

Module 2: Security Management in AWS

Demo Document 8

edureka!

edureka!

© Brain4ce Education Solutions Pvt. Ltd.

Title: Setup AWS Config to Evaluate Compliance For the Resources in Your Environment

Use Case:

You are the Cloud Engineer in one of the Pharma company who has multiple workloads running in their AWS account. Your IT manager wants you to ensure a proper security checks are put in place for compliance and take appropriate remediate action in case of non-compliance. You are instructed to deploy AWS Config solution that should meet following requirements.

1. Requirement 1

- a. Check whether your running EC2 instances are using approved AMIs.
- b. If non-compliance found, configure Remediation Action to stop the EC2 instance automatically.
- c. Identify who created the non-compliant resource using AWS CloudTrail

2. Requirement 2

- a. Identify if public read access enabled for any S3 bucket.
- b. If non-compliance, configure Remediation Action to send notification to Security Team
- c. Identify who created the non-compliant resource using AWS CloudTrail

edureka!

Objective:

You will learn how to setup AWS Config to evaluate compliance for the resources in your environment.

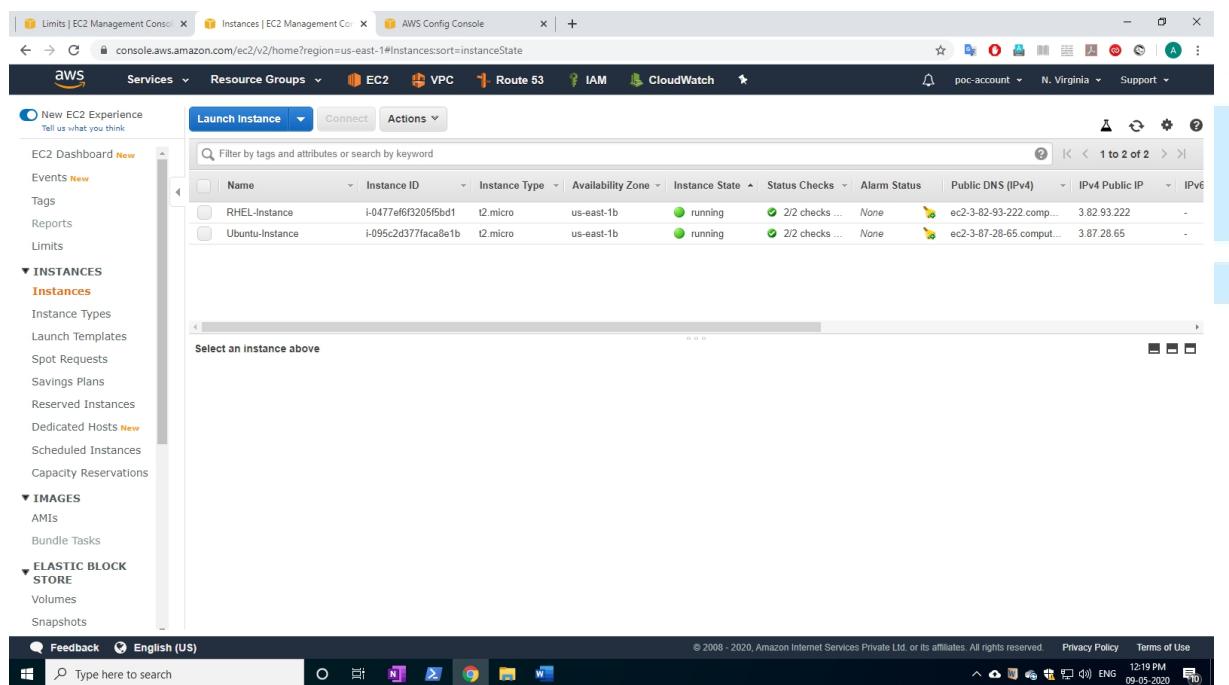
Topics Covered:

- Provision **AWS EC2** servers
- Create **IAM** role for AWS Config Remediation Automation
- Setup initial **AWS Config**
- Create **AWS Config rule** to identify non-compliance as mentioned in the requirements
- Create **Remediation Action** in AWS Config as mentioned in the requirements

Solution Steps:

[Note: us-east-1 region is used to provision all resources in this lab]

1. Create 2 EC2 instances to evaluate compliance as follows.
 - A RHEL instance, (Considering it will be *is compliant* to your upcoming rule)
 - An Ubuntu instance, (Considering it will be *non-compliant* to your upcoming rule)



The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for Events, Tags, Reports, Limits, Instances (with sub-links for Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Scheduled Instances, Capacity Reservations), Images (AMIs, Bundle Tasks), and Elastic Block Store (Volumes, Snapshots). The main content area displays a table of instances. The table has columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), IPv4 Public IP, and IPv6. Two instances are listed: 'RHEL-Instance' and 'Ubuntu-Instance'. Both instances are in the 'running' state in 'us-east-1b'. The 'Status Checks' column shows '2/2 checks ...' for both. The 'Alarm Status' column shows 'None' for both. The 'Public DNS (IPv4)' and 'IPv4 Public IP' columns show the respective hostnames and IP addresses for each instance. The 'Actions' dropdown menu for the first instance is open, showing options like 'Stop', 'Terminate', 'Reboot', 'Launch', and 'Clone'.

2. Create an IAM role for Automation Role

- Navigate *IAM -> Roles -> Create role*

Security Management in AWS

The screenshot shows the AWS IAM Management Console. On the left, there's a sidebar with navigation links like Dashboard, Access management, Roles, Access reports, and Credential report. The main area displays a table of roles, with columns for Role name, Trusted entities, and Last activity. A search bar at the top allows filtering. The table lists various roles such as AmazonSSMRoleForAutomationAssumeQuickSetup, AWSControlTowerAdmin, and AWSReservedSSO_AWSAdministratorAccess_1ac5a02161... The status for most roles is 'Today' or 'None'.

- Navigate *Create role -> AWS service -> EC2*
- Click *Next Permissions*

This screenshot shows the 'Create role' wizard, step 1: Select type of trusted entity. It features a tab navigation with step 1 highlighted. Below it, there are four options: 'AWS service' (selected), 'Another AWS account', 'Web identity', and 'SAML 2.0 federation'. A note below says 'Allows AWS services to perform actions on your behalf.' At the bottom, there are sections for 'Choose a use case' (Common use cases: EC2, Lambda) and 'Or select a service to view its use cases' (listing services like API Gateway, AWS Backup, AWS Chatbot, AWS Support, Amplify, AppStream 2.0, CodeDeploy, CodeGuru, CodeStar Notifications, Comprehend, Connect, ElastiCache, Elastic Beanstalk, Elastic Container Service, IoT Things Graph, EMR, Kinesis, Lake Formation, Lambda, Lex, Rekognition, RoboMaker, S3, SMS, SNS, SWF). Buttons for 'Required' (marked with an asterisk), 'Cancel', and 'Next: Permissions' are at the bottom.

- Select *Attach permissions policies -> AmazonSSMAutomationRole*

Security Management in AWS

The screenshot shows the 'Create role' wizard in the IAM Management Console. Step 2, 'Attach permissions policies', is active. A search bar filters results for 'ssm'. The 'AmazonSSMAutomationRole' policy is selected and highlighted with a blue border. Other policies listed include 'AmazonEC2RoleforSSM', 'AmazonSSMAutomationApproverAccess', 'AmazonSSMDirectoryServiceAccess', 'AmazonSSMFullAccess', 'AmazonSSMMaintenanceWindowRole', 'AmazonSSMManagedinstanceCore', and 'AmazonSSMReadOnlyAccess'. The 'Used as' column indicates which policies are currently used.

- Give a suitable name for *Role name* and click *Create*

The screenshot shows the 'Create role' wizard in the IAM Management Console. Step 4, 'Review', is active. The 'Role name' field contains 'AWSConfig-SSMAutomationRole'. The 'Role description' field contains 'Allows EC2 instances to call AWS services on your behalf.' Under 'Trusted entities', it lists 'AWS service: ec2.amazonaws.com'. Under 'Policies', it lists 'AmazonSSMAutomationRole'. The 'Permissions boundary' section notes 'Permissions boundary is not set'. The status bar at the bottom right shows the date and time as 09-05-2020 03:15 PM.

- After the IAM role is created, go to the summary section of the role
- Navigate *Trust relationship* -> *Edit Trust relationship*
- Replace "ec2.amazonaws.com" with "ssm.amazonaws.com"
- Click *Update Trust Policy*

Security Management in AWS

[The reason why you are doing this is, AWS Config internally calls SSM Automation Document to perform Remediation Action. The SSM Automation requires IAM roles to perform Automation Actions such as EC2-Stop, EC2-Start etc.]

The screenshot shows the AWS IAM Management Console with the URL console.aws.amazon.com/iam/home?region=us-east-1#/roles/AWSConfig-SSMAutomationRole?section=trust. The page title is "Edit Trust Relationship". It displays a JSON policy document:

```
1. { "Version": "2012-10-17",  
2.   "Statement": [  
3.     {  
4.       "Effect": "Allow",  
5.       "Principal": {  
6.         "Service": "ssm.amazonaws.com"  
7.       },  
8.       "Action": "sts:AssumeRole"  
9.     }  
10.   ]  
11. }
```

At the bottom right, there are "Cancel" and "Update Trust Policy" buttons.

- You should see the *Trusted entities* showing `ssm.amazonaws.com` under *Trust relationship* tab of the role summary

The screenshot shows the AWS IAM Management Console with the URL console.aws.amazon.com/iam/home?region=us-east-1#/roles/AWSConfig-SSMAutomationRole. The left sidebar shows the navigation menu for IAM. The main page title is "Summary" for the role "AWSConfig-SSMAutomationRole".

Role Details:

- Role ARN: arn:aws:iam::938577227724:role/AWSConfig-SSMAutomationRole
- Role description: Allows EC2 instances to call AWS services on your behalf. | Edit
- Instance Profile ARNs: arn:aws:iam::938577227724:instance-profile/AWSConfig-SSMAutomationRole
- Path: /
- Creation time: 2020-05-09 15:22 UTC+0530
- Last activity: Not accessed in the tracking period
- Maximum CLI/API session duration: 1 hour | Edit

Permissions Tab:

- Permissions: You can view the trusted entities that can assume the role and the access conditions for the role. Show policy document
- Edit trust relationship

Trusted entities:

- The following trusted entities can assume this role:
 - Trusted entities: The identity provider(s) ssm.amazonaws.com

Conditions:

- The following conditions define how and when trusted entities can assume the role.
 - There are no conditions associated with this role.

- Now IAM role is created successfully. You can move on to next section to create AWS Config service and configure the service using the IAM role created above.

3. Setup AWS Config service

- Launch AWS Config and click Get Started



Simple setup

AWS Config automatically discovers your AWS resources and starts recording configuration changes. You can create Config rules from a set of pre-built managed rules to get started.

[Learn more](#)



Customize rules

You can configure any of the pre-built rules to suit your needs, or create your own rules using AWS Lambda to check configurations for compliance.

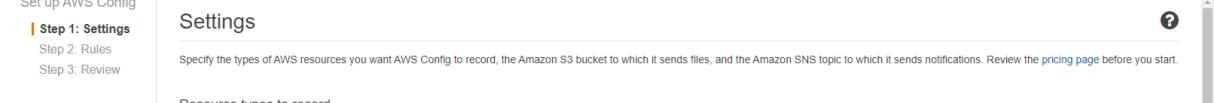
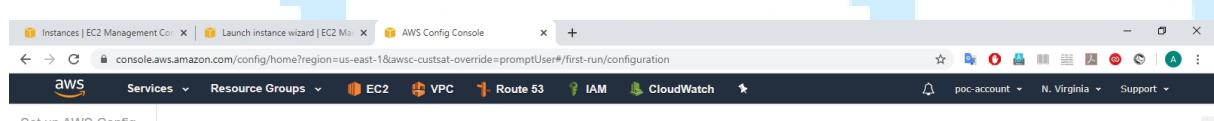
[Learn more](#)



Continuous compliance

AWS Config continuously records configuration changes to resources and automatically evaluates these changes against relevant rules. You can use a dashboard to assess overall configuration compliance.

[Learn more](#)



Security Management in AWS

Amazon SNS topic

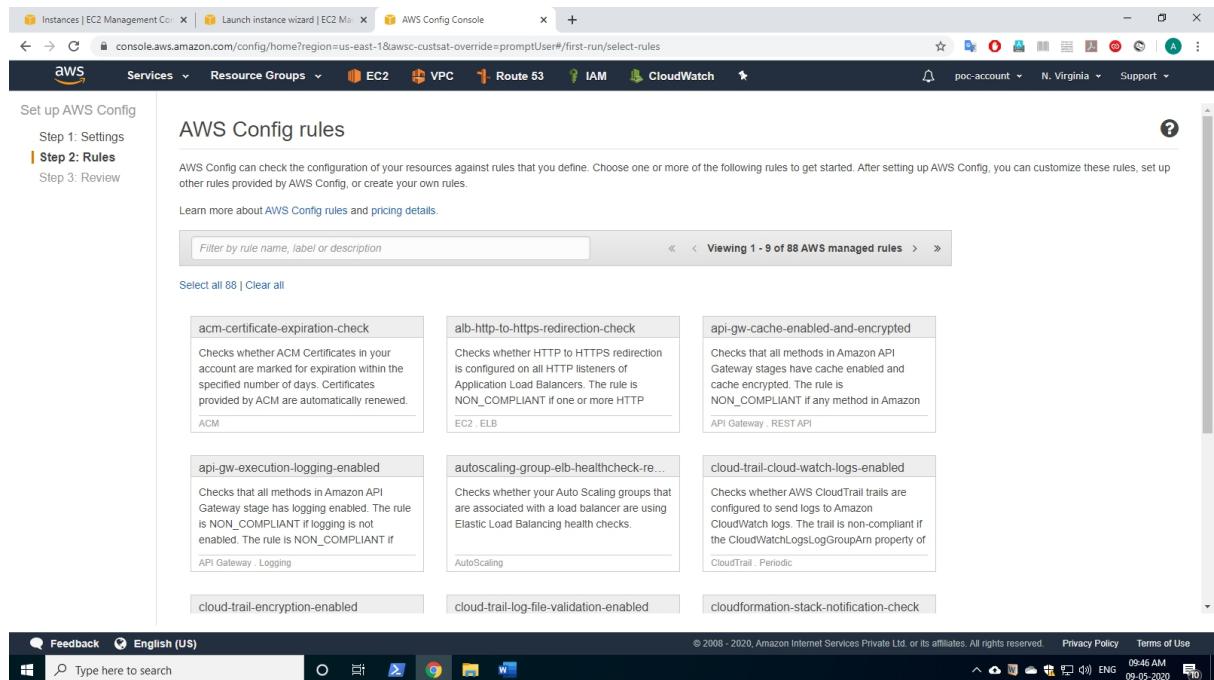
Stream configuration changes and notifications to an Amazon SNS topic.

AWS Config role*

Grant AWS Config read-only access to your AWS resources so that it can record configuration information, and grant it permission to send this information to Amazon S3 and Amazon SNS.

Use an existing AWS Config service-linked role
 Choose a role from your account

- Click **Next** in the *Rules*



Step 1: Settings | Step 2: Rules | Step 3: Review

AWS Config rules

AWS Config can check the configuration of your resources against rules that you define. Choose one or more of the following rules to get started. After setting up AWS Config, you can customize these rules, set up other rules provided by AWS Config, or create your own rules.

Learn more about AWS Config rules and pricing details.

Filter by rule name, label or description

Viewing 1 - 9 of 88 AWS managed rules

Select all 88 | Clear all

acm-certificate-expiration-check	alb-http-to-https-redirection-check	api-gw-cache-enabled-and-encrypted
Checks whether ACM Certificates in your account are marked for expiration within the specified number of days. Certificates provided by ACM are automatically renewed.	Checks whether HTTP to HTTPS redirection is configured on all HTTP listeners of Application Load Balancers. The rule is NON_COMPLIANT if one or more HTTP	Checks that all methods in Amazon API Gateway stages have cache enabled and cache encrypted. The rule is NON_COMPLIANT if any method in Amazon
ACM	EC2 . ELB	API Gateway . REST API
api-gw-execution-logging-enabled	autoscaling-group-elb-healthcheck-re...	cloud-trail-cloud-watch-logs-enabled
Checks that all methods in Amazon API Gateway stage has logging enabled. The rule is NON_COMPLIANT if logging is not enabled. The rule is NON_COMPLIANT if	Checks whether your Auto Scaling groups that are associated with a load balancer are using Elastic Load Balancing health checks.	Checks whether AWS CloudTrail trails are configured to send logs to Amazon CloudWatch logs. The trail is non-compliant if the CloudWatchLogsLogGroupArn property of
API Gateway . Logging	AutoScaling	CloudTrail . Periodic
cloud-trail-encryption-enabled	cloud-trail-log-file-validation-enabled	cloudformation-stack-notification-check

Feedback English (US) Type here to search © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use 09:46 AM ENG 09-05-2020

- Review and click **Confirm**

Security Management in AWS

The screenshot shows the AWS Config Review step 3: Review page. At the top, there are tabs for Step 1: Settings, Step 2: Rules, and Step 3: Review, with Step 3: Review selected. The main area is titled "Review" and contains sections for "AWS Config rules (0)" and "Settings". Under "Settings", there is a "Resource types" section showing "All resources (excluding global resources)". Below it, there are entries for "Amazon S3 bucket" (config-bucket-654615032619) and "AWS Config role" (AWSConfigRoleForConfig). At the bottom right, there are "Cancel", "Previous", and "Confirm" buttons.

- You should see the *Dashboard* now

The screenshot shows the AWS Config Config Dashboard. On the left, a sidebar menu includes options like Dashboard, Conformance packs, Rules, Resources, Aggregated view, and Learn More. The main dashboard has sections for "Resources" (showing 0 total resources), "Config rule compliance" (with a button to "Add rule"), and "Noncompliant rules" (listing 0 noncompliant rules). A message at the top states: "The redesigned AWS Config console is now available for use. We've completely redesigned the console to improve the overall look and feel. Try it out now." The status bar at the bottom indicates the date as 09-05-2020 and the time as 09:46 AM.

- Click *Add rule* in the dashboard. Use search bar to search and find the rule *approved-amis-by-id*

Security Management in AWS

The redesigned AWS Config console is now available for use.
We've completely redesigned the console to improve the overall look and feel. Try it out now.

Add rule

Add rules to define the desired configuration settings of your AWS resources. Customize any of the following rules to suit your needs, or add a custom rule. To add a custom rule, you must create an AWS Lambda function for the rule.

Add custom rule

Viewing 1 - 2 of 2 AWS managed rules

approved

approved-ami-by-id

Checks whether running instances are using specified AMIs. Specify a list of approved AMI IDs. Running instances with AMIs that are not on this list are noncompliant.

EC2

approved-ami-by-tag

Checks whether running instances are using specified AMIs. Specify the tags that identify the AMIs. Running instances with AMIs that don't have at least one of the specified tags

EC2

- Click Next

How would you rate your experience with this service console?

by AWS Config is created, changed, or deleted

matches the specified type, or the type plus identifier, is created, changed, or deleted

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 resources Resource type: Multiple Selected

AWS EC2 Instance

Resource identifier - optional

Enter resource identifier

Parameters

Rule parameters define attributes for which your resources are evaluated; for example, a required tag or S3 bucket.

Key	Value
amilds	ami-098f16afa9edf40be

Add another row

Back **Next**

- Under Parameters section, copy and paste the ami-id you want to evaluate with.
- For this demo, we are evaluating compliance for RHEL 8 image with ami-id *ami-098f16afa9edf40be*.
- To get the AMI ID of your RHEL instance, go to EC2 Dashboard -> Click on Instances -> Select the RHEL instance. In the description section search for AMI ID.
- Choose Remediation Action section -> Remediation Action -> *AWS-StopEC2Instance*
- Select Auto remediation -> Yes

Security Management in AWS

The screenshot shows the AWS IAM Management Console with the 'Rules' section selected. A new rule is being created with the following parameters:

- Key:** amilds
- Value:** ami-098f16afa9edf40be
- Remediation action:** AWS-StopEC2Instance
- Auto remediation:** Yes (selected)
- Retries in:** 5
- Seconds:** 60
- Rate Limits:** Concurrent Execution Rate and Error Rate fields are present but not filled.

- Select *Resource ID parameter -> Instance Id*
- Fill the Value for Key *AutomationAssumeRole* with ARN of the IAM role you created earlier

The screenshot shows the AWS IAM Management Console with the 'Rules' section selected. A rule is being edited with the following parameter configuration:

- Resource ID parameter:** Instanceld
- Parameters:** AutomationAssumeRole* is set to :Config_SSMAutomationRole
- Instanceld*:** An empty value field.
- Required fields:** * Required fields
- Delete remediation action:** Delete rule button
- Delete rule:** Before deleting the rule, ensure to delete remediation action associated with the rule. Deleting the rule removes the rule and its evaluation results. The deletion might take several minutes.
- Save:** Save button

- Under *Rules ->* you can see the rule created just now

Security Management in AWS

The screenshot shows the AWS Config Rules page. A banner at the top says "The redesigned AWS Config console is now available for use. We've completely redesigned the console to improve the overall look and feel. Try it out now." Below this, a table lists a single rule named "approved-amis-by-id". The table has columns for Rule name, Compliance, and Remediation action. The rule is marked as "Noncompliant" and the remediation action is "AWS-StopEC2instance".

- Click the rule name, you can see the Action status for the noncompliant resource. If it shows "Action executed successfully" it means the remediation action is already taken place and the noncompliant EC2 instance is stopped.

The screenshot shows the "Rules > Rule details" page for the rule "approved-amis-by-id". It displays the rule configuration, including trigger type (Configuration changes), scope (Resources, EC2 Instance), and auto remediation settings. The "Overall rule status" shows the last successful invocation on May 9, 2020, at 2:45:05 PM, with an "Action executed successfully" status. The "Choose resources in scope" section shows a single EC2 instance (i-095c2d377faca8e1b) listed as noncompliant.

- Go to EC2 console and verify the non-compliant EC2 instance is indeed stopped by *AWS Config Remediation Action*. Here the Ubuntu instance is non-compliant EC2 instance, so it is stopped automatically.

Security Management in AWS

The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with navigation links like EC2 Dashboard, Instances, AMIs, and EBS. The main area displays a table of instances. One instance, 'Ubuntu-Instance', is highlighted and shown in a detailed view below. This view includes tabs for Description, Status Checks, Monitoring, and Tags. Under the Description tab, it shows details such as Instance ID (i-095c2d377faca8e1b), Instance state (stopped), Instance type (t2.micro), and Availability zone (us-east-1b). It also lists network interfaces (Private DNS: ip-172-31-83-85.ec2.internal, Private IP: 172.31.83.85, VPC ID: vpc-00f2e57a, Subnet ID: subnet-f3d31252, Network interfaces: eth0) and platform details (Platform details: Linux/UNIX, Usage operation: RunInstances).

4. Identify who launched the non-compliant instance

- Navigate to Open CloudTrail -> Event History
- Filter Event Name -> RunInstances
- Here you can see all the instances launched and who launched them. In below screenshot, you can see the entries for RHEL and Ubuntu instances launched and both of them are launched by user "root".

The screenshot shows the AWS CloudTrail Management Console. On the left, there's a sidebar with links for CloudTrail, Dashboard, Event history, Insights, Trails, Pricing, Documentation, Forums, and FAQs. The main area is titled 'Event history' and shows a table of events. The table has columns for Event time, User name, Event name, Resource type, and Resource name. Two entries are listed: one from 2020-05-09 at 12:10:49 PM by user 'root' (Event name: RunInstances, Resource type: EC2 VPC and 6 more, Resource name: vpc-00f2e57a and 7 more), and another from 2020-05-09 at 12:08:45 PM by user 'root' (Event name: RunInstances, Resource type: EC2 VPC and 6 more, Resource name: vpc-00f2e57a and 7 more). A note at the bottom says 'No more events'.

Requirement 2:

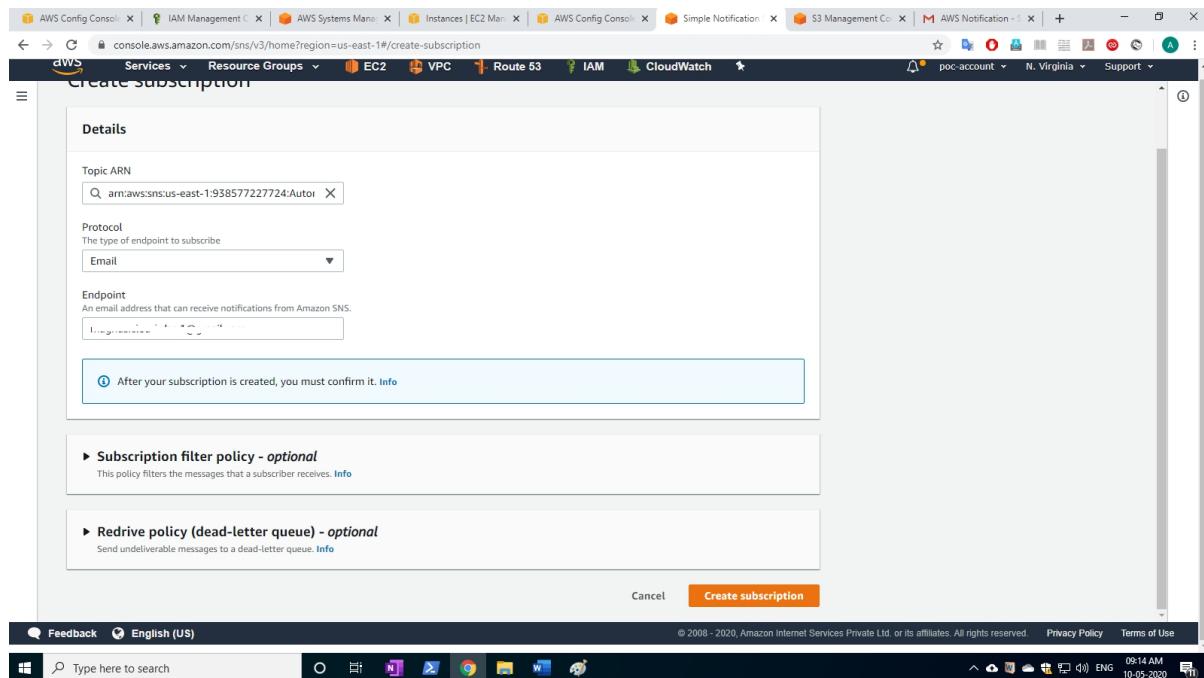
1. Create an SNS topic to send email to security team for non-compliance
 - Open *Amazon SNS* -> provide topic name as “Automation-Config-Notification” -> click *Create topic*
 - The SNS topic is successfully created as shown below.

[Note: The topic name should start with “Automation” for IAM role to be able to publish the notification as per the default permissions provided]

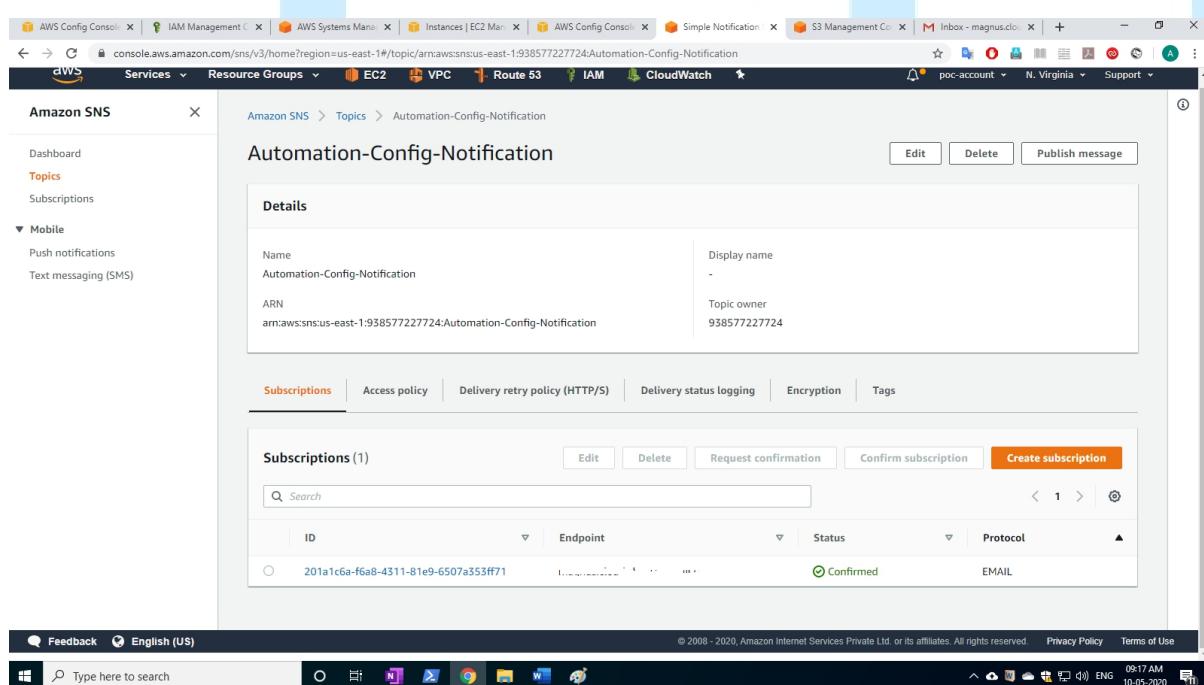
The screenshot shows the AWS SNS console with a topic named "Automation-Config-Notification". The "Details" section shows the Name as "Automation-Config-Notification" and the ARN as "arn:aws:sns:us-east-1:938577227724:Automation-Config-Notification". The "Subscriptions" tab is selected, showing "Subscriptions (0)". A "Create subscription" button is visible. The browser address bar shows the URL: https://console.aws.amazon.com/sns/v3/home?region=us-east-1#topic/arn:aws:sns:us-east-1:938577227724:Automation-Config-Notification

- Click *Create subscription* -> Select *Protocol as Email*
- *Endpoint* -> Your email id
- Click *Create Subscription*

Security Management in AWS



- You are will receive a notification in your email for confirmation. Click *Confirm subscription* in your email
- With this your SNS topic is fully configured and ready to send notification to your email id.



2. Create AWS Config rule for non-compliance

- AWS Config -> Rules -> Add rule -> choose *s3-bucket-public-read-prohibited*

Security Management in AWS

The redesigned AWS Config console is now available for use.
We've completely redesigned the console to improve the overall look and feel. Try it out now.

Add rule

Add rules to define the desired configuration settings of your AWS resources. Customize any of the following rules to suit your needs, or add a custom rule. To add a custom rule, you must create an AWS Lambda function for the rule.

Add custom rule

s3-bucket-public-read Viewing 1 - of 1 AWS managed rules

s3-bucket-public-read-prohibited
Checks that your S3 buckets do not allow public read access. If an S3 bucket policy or bucket ACL allows public read access, the bucket is noncompliant.

S3_Zelkova

- You will see a page called "Add AWS managed rule"

Name* s3-bucket-public-read-prohibited
A unique name for the rule. 128 characters max. No special characters or spaces.

Description Checks that your S3 buckets do not allow public read access. If an S3 bucket policy or bucket ACL allows public read access, the bucket is noncompliant.

Managed rule name S3_BUCKET_PUBLIC_READ_PROHIBITED

Trigger

AWS Config evaluates resources when the trigger occurs.

Trigger type* Configuration changes Periodic

Scope of changes* Resources Tags All changes

Resources* S3 Bucket
Resource identifier (optional)

- Choose *Remediation action* -> Select *AWS-PublishSNSNotification*
- Auto Remediation* -> Yes

Security Management in AWS

The screenshot shows the AWS Config Console interface. On the left sidebar, under the 'Rules' section, there is a 'Remediation actions' section with options like 'Advanced query', 'Settings', 'Authorizations', 'Aggregated view', 'Rules', 'Resources', and 'Aggregators'. Below this is a 'What's new' section. The main content area is titled 'Choose remediation action' and includes fields for 'Frequency' (set to '24 hours'), 'Remediation action' (set to 'AWS-PublishSNSNotification'), 'Auto remediation' (set to 'Yes'), 'Rate Limits' (with 'Concurrent Execution Rate' and 'Error Rate' fields), and a 'Resource ID parameter' dropdown set to 'n/a'. A detailed description of parameters is provided at the bottom. The top navigation bar shows tabs for 'AWS Config Console', 'Simple Notification Service', 'IAM Management Console', and the current page 'AWS Config'. The bottom status bar shows the date and time.

- Fill the Parameters as follows

- *AutomationAssumeRole* -> The arn of the IAM role you created previously.
- *Message* -> Any custom message you want to receive for non-compliance
- *TopicArn* -> the arn of the SNS topic you created previously
- Click on "Save" option

The screenshot shows the same AWS Config Console interface as before, but with the remediation rule parameters filled in. In the 'Parameters' section, three fields are defined:

- Key: AutomationAssumeRole Value: arn:aws:iam:938577227724
- Key: Message Value: S3 Public Read Access Enat
- Key: TopicArn Value: arn:aws:sns:us-east-1:938577227724

The status bar at the bottom indicates the date and time as 08:50 AM 10-05-2020.

- Navigate *Settings -> Resource types to record -> All resources*
- Check *include Global resources*, and then click *Save*

Security Management in AWS

The screenshot shows the AWS Config Console with the 'Settings' tab selected. Under 'Recording is on', there is a 'Turn off' button. The 'Resource types to record' section allows selecting 'All resources' (checked) or 'Specific types'. The 'Data retention period' is set to 7 years. The 'Amazon S3 bucket*' section shows options to 'Create a bucket', 'Choose a bucket from your account' (selected), or 'Choose a bucket from another account'. The browser status bar at the bottom indicates it's from console.aws.amazon.com.

3. Create a S3 bucket with public read/write permissions

- Navigate S3 -> *Create bucket*
- *Bucket Name* -> any unique name as bucket names are unique across all AWS customers
- *Region* -> us-east-1
- Under *Bucket Settings for Public Access*, uncheck *Block all public access*
- Click *I acknowledge* and *Create Bucket*

The screenshot shows the AWS S3 Management Console with the 'Create bucket' wizard open. In the 'General configuration' step, a bucket name 'edureka-test-bucket-12345' is entered, and the region is set to 'US East (N. Virginia) us-east-1'. In the 'Bucket settings for Block Public Access' step, the 'Block all public access' checkbox is unchecked. Other options like 'Block public access to buckets and objects granted through new access control lists (ACLs)' and 'Block public access to buckets and objects granted through any access control lists (ACLS)' are also shown. The browser status bar at the bottom indicates it's from s3.console.aws.amazon.com.

Security Management in AWS

The screenshot shows the AWS S3 Management Console with a new bucket creation wizard. Under the 'Public access' section, the 'Block all public access' checkbox is selected. A warning message states: 'Turning off block all public access might result in this bucket and the objects within becoming public. AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.' Below this, a checkbox for acknowledging the risk is checked. At the bottom right of the wizard is a large orange 'Create bucket' button.

- Navigate to the newly created bucket -> Permissions -> Access Control List -> Public Access -> Check Everyone
- Check Access to the objects -> click List Objects -> Click Save

The screenshot shows the AWS S3 Management Console on the 'Permissions' tab for the 'edureka-test-bucket-12345'. In the 'Access for bucket owner' section, the 'Canonical ID' row for the user's account has 'List objects' checked. In the 'Access for other AWS accounts' section, there is a 'Everyone' group entry with 'List objects' checked. A modal dialog box titled 'Everyone' is open, showing the same permission settings. The 'List objects' checkbox is checked and highlighted in blue. At the bottom right of the dialog are 'Cancel' and 'Save' buttons.

- Now the bucket is enabled with public read access

4. Verify the compliance and remediation action with AWS Config

- Go back to AWS Config Dashboard. In the navigation pane, select Rules and select the S3 rule that you had created earlier.
- Once the rule opens, click Re-evaluate

Security Management in AWS

The screenshot shows the AWS Config Console interface. On the left, there's a navigation sidebar with sections like AWS Config, Dashboard, Conformance packs, Rules (which is selected), Resources, Advanced query, Settings, Authorizations, Aggregated view, Rules, Resources, Aggregators, and What's new. The main content area displays a rule titled 's3-bucket-public-read-prohibited'. The rule details include:

- Description: Checks that your S3 buckets do not allow public read access. If an S3 bucket policy or bucket ACL allows public read access, the bucket is noncompliant.
- Trigger type: Configuration changes, 24 hours.
- Scope of changes: Resources.
- Resource types: S3 Bucket.
- Auto remediation: On, Retry 5 times in 60 seconds.
- Config rule ARN: arn:aws:config:us-east-1:938577227724:config-rule/config-rule-xdl5gp.
- Parameters: null.
- Overall rule status: Last successful invocation on May 10, 2020 at 9:44:23 AM (green checkmark).
- Last successful evaluation on May 10, 2020 at 9:44:24 AM (green checkmark).

Below the rule details, there's a section titled 'Choose resources in scope' with a table:

Resource ID	Resource type	Resource compliance status	Action status
edureka-test-bucket-12345	S3 Bucket	Noncompliant	n/a

The bottom of the screenshot shows a Windows taskbar with various icons and the system tray indicating the date and time as 10-05-2020, 09:48 AM.

- Wait for a minute and then refresh. You should see the non-compliant resource as below.

This screenshot is identical to the one above, showing the 's3-bucket-public-read-prohibited' rule in the AWS Config Console. The rule details and the table in the 'Choose resources in scope' section remain the same. The table shows a single row for the S3 bucket 'edureka-test-bucket-12345' with a 'Noncompliant' status under 'Resource compliance status'. The bottom of the screenshot shows a Windows taskbar with various icons and the system tray indicating the date and time as 10-05-2020, 09:49 AM.

- Wait for another minute and notice Action Status changed to *Action executed successfully*.

Security Management in AWS

The screenshot shows the AWS Config Rule Details page for a rule named "s3-bucket-public-read-prohibited". The rule checks if S3 buckets allow public read access. It has a trigger type of "Configuration changes: 24 hours", a scope of "Resources", and applies to "S3 Bucket". The auto remediation is set to "On, Retry 5 times in 60 seconds". The config rule ARN is "arn:aws:config:us-east-1:938577227724:config-rule/config-rule-xdl5gp". The overall rule status is "Last successful invocation on May 10, 2020 at 9:49:12 AM" and the last successful evaluation was on the same day at 9:49:09 AM. Below this, there's a table titled "Choose resources in scope" showing one resource: "edureka-test-bucket-12345" which is an S3 Bucket and is listed as "Noncompliant". A remediation button is visible.

- It means the Remediation Action successfully executed and you should have received email notification as below in email mailbox.



- Navigate to AWS Systems Manager -> Automation -> Execution ID

The screenshot shows the AWS Systems Manager Automation page. The left sidebar is collapsed, and the main area displays the title "AWS Systems Manager" with the tagline "Gain Operational Insight and Take Action on AWS Resources." A prominent orange button says "Get Started with Systems Manager". Below this, a section titled "How it works" shows a diagram of a server interacting with AWS services. A "More resources" link is also present.

- You should be able to see the details of Remediation Automation run by Systems Manager which was invoked by AWS Config.

Security Management in AWS

The screenshot shows the AWS Systems Manager Automation Execution detail page. The execution ID is 84e21642-feca-4aa8-913c-7a1f28df994b. The execution status is Success, with 1 step succeeded, 0 failed, 0 cancelled, and 0 timed out. The execution mode is Auto. The document version is 1. The execution started at 04:20:06 GMT on May 10, 2020, and ended at 04:20:07 GMT on the same day.

5. Identify who caused non-compliance using CloudTrail

- Navigate *CloudTrail* -> *Event History* -> Filter *Event Name: CreateBucket*
- As you can see, the bucket in question was created by the user “root”

The screenshot shows the CloudTrail Management Console Event history page. A single event is listed: a CreateBucket operation performed by user 'root' on 2020-05-10 at 08:11:21 AM. The event details include the AWS access key, region (us-east-1), error code, event ID, and source IP address. The resource type is S3 Bucket, and the resource name is edureka-test-bucket-12345.

Event time	User name	Event name	Resource type	Resource name
2020-05-10, 08:11:21 AM	root	CreateBucket	S3 Bucket	edureka-test-bucket-12345

Clean Up

Delete the resources that you have created in the demo.

- This is the end of the lab.

edureka!