The left sidebar shows 'Step 1: Specify permissions' is selected. The main area contains a 'Policy editor' with a JSON view of the policy document:

```
1 ▼ {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "s3:GetBucketPublicAccessBlock",
8         "s3:PutBucketPublicAccessBlock",
9         "s3:GetBucketPolicy",
10        "s3:DeleteBucketPolicy",
11        "s3:PutBucketPolicy",
12        "logs:CreateLogGroup",
13        "logs:CreateLogStream",
14        "logs:PutLogEvents"
15      ],
16      "Resource": "*"
17    }
18  ]
19 }
20 |
```

To the right of the JSON editor is a sidebar with options to 'Edit statement', 'Select a statement', and a button to '+ Add new statement'.

PERMISSIONS | **TRUST RELATIONSHIPS** | **TAGS** | **LAST ACCESSED** | **REVOKE SESSIONS**

Permissions policies (1) Info

You can attach up to 10 managed policies.

Filter by Type

Policy name	Type	Attached entities
S3PublicAccessBlockLambdaRole	Customer inline	0

▶ Permissions boundary (not set)

Step 3: Create the Lambda Function for Remediation

Attach a role to lambda which we created before

Code source Info

EXPLORER

- S3_BUCKET_PUBLICACCESS_BLOCK
 - lambda_function.py

DEPLOY

- Deploy (Ctrl+Shift+U)
- Test (Ctrl+Shift+I)

TEST EVENTS (NONE SELECTED)

- Create new test event

Open in Visual Studio Code Upload from

Info Tutorials

Learn how to implement common use cases in AWS Lambda.

Create a simple web app ^

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more ?

Start tutorial

Step 4: Create an EventBridge Rule

- Go to **Amazon EventBridge Console**.
- Click **Rules > Create rule**.
- Name: **S3PublicAccessBlockScheduledRemediation**
- Select **Target**: Lambda function
- Define event pattern
- Create rule.

{

"source": ["aws.config"],

```

"detail-type": ["Config Rules Compliance Change"],

"detail": {

  "configRuleName": [
    "s3-bucket-public-read-prohibited",
    "s3-bucket-public-write-prohibited"
  ],
  "newEvaluationResult": {
    "complianceType": ["NON_COMPLIANT"]
  }
}
}
}

```

The screenshot shows the 'Define rule detail' step of the Amazon EventBridge rule creation wizard. The left sidebar lists various developer resources like Learn, Sandbox, and Quick starts, along with Buses, Pipes, Scheduler, Integration, and Schema registry. The main panel shows the following details:

- Step 1: Define rule detail**
- Name:** S3PublicAccessBlockScheduledRemediation
- Description - optional:** Enter description
- Event bus:** default
- Rule type:**
 - Rule with an event pattern:** Selected. Description: A rule that runs when an event matches the defined event pattern. EventBridge sends the event to the specified target.
 - Schedule:** A rule that runs on a schedule.

aws | Search [Alt+S] | United States (N. Virginia) | sailakshmi @ 1851-8779-3433 | Create rule

Amazon EventBridge < Rules > Create rule

Event pattern Info

Creation method

- Use schema Use an Amazon EventBridge schema to generate the event pattern.
- Use pattern form Use a template provided by EventBridge to create an event pattern.
- Custom pattern (JSON editor) Write an event pattern in JSON.

Event pattern Write an event pattern in JSON. You can test the event pattern against the sample event. You can also go to pre-defined pattern.

Prefix matching Insert Content-based filter syntax

```

1 {
2   "source": ["aws.config"],
3   "detail-type": ["Config Rules Compliance Change"],
4   "detail": {
5     "configRuleName": [
6       "s3-bucket-public-read-prohibited",
7       "s3-bucket-public-write-prohibited"
8     ],
9     "newEvaluationResult": {
10       "complianceType": ["NON_COMPLIANT"]
11     }
12   }
13 }
14

```

JSON is valid

aws | Search [Alt+S] | United States (N. Virginia) | sailakshmi @ 1851-8779-3433 | Create rule

Amazon EventBridge < Rules > Create rule

Step 4 - optional Configure tags
Step 5 Review and create

Target 1

Target types Select an EventBridge event bus, EventBridge API destination (SaaS partner), or another AWS service as a target.

- EventBridge event bus
- EventBridge API destination
- AWS service

Select a target Info Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule)

Lambda function

Target location Target in this account Target in another AWS account

Function S3_bucket_publicaccess_block

Configure version/alias

Permissions Use execution role (recommended)

Execution role EventBridge needs permission to send events to the target specified above. By continuing, you are allowing us to do so. [EventBridge and AWS Identity](#) and [Access Management](#)

Create a new role for this specific resource Use existing role

Role name Amazon_EventBridge_Invoke_Lambda_1710009630

Additional settings

The screenshot shows the AWS EventBridge Rules page. On the left, there's a navigation sidebar with sections like Dashboard, Developer resources, Buses, Pipes, Scheduler, Integration, and Schema registry. The main area has a green header bar indicating "Rule S3PublicAccessBlockScheduledRemediation was created successfully". Below this, there's a "Rules" section with a sub-section "Select event bus" where "Event bus" is set to "default". The "Rules (1)" table lists one rule:

Name	Status	Type	ARN
S3PublicAccessBlockScheduledRemediation	Enabled	Standard	arn:aws:events:us-east-1:85187793433:rule/S3PublicAccessBlockScheduledRemediation

Step 5 : Test the set up

- Create a Test S3 Bucket
- Make the Bucket Public
- Wait for AWS Config to Evaluate and Check the Rule It should show Noncompliant
- You should receive an email notification from SNS

Created a bucket with public access

The screenshot shows the AWS S3 console with the bucket 'oneill123'. The 'Permissions' tab is active. Under 'Access finding', it says 'Access findings are provided by IAM external access analyzers. Learn more about How IAM analyzer findings work.' A link to 'View analyzer for us-east-1' is present. In the 'Block public access (bucket settings)' section, 'Block all public access' is set to 'Off'. The 'Bucket policy' section indicates 'No policy to display.'

oneill123 Info

Objects | Metadata | Properties | **Permissions** | Metrics | Management | Access Points

Permissions overview

Access finding

Access findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer findings work](#).

[View analyzer for us-east-1](#)

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#).

Block all public access

⚠ Off

► Individual Block Public Access settings for this bucket

[Edit](#)

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#).

[Edit](#)

[Delete](#)

No policy to display.

[Copy](#)

The screenshot shows the AWS S3 console with the bucket 'oneill123'. The 'Permissions' tab is active. Under 'Access finding', it says 'Access findings are provided by IAM external access analyzers. Learn more about How IAM analyzer findings work.' A link to 'View analyzer for us-east-1' is present. In the 'Block public access (bucket settings)' section, 'Block all public access' is set to 'Off'. The 'Bucket policy' section indicates 'No policy to display.'

Permissions overview

Access finding

Access findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer findings work](#).

[View analyzer for us-east-1](#)

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#).

Block all public access

⚠ Off

► Individual Block Public Access settings for this bucket

[Edit](#)

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#).

[Edit](#)

[Delete](#)

No policy to display.

[Copy](#)

Received an email notification from SNS

Search mail

[AWS Config:us-east-1] AWS::S3::Bucket oneill123 is NON_COMPLIANT with s3-bucket-public-read-proh...

AWS Notifications <no-reply@sns.amazonaws.com> to me 2:50 PM (0 minutes ago)

View the Timeline for this Resource in AWS Config Management Console:
<https://console.aws.amazon.com/config/home?region=us-east-1#/timeline/AWS::S3::Bucket/oneill123?time=2025-07-25T18:50:09.284Z>

New Compliance Change Record:

```
{ "awsAccountId": "185187793433", "configRuleName": "s3-bucket-public-read-prohibited", "configRuleARN": "arn:aws:config:us-east-1:185187793433:config-rule/config-rule-x2gzqs", "resourceType": "AWS::S3::Bucket", "resourceId": "oneill123", "awsRegion": "us-east-1", "newEvaluationResult": { "evaluationResultIdentifier": { "evaluationResultQualifier": { "configRuleName": "s3-bucket-public-read-prohibited", "resourceType": "AWS::S3::Bucket", "resourceId": "oneill123", "evaluationMode": "DETECTIVE" }, "resourceEvaluationId": null, "orderingTimestamp": "2025-07-25T18:50:09.284Z" }, "complianceType": "NON_COMPLIANT", "status": "PENDING_EVALUATION" } }
```

AWS Config detects it's **NON_COMPLIANT**

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules/details?configRuleName=s3-bucket-public-read-prohibited

AWS Config

Dashboard Conformance packs Rules Resources Aggregators Compliance Dashboard Conformance packs Rules Inventory Dashboard Resources Authorizations Advanced queries [Preview](#) Settings What's new Documentation Partners FAQs Pricing

Data classification and handling Network architecture and secure configuration Common controls Control access to data Network access control design and configuration Frameworks CIS-AWS-Benchmark-v1.4 ISO-IEC-27001:2013-Annex-A NIST-SP-800-53-r5 PCI-DSS-v3.2.1 PCI-DSS-v4.0 Scope of changes Resources Remediation action Not set

Amazon Simple Storage Service (Amazon S3) Governed resources API identifier Last successful evaluation July 25, 2025 2:50 PM Detective compliance Noncompliant resource(s)

ED Deployable Regions 33 of 33 Regions

Trigger type • Periodic: 24 hours • Configuration changes

AWS Config resource types S3 Bucket

Resources in scope

ID	Type	Status	Annotation	Compliance
oneill123	S3 Bucket	-	The S3 bucket policy allows public read access.	Noncompliant

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules

AWS Config > Rules

Rules

A rule is a compliance check that helps you manage your ideal configuration settings. AWS Config evaluates whether your resource configurations comply with relevant rules and displays the compliance results.

Name	Remediation action	Type	Enabled evaluation mode	Detective compliance
s3-bucket-public-read-pr...	Not set	AWS managed	DETECTIVE	⚠ 1 Noncompliant resou...

us-east-1.console.aws.amazon.com/s3/buckets/oneill123?region=us-east-1&bucketType=general&tab=permissions

Amazon S3 > Buckets > oneill123

Permissions overview

Access finding
Access findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer findings work](#).
[View analyzer for us-east-1](#)

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
On
► Individual Block Public Access settings for this bucket

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

Cloud watch logs

CloudWatch Log groups /aws/lambda/S3_bucket_publicaccess_block

CloudWatch

- Favorites and recents
- Dashboards
- AI Operations
- Alarms △ 0 ○ 0 ○ 0
 - In alarm
 - All alarms
 - Billing
- Logs
 - Log groups
 - Log Anomalies
 - Live Tail
 - Logs Insights
 - Contributor Insights
- Metrics
- Application Signals (APM)
- GenAI Observability Preview
- Network Monitoring
- Insights

Standard

ARN arn:aws:logs:us-east-1:185187793453:log-group:/aws/lambda/S3_bucket_publicaccess_block:*

Creation time 20 minutes ago

Retention Never expire

Stored bytes -

Subscription filters 0

Contributor Insights rules -

KMS key ID -

Anomaly detection Configure

Sensitive data count -

Field indexes Configure

Transformer Configure

Log streams

Log streams (4)

Log stream	Last event time
2025/07/25/[LATEST]024d3b3a1b2d4208a6f7453f5ee44ce9	2025-07-25 18:50:48 (UTC)
2025/07/25/[LATEST]d1c713253fc14e2a83bd9bc21901134b	2025-07-25 18:41:54 (UTC)
2025/07/25/[LATEST]622975a2bf8644018fd5c989e4fc28d	2025-07-25 18:40:32 (UTC)
2025/07/25/[LATEST]c2fb466a55354d7cbccb2c68818880f2	2025-07-25 18:33:59 (UTC)

Filter log streams or try prefix search

Exact match Show expired Info

Create log stream Search all log streams

CloudWatch Log groups /aws/lambda/S3_bucket_publicaccess_block > 2025/07/25/[LATEST]024d3b3a1b2d4208a6f7453f5ee44ce9

CloudWatch

- Favorites and recents
- Dashboards
- AI Operations
- Alarms △ 0 ○ 0 ○ 0
 - In alarm
 - All alarms
 - Billing
- Logs
 - Log groups
 - Log Anomalies
 - Live Tail
 - Logs Insights
 - Contributor Insights
- Metrics
- Application Signals (APM)
- GenAI Observability Preview
- Network Monitoring
- Insights

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Filter events - press enter to search

Display ▾

Actions Start tailing Create metric filter

Timestamp	Message
No older events at this moment. Retry	
2025-07-25T18:50:46.584Z	INIT_START Runtime Version: python:3.13.v50 Runtime Version ARN: arn:aws:lambda:us-east-1:runtime:83a0b29e480e14176225231a6e561282aa...
2025-07-25T18:50:47.081Z	START RequestId: 556b2754-3ad6-4207-8ff3-48343d95ab13 Version: \$LATEST
2025-07-25T18:50:47.886Z	Applying full public access block to bucket: oneill1123
2025-07-25T18:50:48.027Z	Removing public statements from bucket policy for: oneill1123
2025-07-25T18:50:48.137Z	END RequestId: 556b2754-3ad6-4207-8ff3-48343d95ab13
2025-07-25T18:50:48.137Z	REPORT RequestId: 556b2754-3ad6-4207-8ff3-48343d95ab13 Duration: 1055.77 ms Billed Duration: 1056 ms Memory Size: 128 MB Max Memory U...
No newer events at this moment. Auto retry paused. Resume	

Actions Start tailing Create metric filter

AWS Search [Alt+S] United States (N. Virginia) sailakshmi @ 1851-8779-3433

Amazon EventBridge > Rules > S3PublicAccessBlockScheduledRemediation

S3PublicAccessBlockScheduledRemediation

[Edit](#) [Disable](#) [Delete](#) [CloudFormation Template](#)

Rule details [Info](#)

Rule name S3PublicAccessBlockScheduledRemediation	Status Enabled	Event bus name default	Type Standard
Description This rule triggers a scheduled remediation task to block public access for S3 buckets.	Rule ARN arn:aws:events:us-east-1:185187793433:rule/S3PublicAccessBlockScheduledRemediation	Event bus ARN arn:aws:events:us-east-1:185187793433:event-bus/default	

[Event pattern](#) [Targets](#) [Monitoring](#) [Tags](#)

Targets

Details	Target Name	Type	ARN	Input	Role
▼	S3_bucket_publicaccess_block	Lambda function	arn:aws:lambda:us-east-1:185187793433:function:S3_bucket_publicaccess_block	Matched event	Amazon_EventBridge_Invoke_Lambda_1710009630
Input to target: Matched event Additional parameters: -- Dead-letter queue (DLQ): -					