

Introduction to Cloud Computing — Security & Identity Management (IAM)

Author: Oluwaseun Osunsola

Environment: AWS

Project Link: [https://github.com/Oluwaseunoa/DevOps-Projects/blob/main/Cloud%20Computing/Security%20%26%20Identity%20Management%20\(IAM\)Security%20%26%20Identity%20Management%20\(IAM\)/README.mdes](https://github.com/Oluwaseunoa/DevOps-Projects/blob/main/Cloud%20Computing/Security%20%26%20Identity%20Management%20(IAM)Security%20%26%20Identity%20Management%20(IAM)/README.mdes)

1. Introduction

This project focuses on **Amazon Web Services (AWS) Identity and Access Management (IAM)** and its application in securing cloud resources for Zappy e-Bank, a fintech startup.

The objective is to learn how IAM can help enforce **authentication and authorization** for users, ensuring that sensitive financial data is properly secured.

Through this project, I gained hands-on experience in creating **policies, groups, users, and multi-factor authentication (MFA)**, while applying the **principle of least privilege**.

2. Importance of IAM for Zappy e-Bank

For a fintech company like Zappy e-Bank, **security and compliance** are critical. IAM helps to:

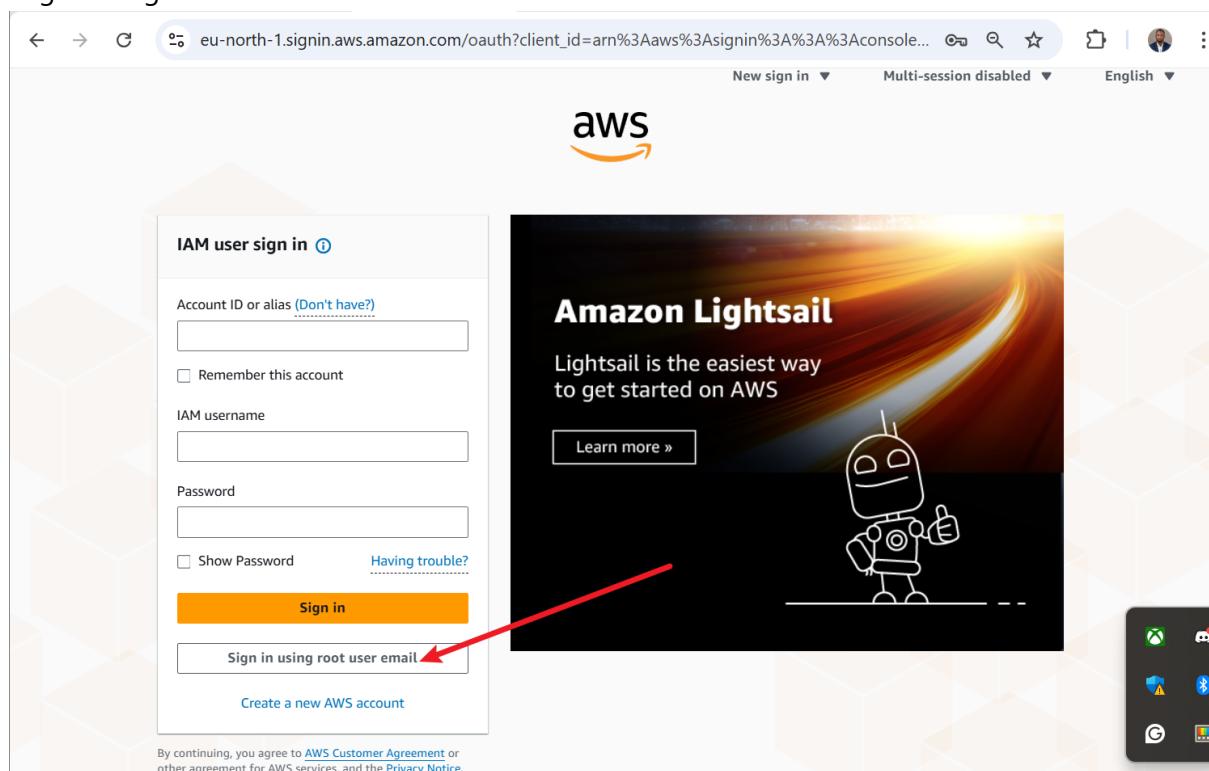
- Create and manage AWS **users and groups**.
- Assign fine-grained **permissions** through IAM **policies**.
- Implement **multi-factor authentication (MFA)** for stronger access control.
- Enforce the **principle of least privilege**, ensuring users only have access to what they need.

This reduces the risk of insider threats, unauthorized access, and compliance violations.

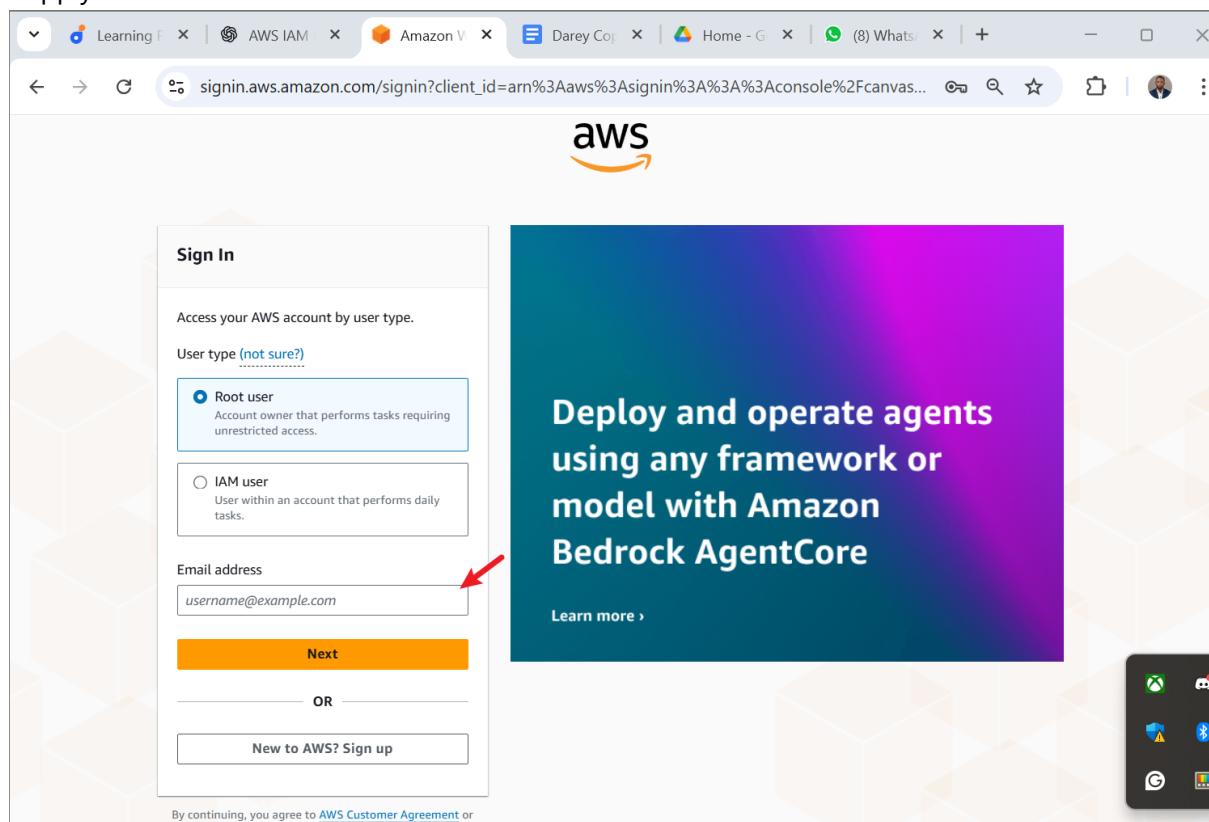
3. Project Setup

1. Logged into **AWS Management Console** using the administrator account.

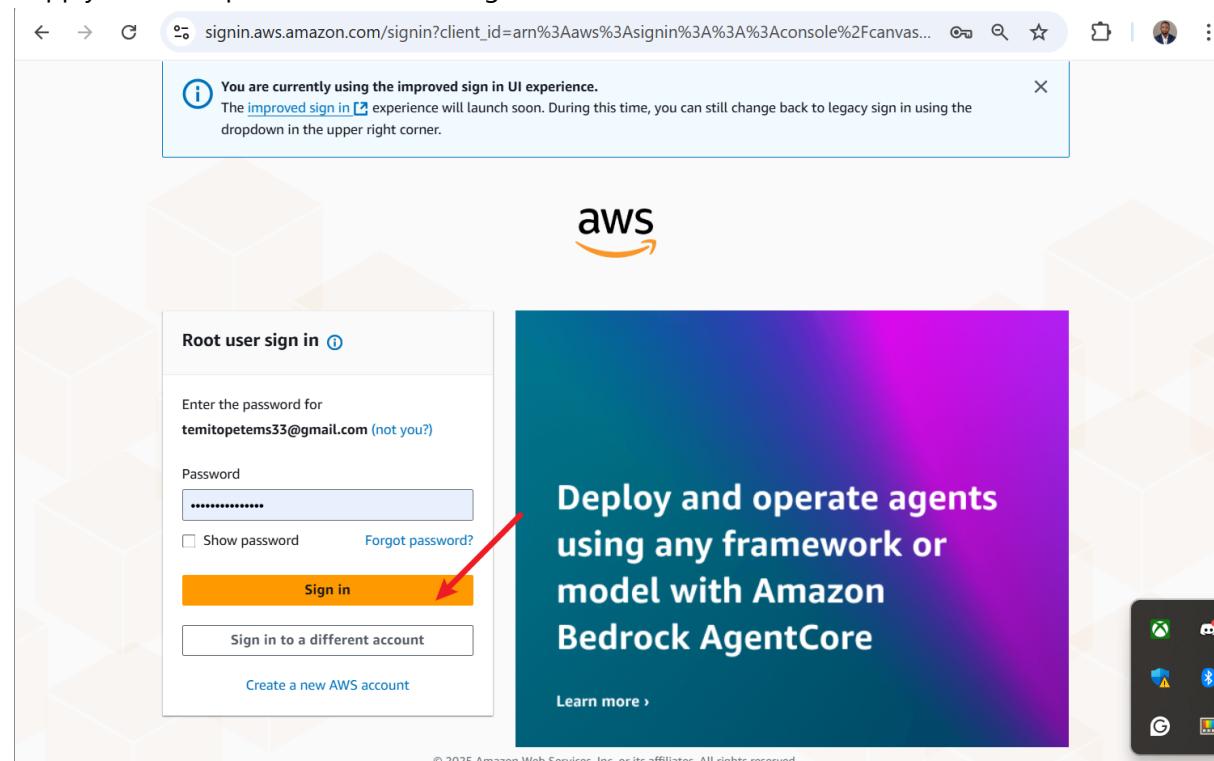
- Log in using root email.



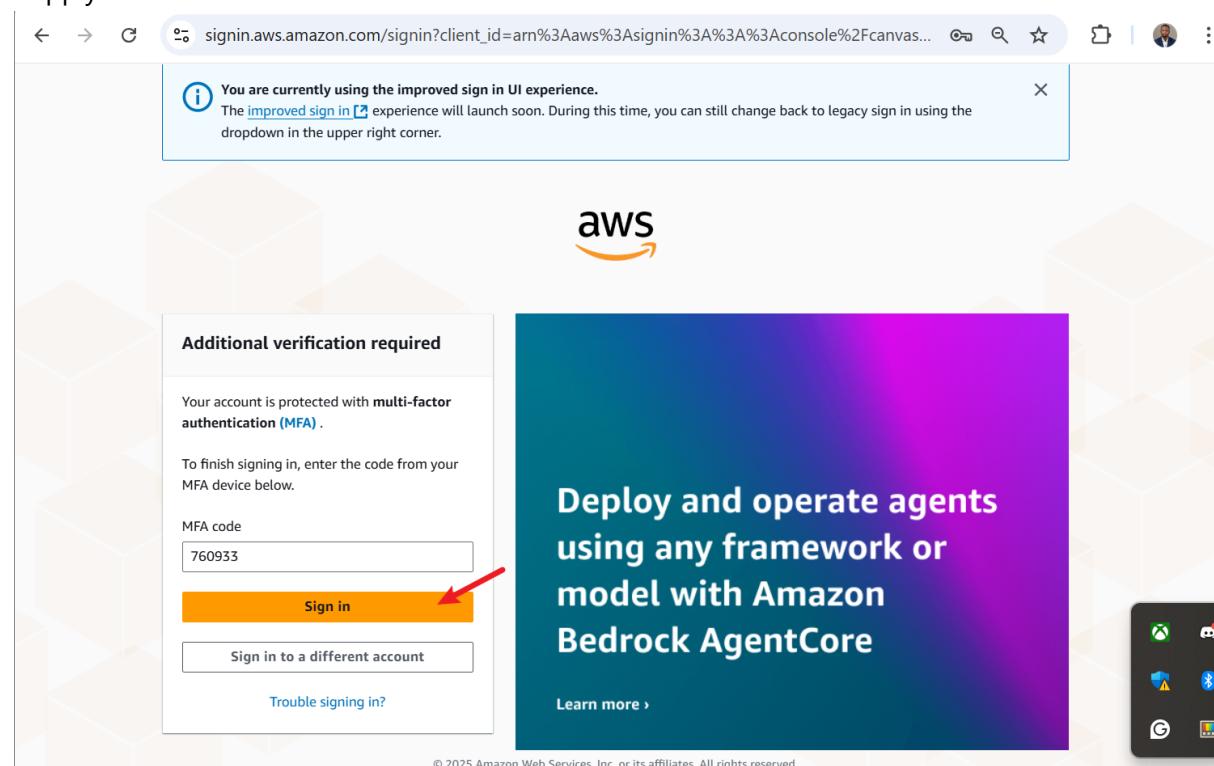
- Supply root user email.



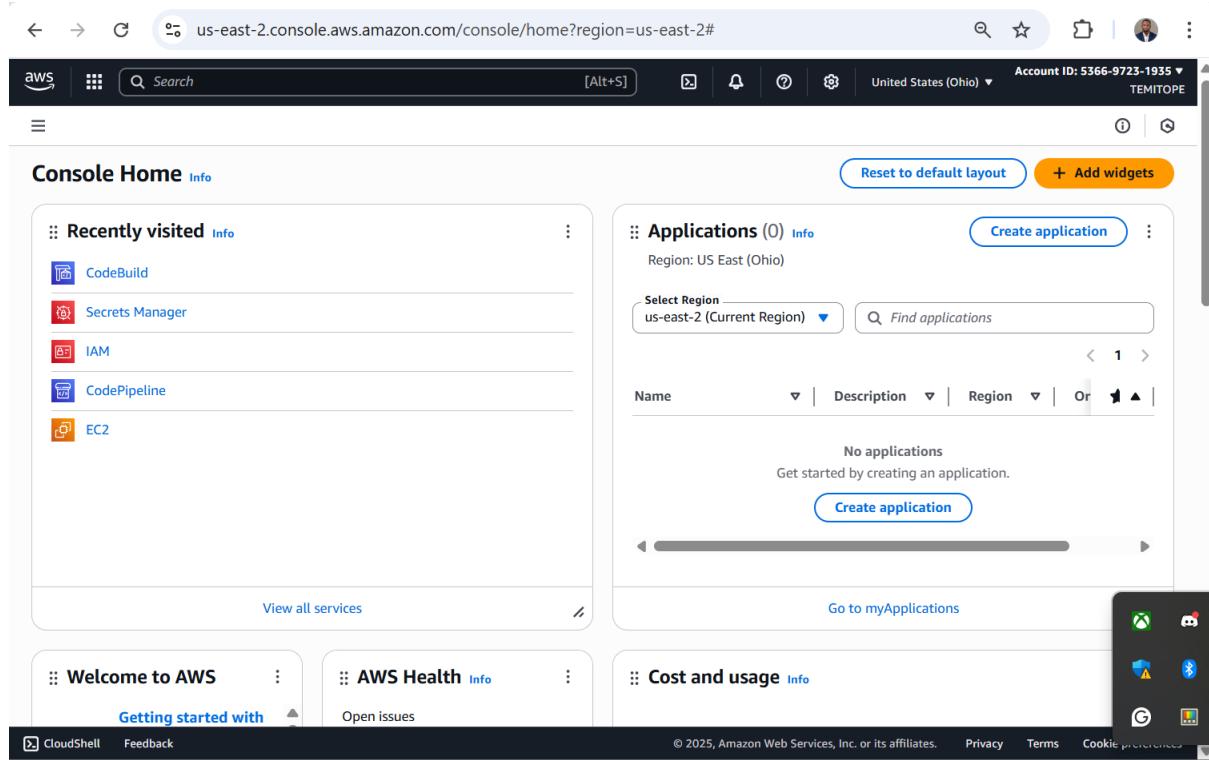
- Supply root user password and click sign in.



- Supply MFA code for additional verification.

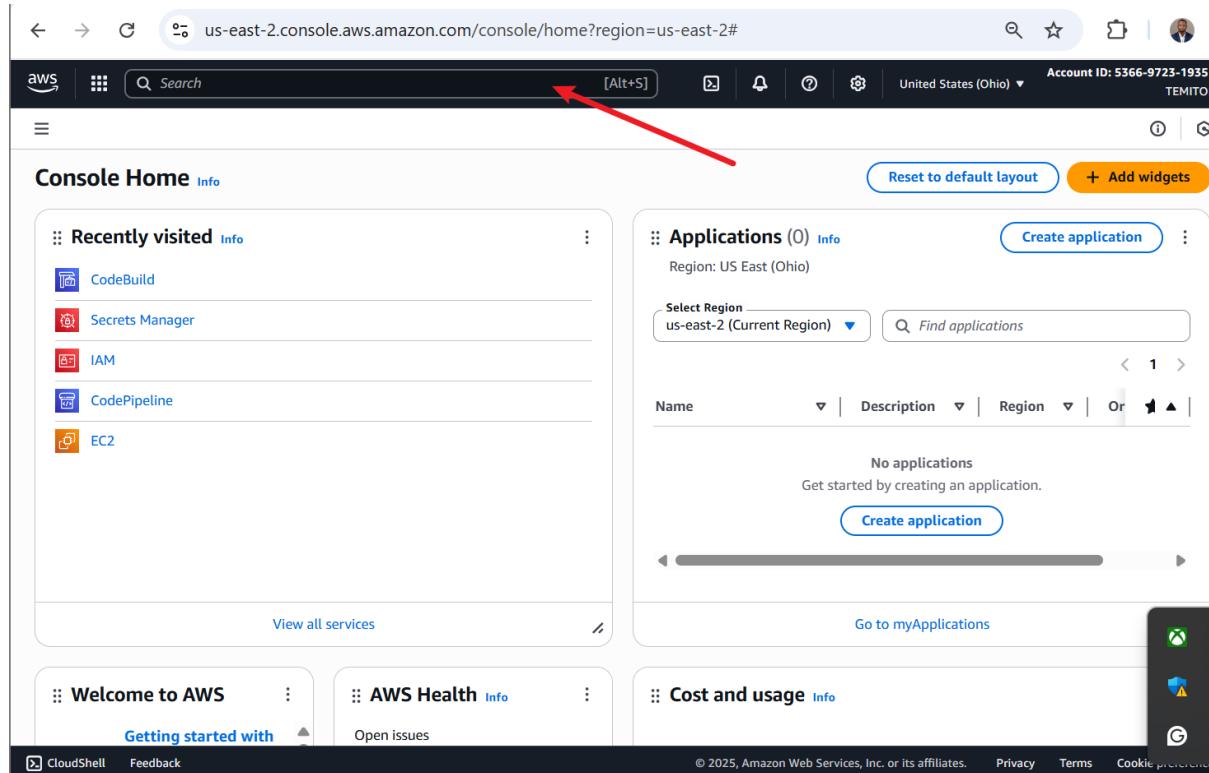


- Landed on the console dashboard.

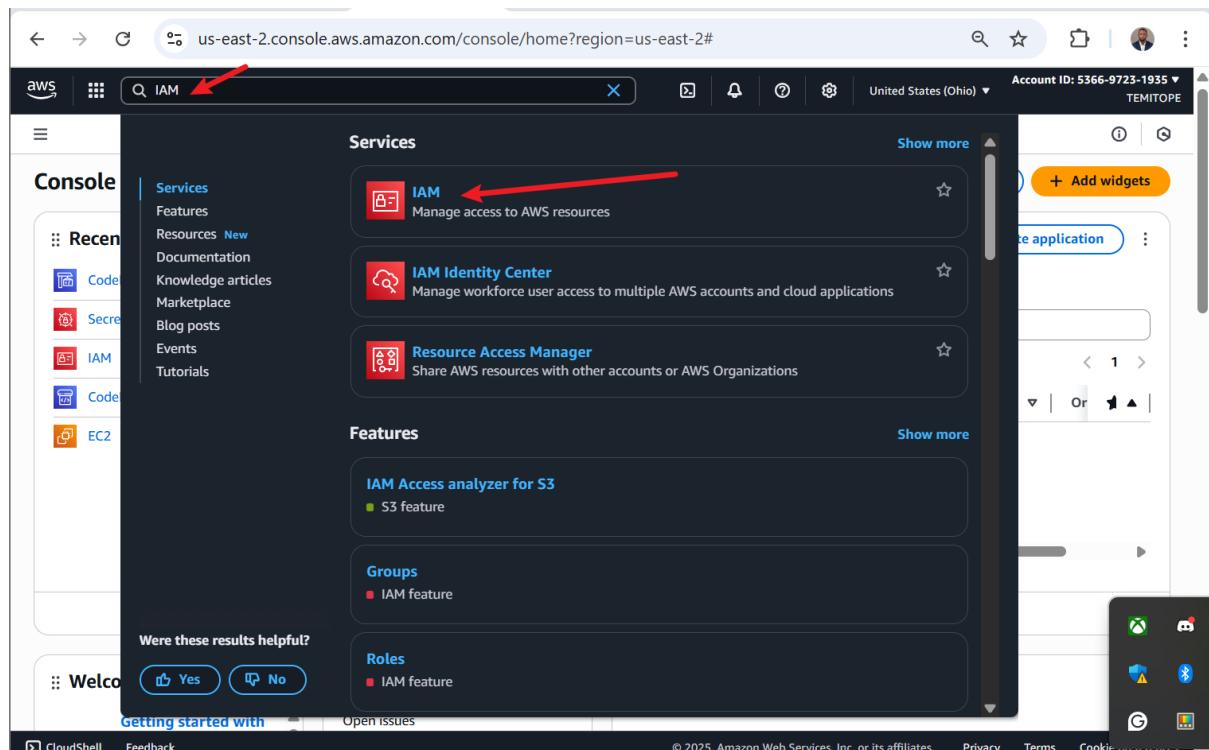


2. Navigated to the IAM Dashboard.

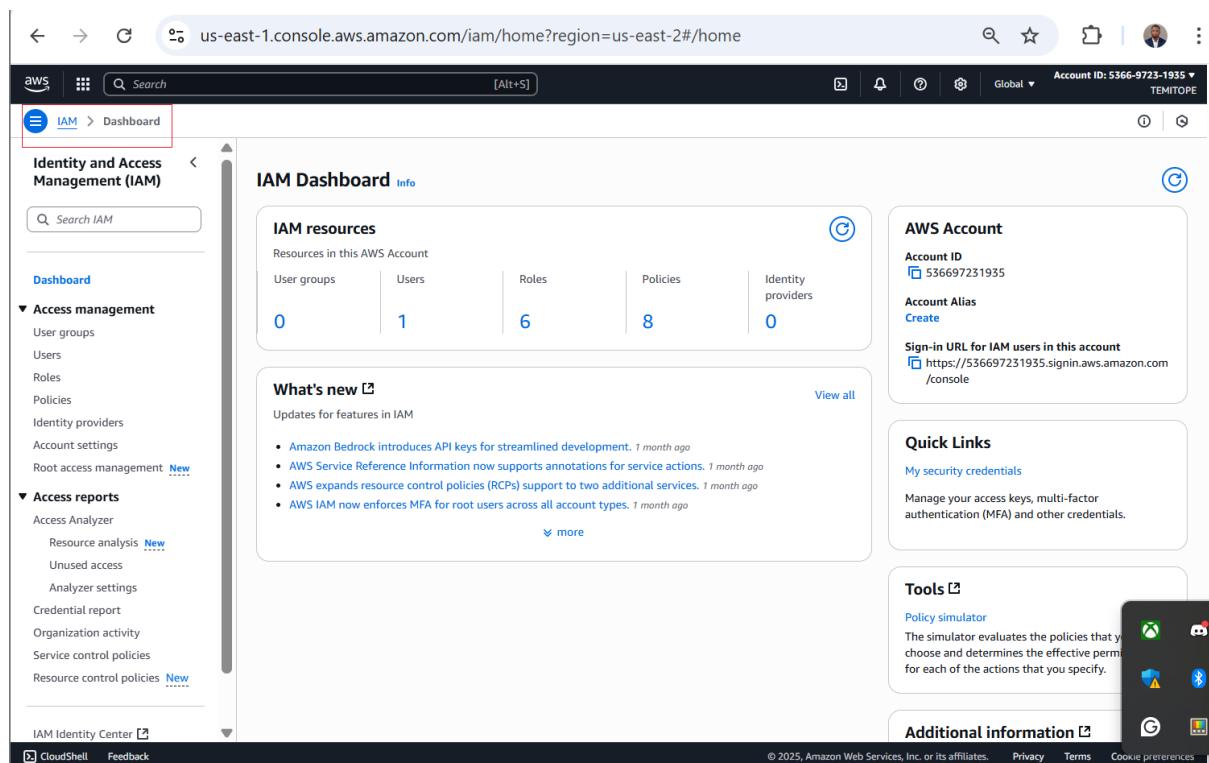
- Use the service search.



- Search for IAM and click on it.



- Now on IAM dashboard.



4. Creating IAM Policies

Developer Policy (EC2 Access)

- Created a **customer-managed policy** for backend developers.
 - Service: **EC2**.
 - Actions: **All EC2 actions**.
 - Resources: **All**.

- Policy Name: developer-policy.

Steps:

- Click on policies.

The screenshot shows the AWS IAM Dashboard. On the left sidebar, under 'Access management', the 'Policies' link is highlighted with a red arrow. The main content area displays the 'IAM Dashboard' with sections for 'IAM resources' (User groups: 0, Users: 1, Roles: 6, Policies: 8, Identity providers: 0) and 'What's new' (listing recent changes like Amazon Bedrock API keys and AWS Service Reference Information annotations). A sidebar on the right provides account information (Account ID: 536697231935, Account Alias: Create), quick links for security credentials and MFA, tools for policy simulation, and additional information.

- Click on create policy.

The screenshot shows the 'Policies' page with 1387 results. The left sidebar shows the 'Policies' link is also highlighted with a red arrow. The main table lists various AWS managed policies with their descriptions. At the top right of the table, there is a 'Create policy' button, which is highlighted with a yellow oval and a red arrow. The right sidebar contains links for Policy simulator, Tools, and Additional information.

Policy name	Type	Used as	Description
AccessAnalyzerServiceRolePolicy	AWS managed	None	Allow Access Analyzer to analyze resources
AdministratorAccess	AWS managed - job function	Permissions policy (1)	Provides full access to AWS services and resources
AdministratorAccess-Amp...	AWS managed	None	Grants account administrative permissions
AdministratorAccess-AWS...	AWS managed	None	Grants account administrative permissions
AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly permissions required by AI
AIOpsConsoleAdminPolicy	AWS managed	None	Grants full access to Amazon AI Operations
AIOpsOperatorAccess	AWS managed	None	Grants access to the Amazon AI Operations
AIOpsReadOnlyAccess	AWS managed	None	Grants ReadOnly permissions to the AI
AlexaForBusinessDeviceSetup...	AWS managed	None	Provide device setup access to Alexa for Business
AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness
AlexaForBusinessGateway...	AWS managed	None	Provide gateway execution for Alexa
AlexaForBusinessLifesize...	AWS managed	None	Provide access to Lifesize AVS
AlexaForBusinessNetwork...	AWS managed	None	This policy enables Alexa for Business
AlexaForBusinessPolyDelete...	AWS managed	None	Provide access to Poly AVS

- Under select service section, choose EC2 service.

Screenshot of the AWS IAM Policy Editor. The left sidebar shows 'Step 1 Specify permissions' and 'Step 2 Review and create'. The main area is titled 'Specify permissions' with a 'Policy editor' tab selected. Under 'Select a service', there is a dropdown menu labeled 'Choose a service' with a red arrow pointing to its dropdown icon. At the bottom of the screen, there is a navigation bar with 'CloudShell', 'Feedback', '© 2025, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

- Under action, check all EC2.

Screenshot of the AWS IAM Policy Editor. The left sidebar shows 'Step 1 Specify permissions' and 'Step 2 Review and create'. The main area is titled 'Specify permissions' with a 'Policy editor' tab selected. Under 'EC2', the 'Allow All actions' section is expanded. In the 'Actions allowed' section, there is a list of actions under 'Manual actions | Add actions'. A red arrow points to the checkbox for 'All EC2 actions (ec2:*)'. On the right side, there is an 'Effect' section with 'Allow' checked. At the bottom, there is a warning message: 'Dependent permissions not selected. To grant permissions for the selected resource actions, including additional dependent actions might be required.' with two bullet points: 'ec2:AcceptAddressTransfer requires 1 more action.' and 'ec2:AllocateAddress requires 1 more action.'. The bottom navigation bar includes 'CloudShell', 'Feedback', '© 2025, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

- Under resources, check all.

The screenshot shows the 'Create policy' wizard on the AWS IAM console. In the 'Resources' section, there is a note: 'Specify resource ARNs for these actions.' Below it are two radio buttons: 'All' (selected) and 'Specific'. A red arrow points to the 'All' button. To the right of the radio buttons is a list of service names (capacity-block, capacity-reservation, etc.) each with an 'Info' link and a note about ARN specification. At the bottom of the 'Resources' section is a note: 'The all wildcard '*' may be overly permissive for the selected actions. Allowing specific ARNs for these service resources can improve security.'

- Having checked all, click next.

This screenshot continues from the previous one, showing the 'Create policy' wizard after selecting 'All' for resources. The 'Resources' section now lists many actions requiring more permissions (e.g., ec2:CreateLaunchTemplate, ec2:CreateLaunchTemplateVersion, etc.). A red arrow points to the 'All' radio button. At the bottom right of the page, a large red arrow points to the 'Next' button.

- Name policy and click create policy.

Review and create Info

Step 1
Specify permissions

Step 2
Review and create

Policy details

Policy name
Enter a meaningful name to identify this policy.

Maximum 128 characters. Use alphanumeric and '+', '=', '@', '-' characters.

Description - optional
Add a short explanation for this policy.

Permissions defined in this policy Info

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it.

Allow (1 of 449 services)

Service	Access level	Resource	Request condition
EC2	Full access	All resources	None

Show remaining 448 services

Add tags - optional Info

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag
You can add up to 50 more tags.

Create policy

- Developer policy successfully created.

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

us-east-1.console.aws.amazon.com/iam/home?region=us-east-2#/policies

Identity and Access Management (IAM)

Policies (1388) Info

Policy developer-policy created.

Actions Delete Create policy

Filter by Type All types

Policy name Type Used as Description

AccessAnalyzerServiceRolePolicy	AWS managed	None	Allow Access Analyzer to analyze resou.
AdministratorAccess	AWS managed - job function	Permissions policy (1)	Provides full access to AWS services an.
AdministratorAccess-Amplify	AWS managed	None	Grants account administrative permiss.
AdministratorAccess-AWSElasticBeanstalk	AWS managed	None	Grants account administrative permiss.
AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly permissions require.
AIOpsConsoleAdminPolicy	AWS managed	None	Grants full access to Amazon AI Opera..
AIOpsOperatorAccess	AWS managed	None	Grants access to the Amazon AI Opera..
AIOpsReadOnlyAccess	AWS managed	None	Grants ReadOnly permissions to the A..
AlexaForBusinessDeviceSetup	AWS managed	None	Provide device setup access to AlexaFo.
AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness ..
AlexaForBusinessGatewayExecution	AWS managed	None	Provide gateway execution access to Al.
AlexaForBusinessLifesizeDelegatedAccess...	AWS managed	None	Provide access to Lifesize AVS devices
AlexaForBusinessNetworkProfileServicePo...	AWS managed	None	This policy enables Alexa for Business ..
AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None	Provide access to Poly AVS devices
AlexaForBusinessReadonlyAccess	AWS managed	None	Provide read only access to AlexaForBu.
AmazonAPIGatewayAdministrator	AWS managed	None	Provides full access to create/edit/dele.
AmazonAPIGatewayInvokeFullAccess	AWS managed	None	Provides full access to invoke APIs in A.
AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	None	Allows API Gateway to push logs to us..
AmazonAppFlowFullAccess	AWS managed	None	Provides full access to Amazon AppFl.
AmazonTextractDelegatedAccessPolicy	AWS managed	None	Provide read only access to Amazon T.

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

Analyst Policy (S3 Access)

- Created a **customer-managed policy** for data analysts.
 - Service: **S3**.
 - Actions: **All S3 actions**.
 - Resources: **All**.
 - Policy Name: **analyst-policy**.

Steps:

- Click create policy to create another one.

The screenshot shows the AWS IAM Policies page. The URL is `us-east-1.console.aws.amazon.com/iam/home?region=us-east-2#/policies`. A green banner at the top says "Policy developer-policy created." Below it, a table lists 1388 policies. The "Create policy" button is highlighted with a red arrow. The left sidebar shows navigation options like Identity and Access Management (IAM), Access management, Access reports, and IAM Identity Center.

Policy name	Type	Used as	Description
AccessAnalyzerServiceRolePolicy	AWS managed	None	Allow Access Analyzer to analyze resou.
AdministratorAccess	AWS managed - job function	Permissions policy (1)	Provides full access to AWS services an.
AdministratorAccess-Amplify	AWS managed	None	Grants account administrative permis.
AdministratorAccess-AWSElasticBeanstalk	AWS managed	None	Grants account administrative permis.
AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly permissions require.
AIOpsConsoleAdminPolicy	AWS managed	None	Grants full access to Amazon AI Opera..
AIOpsOperatorAccess	AWS managed	None	Grants access to the Amazon AI Opera..
AIOpsReadOnlyAccess	AWS managed	None	Grants ReadOnly permissions to the A..
AlexaForBusinessDeviceSetup	AWS managed	None	Provide device setup access to AlexaFo.
AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness .
AlexaForBusinessGatewayExecution	AWS managed	None	Provide gateway execution access to Al.
AlexaForBusinessLifesizeDelegatedAccess...	AWS managed	None	Provide access to Lifesize AVS devices
AlexaForBusinessNetworkProfileServicePo...	AWS managed	None	This policy enables Alexa for Business ..
AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None	Provide access to Poly AVS devices
AlexaForBusinessReadOnlyAccess	AWS managed	None	Provide read only access to AlexaForBu.
AmazonAPIGatewayAdministrator	AWS managed	None	Provides full access to create/edit/delete.
AmazonAPIGatewayInvokeFullAccess	AWS managed	None	Provides full access to invoke APIs in A.
AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	None	Allows API Gateway to push logs to us..
AmazonAppFlowFullAccess	AWS managed	None	Provides full access to Amazon AppFl.
AmazonAppFlowReadOnlyAccess	AWS managed	None	Provides read-only access to AmazonAppF.

- Under select a service, choose S3.

The screenshot shows the 'Specify permissions' step of an IAM policy creation. The 'Service' dropdown is open, displaying the placeholder 'Choose a service'. The 'Visual' tab is selected in the top right corner. At the bottom right are 'Cancel' and 'Next' buttons.

- Under action, check all S3.

The screenshot shows the 'Specify permissions' step with the 'Actions allowed' section expanded. A red arrow points to the 'All S3 actions (s3:*)' checkbox under the 'Manual actions | Add actions' heading. The 'Effect' dropdown is set to 'Allow'. The 'Visual' tab is selected in the top right corner. At the bottom right are 'Cancel' and 'Next' buttons.

- Under resources, check all.

Specify resource ARNs for these actions.

All
 Specific

accessgrant <small>Info</small>	⚠ Specified accessgrant resource ARN for the DeleteAccessGrant and 4 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
accessgrantsinstance <small>Info</small>	⚠ Specified accessgrantsinstance resource ARN for the AssociateAccessGrantsIdentityCenter and 16 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
accessgrantslocation <small>Info</small>	⚠ Specified accessgrantslocation resource ARN for the CreateAccessGrant and 6 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
accesspoint <small>Info</small>	⚠ Specified accesspoint resource ARN for the CreateAccessPoint and 9 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
bucket <small>Info</small>	⚠ Specified bucket resource ARN for the CreateBucket and 54 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any
job <small>Info</small>	⚠ Specified job resource ARN for the DeleteJobTagging and 5 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
multiregionaccesspoint <small>Info</small>	⚠ Specified multiregionaccesspoint resource ARN for the CreateMultiRegionAccessPoint and 7 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
multiregionaccesspointrequestarn <small>Info</small>	⚠ Specified multiregionaccesspointrequestarn resource ARN for the DescribeMultiRegionAccessPointOperation action. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
object <small>Info</small>	⚠ Specified object resource ARN for the AbortMultipartUpload and 32 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any
objectlambdaaccesspoint <small>Info</small>	⚠ Specified objectlambdaaccesspoint resource ARN for the CreateAccessPointForObjectLambda and 8 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
storagelensconfiguration <small>Info</small>	⚠ Specified storagelensconfiguration resource ARN for the DeleteStorageLensConfiguration and 5 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account
storagelensgroup <small>Info</small>	⚠ Specified storagelensgroup resource ARN for the DeleteStorageLensGroup and 5 more actions. Add ARNs to restrict access.	<input type="checkbox"/> Any in this account

Request conditions - optional
Actions on resources are allowed or denied only when these conditions are met.

+ Add more permissions

- Having checked all, click next.

Specify what actions can be performed on specific resources in S3.

Actions allowed

Specify actions from the service to be allowed.

Filter Actions

All S3 actions (S3*)

Access level

List (Selected 16/16)
 Read (Selected 61/61)
 Write (Selected 49/49)
 Permissions management (Selected 27/27)
 Tagging (Selected 12/12)

Dependent permissions not selected.
To grant permissions for the selected resource actions, including additional dependent actions might be required.

- s3:CreateBucketMetadataTableConfiguration requires 7 more actions.
- s3:CreateJob requires 1 more action.
- s3:PutReplicationConfiguration requires 1 more action.
- s3:UpdateBucketMetadataInventoryTableConfiguration requires 7 more actions.

Resources

Specify resource ARNs for these actions.

All
 Specific

⚠ The all wildcard '*' may be overly permissive for the selected actions. Allowing specific ARNs for these service resources can improve security.

Request conditions - optional
Actions on resources are allowed or denied only when these conditions are met.

+ Add more permissions

Cancel **Next**

- Name policy and click create policy.

Review and create

Policy details

Policy name
Enter a meaningful name to identify this policy.

Description - optional
Add a short explanation for this policy.

Permissions defined in this policy

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it.

Service	Access level	Resource	Request condition
S3	Full access	All resources	None

Add tags - optional

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Create policy

- Analyst policy successfully created.

Policy analyst-policy created.

Policies (1389)

A policy is an object in AWS that defines permissions.

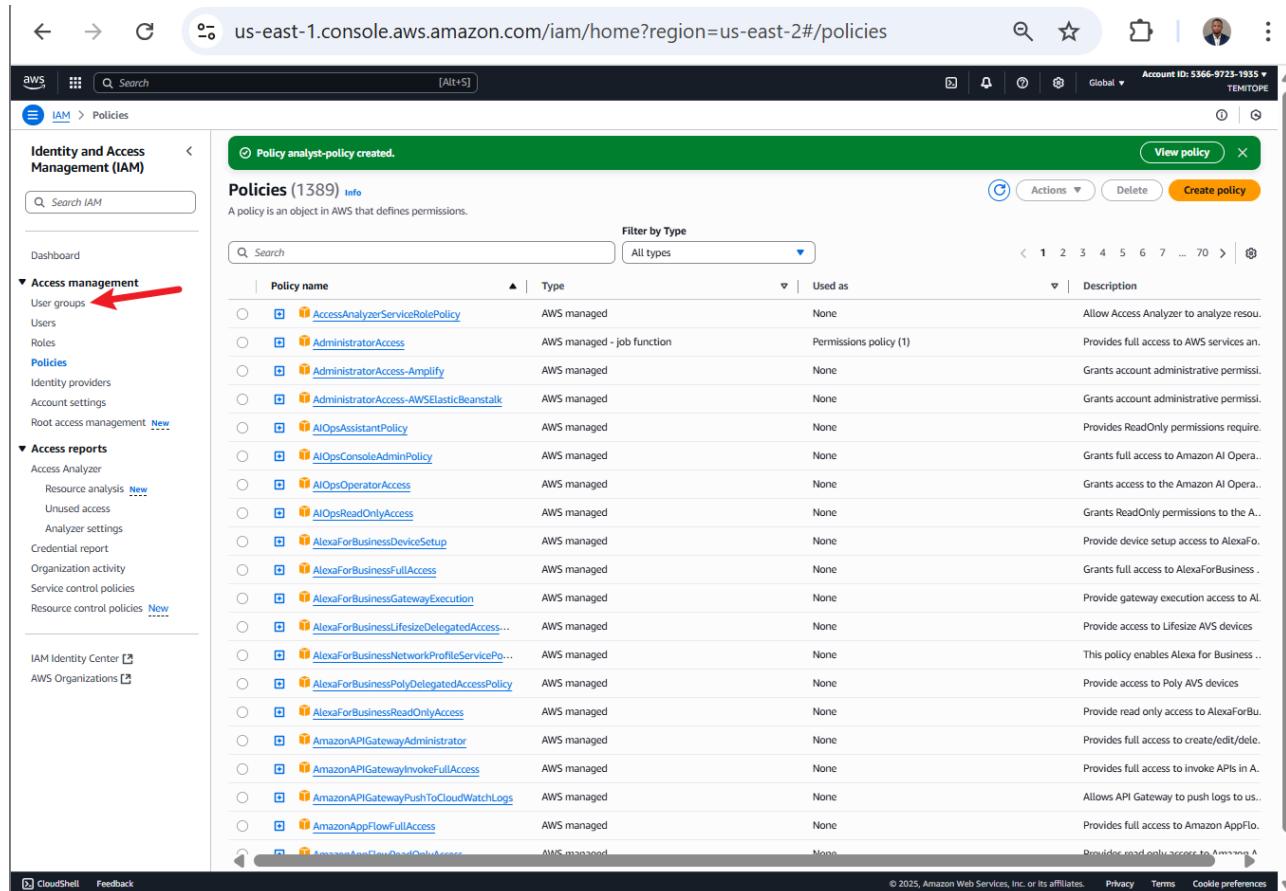
Policy name	Type	Used as	Description
AccessAnalyzerServiceRolePolicy	AWS managed	None	Allow Access Analyzer to analyze resou.
AdministratorAccess	AWS managed - job function	Permissions policy (1)	Provides full access to AWS services an.
AdministratorAccess-Amplify	AWS managed	None	Grants account administrative permis.
AdministratorAccess-AWSElasticBeanstalk	AWS managed	None	Grants account administrative permis.
AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly permissions require.
AIOpsConsolidatedAdminPolicy	AWS managed	None	Grants full access to Amazon AI Opera..
AIOpsOperatorAccess	AWS managed	None	Grants access to the Amazon AI Opera..
AIOpsReadOnlyAccess	AWS managed	None	Grants ReadOnly permissions to the A..
AlexaForBusinessDeviceSetup	AWS managed	None	Provide device setup access to AlexaFo.
AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness ..
AlexaForBusinessGatewayExecution	AWS managed	None	Provide gateway execution access to Al.
AlexaForBusinessLifesizeDelegatedAccess...	AWS managed	None	Provide access to Lifesize AVS devices
AlexaForBusinessNetworkProfileServicePo...	AWS managed	None	This policy enables Alexa for Business ..
AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None	Provide access to Poly AVS devices
AlexaForBusinessReadOnlyAccess	AWS managed	None	Provide read only access to AlexaForBu.
AmazonAPIGatewayAdministrator	AWS managed	None	Provides full access to create/edit/delete.
AmazonAPIGatewayInvokeFullAccess	AWS managed	None	Provides full access to invoke APIs in A..
AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	None	Allows API Gateway to push logs to us..
AmazonAppFlowFullAccess	AWS managed	None	Provides full access to Amazon AppFlo.
AmazonAppFlowReadOnlyAccess	AWS managed	None	Provides read only access to Amazon Ap..

5. Creating IAM Groups

- **Development-Team** → Attached [developer-policy](#).
- **Analyst-Team** → Attached [analyst-policy](#).

Steps for Development-Team:

- Click on user group.



The screenshot shows the AWS IAM Policies page at us-east-1.console.aws.amazon.com/iam/home?region=us-east-2#/policies. A red arrow points to the 'User groups' link in the left sidebar under 'Access management'. A green success message at the top says 'Policy analyst-policy created.' Below it, a table lists 1389 policies. The columns are 'Policy name', 'Type', 'Used as', and 'Description'. Some descriptions include: 'Allow Access Analyzer to analyze resou...', 'Provides full access to AWS services an...', 'Grants account administrative permissi...', 'Grants account administrative permissi...', 'Provides ReadOnly permissions require...', 'Allows full access to Amazon AI Opera...', 'Grants access to the Amazon AI Opera...', 'Grants ReadOnly permissions to the A...', 'Provide device setup access to AlexaFo...', 'Grants full access to AlexaForBusiness...', 'Provide gateway execution access to AL...', 'Provide access to Lifesize AVS devices...', 'This policy enables Alexa for Business ..', 'Provide access to Poly AVS devices...', 'Provide read only access to AlexaForBu...', 'Provides full access to create/edit/delete...', 'Provides full access to invoke APIs in A...', 'Allows API Gateway to push logs to us...', 'Provides full access to Amazon AppFlow...', and 'Provides read only access to Amazon A...'. The bottom of the page includes links for CloudShell, Feedback, and navigation icons.

- Click on create group.

The screenshot shows the AWS IAM User Groups page. On the left, there's a navigation sidebar with sections like 'Access management' (User groups, Users, Roles, Policies, Identity providers, Account settings, Root access management), 'Access reports' (Access Analyzer, Resource analysis, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies, Resource control policies), and 'IAM Identity Center'. The main area is titled 'User groups (0) Info' and contains a search bar and a table header with columns 'Group name', 'Users', and 'Permissions'. A message says 'No resources to display'. At the top right, there are 'Delete' and 'Create group' buttons, with the 'Create group' button highlighted by a red arrow.

- Name the group based on their role.

The screenshot shows the 'Create user group' wizard. The left sidebar is identical to the previous screenshot. The main area has three steps: 'Name the group' (with a 'User group name' input field containing 'Development-Team', highlighted with a red arrow), 'Add users to the group - Optional (1)', and 'Attach permissions policies - Optional (1074)'. The 'Add users to the group' step shows a table with one user named 'jiro'. The 'Attach permissions policies' step shows a table with two policies: 'AdministratorAccess' (AWS managed - job fun...) and 'AdministratorAccess' (AWS managed). Both policies have a blue checkmark icon next to them.

- Attach developer policy and click create group.

The screenshot shows the 'Create user group' interface in the AWS IAM console. On the left, the navigation menu is visible with 'User groups' selected. In the main area, there are two sections: 'Add users to the group - Optional (1)' and 'Attach permissions policies - Optional (1/1074)'. The 'Attach permissions policies' section contains a table with several AWS managed policies listed. One row for 'developer-policy' has a checked checkbox. At the bottom right of this section is a yellow 'Create user group' button.

- Development user group successfully created.

The screenshot shows the 'User groups' page in the AWS IAM console. A green success message at the top states 'Development-Team user group created.' Below it, a table lists the 'User groups (1)'. There is one entry: 'Development-Team', which is defined and has 0 users assigned. The 'Create group' button is visible at the top right of the table area.

Steps for Analyst-Team:

- Click on create group to create one more.

The screenshot shows the AWS IAM User Groups page. On the left, there's a navigation sidebar with sections like 'Identity and Access Management (IAM)', 'Access management', 'Access reports', and 'IAM Identity Center'. The main area displays a table titled 'User groups (1)'. The table has columns for 'Group name', 'Users', 'Permissions', and 'Creation time'. One row is shown for 'Development-Team'. At the top right of the main area, there are buttons for 'Delete' and 'Create group'. A red arrow points to the 'Create group' button.

- Name the group based on their role.

The screenshot shows the 'Create user group' wizard. The first step, 'Name the group', has a red arrow pointing to the 'User group name' input field, which contains 'Analyst-Team'. The second step, 'Add users to the group - Optional', shows a table with one user listed: 'jiro'. The third step, 'Attach permissions policies - Optional (1074)', shows a table with two policies listed: 'AdministratorAccess' and 'AdministratorAcc...'. The 'AdministratorAccess' policy is selected.

- Attach analyst policy and click create group.

The screenshot shows the 'Create user group' wizard in the AWS IAM console. In the 'Name the group' step, the user has entered 'Analyst-Team'. In the 'Add users to the group - Optional' step, one user 'jiro' is selected. In the 'Attach permissions policies - Optional' step, a single policy named 'analyst-policy' is selected. This policy is described as 'Customer managed' and 'None' for 'Used as'. At the bottom right of this step, there is a prominent orange 'Create user group' button.

(Note: Filename references developer policy, but context indicates analyst policy attachment.)

- Analyst user group successfully created.

The screenshot shows the 'User groups' page in the AWS IAM console. It displays two groups: 'Analyst-Team' and 'Development-Team'. Both groups have 0 users and are defined. The 'Analyst-Team' group was created 'Now'. At the top right of the page, there is a yellow 'Create group' button.

6. Creating IAM Users

John (Backend Developer)

- User created: **John**.
- Added to **Development-Team** group.
- Console login enabled.
- Credentials downloaded.

Steps:

- Click users to create users.

The screenshot shows the AWS IAM User Groups page. On the left, there's a sidebar with 'Identity and Access Management (IAM)' and several sections like 'Access management', 'Access reports', and 'AWS Identity Center'. A red arrow points to the 'Users' link under 'Access management'. The main area displays 'User groups (2) Info' with a table showing two entries:

Group name	Users	Permissions	Creation time
Analyst-Team	△ 0	Defined	14 hours ago
Development-Team	△ 0	Defined	15 hours ago

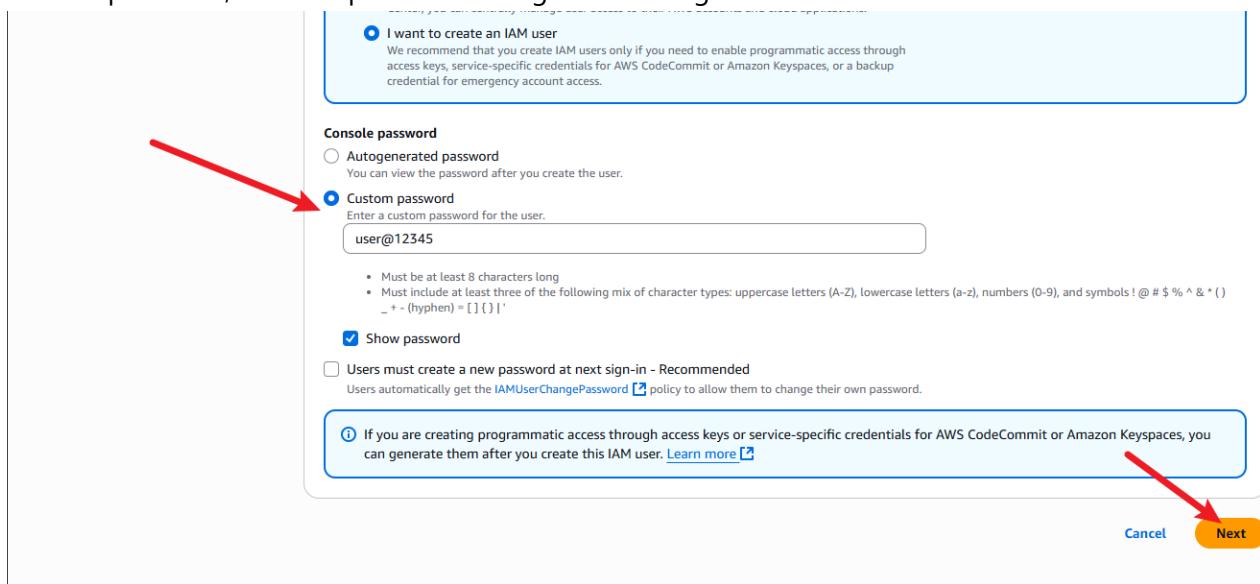
- Click create user.

The screenshot shows the AWS IAM console with the URL us-east-1.console.aws.amazon.com/iam/home?region=us-east-2#/users. The left sidebar is collapsed. The main area displays a table titled 'Users (1) Info' with one entry: 'jira' (User name), Path '/', Group: 0, Last activity 19 days ago. The top right corner shows the account ID '5366-9723-1935' and a user icon. A red arrow points to the 'Create user' button in the top right of the main content area.

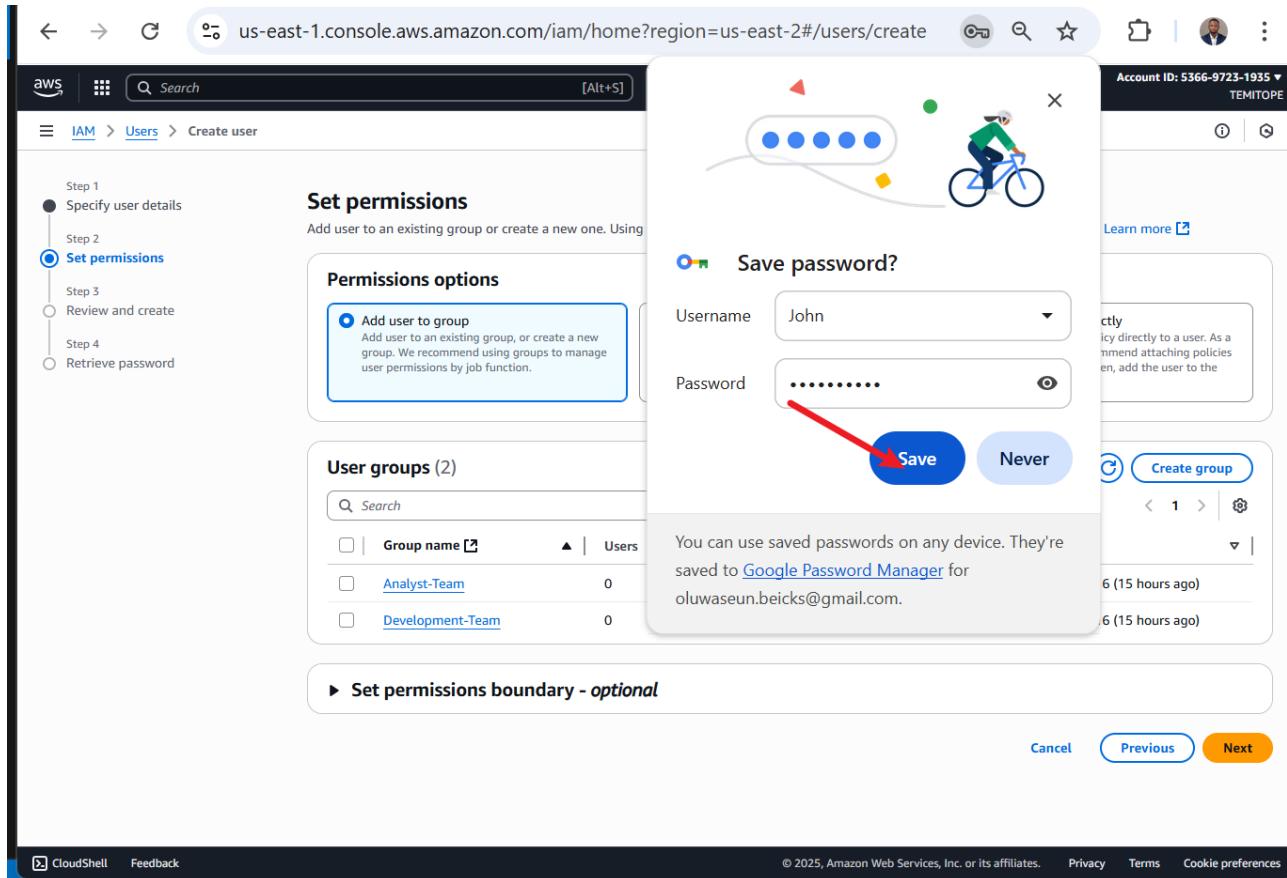
- Enter name, provide access to console, and check IAM user.

The screenshot shows the 'Specify user details' step of the IAM user creation wizard. The URL is us-east-1.console.aws.amazon.com/iam/home?region=us-east-2#/users/create. On the left, a vertical navigation bar shows 'Step 1: Specify user details' (selected), 'Step 2: Set permissions', 'Step 3: Review and create', and 'Step 4: Retrieve password'. The main area is titled 'Specify user details' and contains a 'User details' section with a 'User name' input field containing 'John'. Below it is a note: 'The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . @ _ - (hyphen)'. A checkbox 'Provide user access to the AWS Management Console - optional' is checked. A callout box highlights the 'Are you providing console access to a person?' section, which includes 'User type' radio buttons: 'Specify a user in Identity Center - Recommended' (unchecked) and 'I want to create an IAM user' (checked). A note for the latter says: 'We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.' At the bottom, there's a 'Console password' section with 'Autogenerated password' (checked) and 'Custom password' (unchecked).

- Choose password, uncheck password change on next login.



- Save password for next login.



- Check to add to development group and click next.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

- Add user to group**
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- Copy permissions**
Copy all group memberships, attached managed policies, and inline policies from an existing user.
- Attach policies directly**
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (1/2)

Group name	Users	Attached policies	Created
<input type="checkbox"/> Analyst-Team	0	analyst-policy	2025-08-16 (15 hours ago)
<input checked="" type="checkbox"/> Development-Team	0	developer-policy	2025-08-16 (15 hours ago)

Set permissions boundary - optional

Cancel Previous Next

- Review and create user.

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name John	Console password type Custom password	Require password reset No
-------------------	--	------------------------------

Permissions summary

Name	Type	Used as
Development-Team	Group	Permissions group

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous Create user

- Successfully create user.

User created successfully
You can view and download the user's password and email instructions for signing in to the AWS Management Console.

Step 1 Specify user details
Step 2 Set permissions
Step 3 Review and create
Step 4 **Retrieve password**

Retrieve password
You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

Console sign-in URL: <https://536697231935.signin.aws.amazon.com/console>

User name: John

Console password: ***** [Show](#)

[Email sign-in instructions](#)

Cancel [Download .csv file](#) [Return to users list](#)

- Download user login CSV.

User created successfully
You can view and download the user's password and email instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Step 1 Specify user details
Step 2 Set permissions
Step 3 Review and create
Step 4 **Retrieve password**

Retrieve password
You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

Console sign-in URL: <https://536697231935.signin.aws.amazon.com/console>

User name: John

Console password: ***** [Show](#)

[Email sign-in instructions](#)

Cancel [Download .csv file](#) [Return to users list](#)

Mary (Data Analyst)

- User created: **Mary**.
- Added to **Analyst-Team** group.

- Console login enabled.
- Credentials downloaded.

Steps:

- Click create new user to create new user Mary.

The screenshot shows the AWS IAM service in the AWS Management Console. The left sidebar navigation includes 'Identity and Access Management (IAM)', 'Access management' (with 'Users' selected), and 'Access reports'. The main content area displays a table titled 'Users (2)'. The table has columns for User name, Path, Group, Last activity, MFA, Password age, and Consol. Two users are listed: 'jiro' and 'John'. The 'Last activity' column shows '19 days ago' for 'jiro' and '-' for 'John'. In the top right corner of the main area, there is a 'Create user' button, which is highlighted with a red arrow.

- Enter name, provide access to console, and check IAM user.

The screenshot shows the 'Specify user details' step of the 'Create user' wizard. On the left, a sidebar lists steps: Step 1 (selected), Step 2, Step 3, Step 4, and Retrieve password. The main area is titled 'Specify user details' and contains a 'User details' section. It includes a 'User name' input field with the value 'Mary', a note about valid characters, and a checked checkbox for 'Provide user access to the AWS Management Console - optional'. Below this is a question 'Are you providing console access to a person?' with two options: 'Specify a user in Identity Center - Recommended' (unchecked) and 'I want to create an IAM user' (checked). A note for the second option recommends creating IAM users for programmatic access. The 'Console password' section follows, with 'Autogenerated password' selected. A note states that the password must be at least 8 characters long and include uppercase letters, lowercase letters, numbers, and symbols. At the bottom, there are 'Next Step' and 'Cancel' buttons.

- Choose password, uncheck password change on next login.

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . @ - (hyphen)

Provide user access to the AWS Management Console - *optional*
If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

① Are you providing console access to a person?

User type

Specify a user in Identity Center - **Recommended**
We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.

I want to create an IAM user
We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.

Console password

Autogenerated password
You can view the password after you create the user.

Custom password
Enter a custom password for the user.
User4567

Show password

Users must create a new password at next sign-in - **Recommended**
Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

② If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. [Learn more](#)

[Cancel](#) **Next**

- Save password for next login.

User "Mary" deleted.

Step 1 Specify user details
Step 2 Set permissions
Step 3 Review and create
Step 4 Retrieve password

Set permissions
Add user to an existing group or create a new one

Permissions options

Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

User groups (2)

Group name	Policy	Created
Analyst-Team	analyst-policy	2025-08-16 (16 hours ago)
Development-Team	developer-policy	2025-08-16 (16 hours ago)

Save password?

Username: Mary

Password: [Show](#)

Save **Never**

You can use saved passwords on any device. They're saved to [Google Password Manager](#) for oluwaseun.beicks@gmail.com.

Set permissions boundary - optional

[Cancel](#) [Previous](#) **Next**

- Check to add to analyst group and click next.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

- Add user to group Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- Copy permissions Copy all group memberships, attached managed policies, and inline policies from an existing user.
- Attach policies directly Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (1/2)

Group name	Users	Attached policies	Created
<input checked="" type="checkbox"/> Analyst-Team	0	analyst-policy	2025-08-16 (16 hours ago)
<input type="checkbox"/> Development-Team	0	developer-policy	2025-08-16 (16 hours ago)

► Set permissions boundary - optional

Cancel [Previous](#) **Next**

- Review and create user.

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name	Console password type	Require password reset
Mary	Custom password	No

Permissions summary

Name	Type	Used as
Analyst-Team	Group	Permissions group

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

Cancel [Previous](#) **Create user**

- Successfully create user.

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

Retrieve password

You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

Console sign-in URL
https://536697231935.signin.aws.amazon.com/console

User name
Mary

Console password
***** [Show](#)

[Email sign-in instructions](#)

Cancel [Download .csv file](#) [Return to users list](#)

- Download user login CSV.

Console sign-in details

Console sign-in URL
https://536697231935.signin.aws.amazon.com/console

User name
Mary

Console password
***** [Show](#)

[Email sign-in instructions](#)

Cancel [Download .csv file](#) [Return to users list](#)

7. Testing and Validation

- Return user list.

Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password

Retrieve password
You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

Console sign-in URL
<https://536697231935.signin.aws.amazon.com/console>

User name
[Mary](#)

Console password
[***** Show](#)

[Email sign-in instructions](#)

[Cancel](#) [Download .csv file](#) [Return to users list](#)

- Landed on user list page.

Identity and Access Management (IAM)

[Search IAM](#)

Access management

- User groups
- Users**
- Roles
- Policies
- Identity providers
- Account settings
- Root access management [New](#)

Access reports

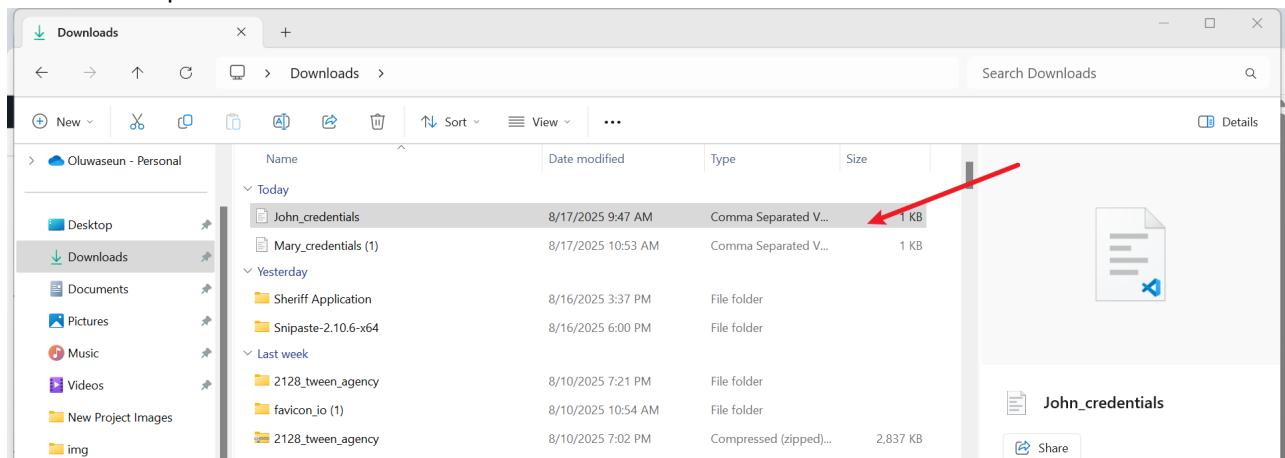
User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active key age	Access
jiro	/	0	19 days ago	-	-	-	Inactive - AKIAZ5NF3...	-	Edit Delete
John	/	1	-	-	1 hour	-	-	-	Edit Delete
Mary	/	1	-	-	1 minute	-	-	-	Edit Delete

John's Access

- Logged in as John.
- Accessed **EC2 Dashboard** .
- Attempted to access **S3 Dashboard** (denied).

Steps:

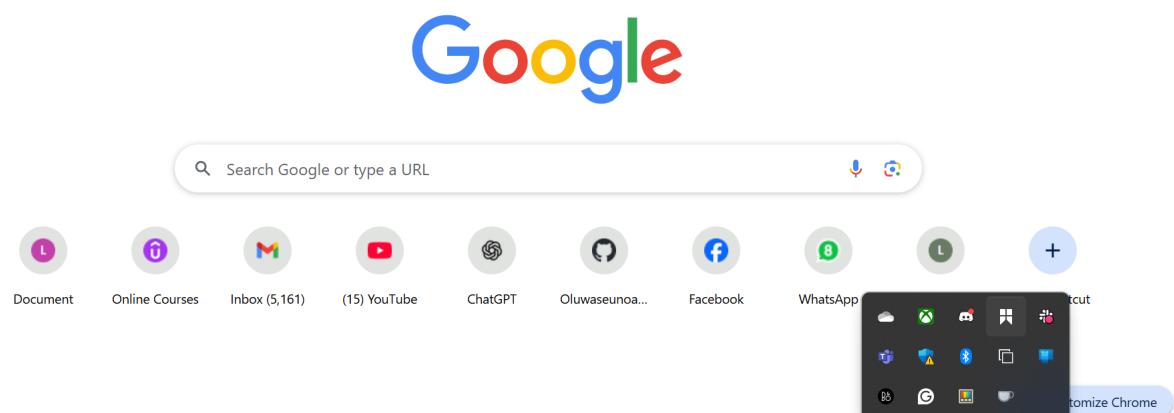
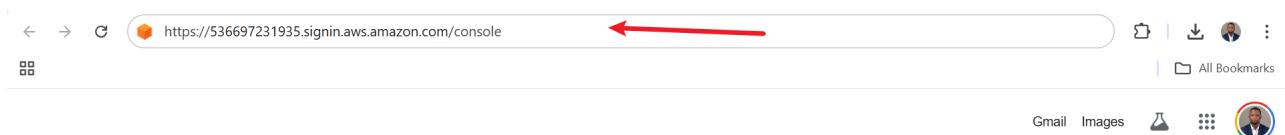
- Locate and open John downloaded CSV file.



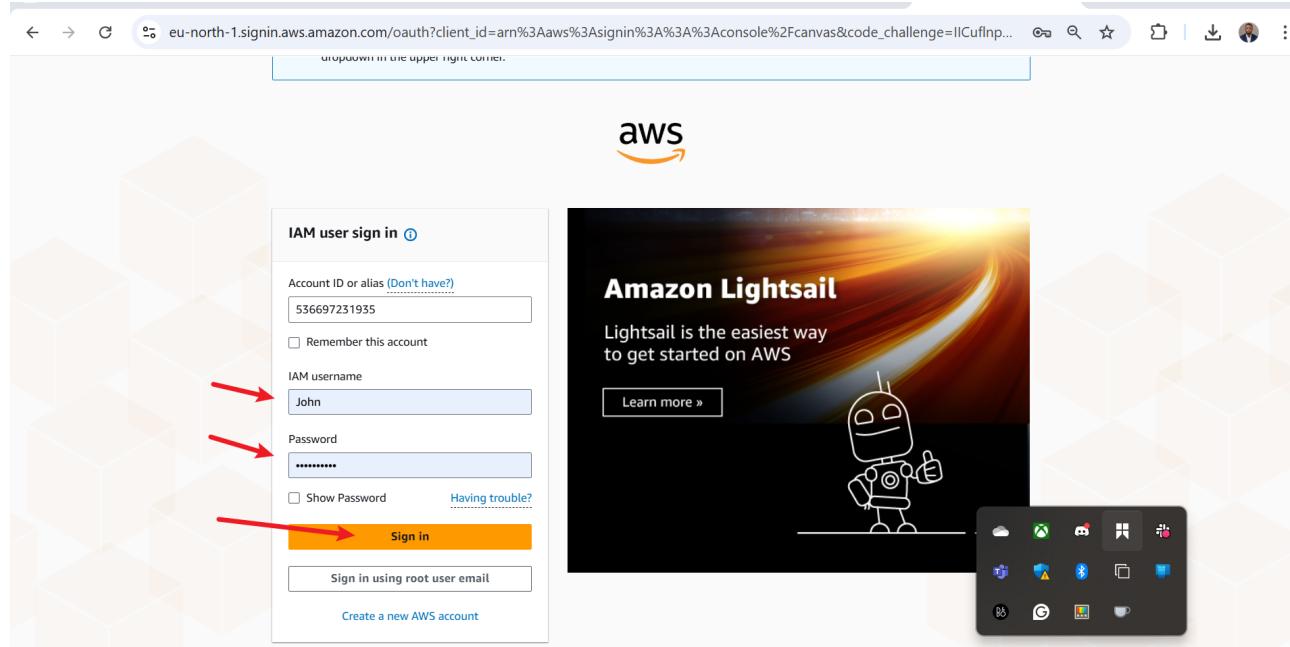
- Copy the HTTPS link.



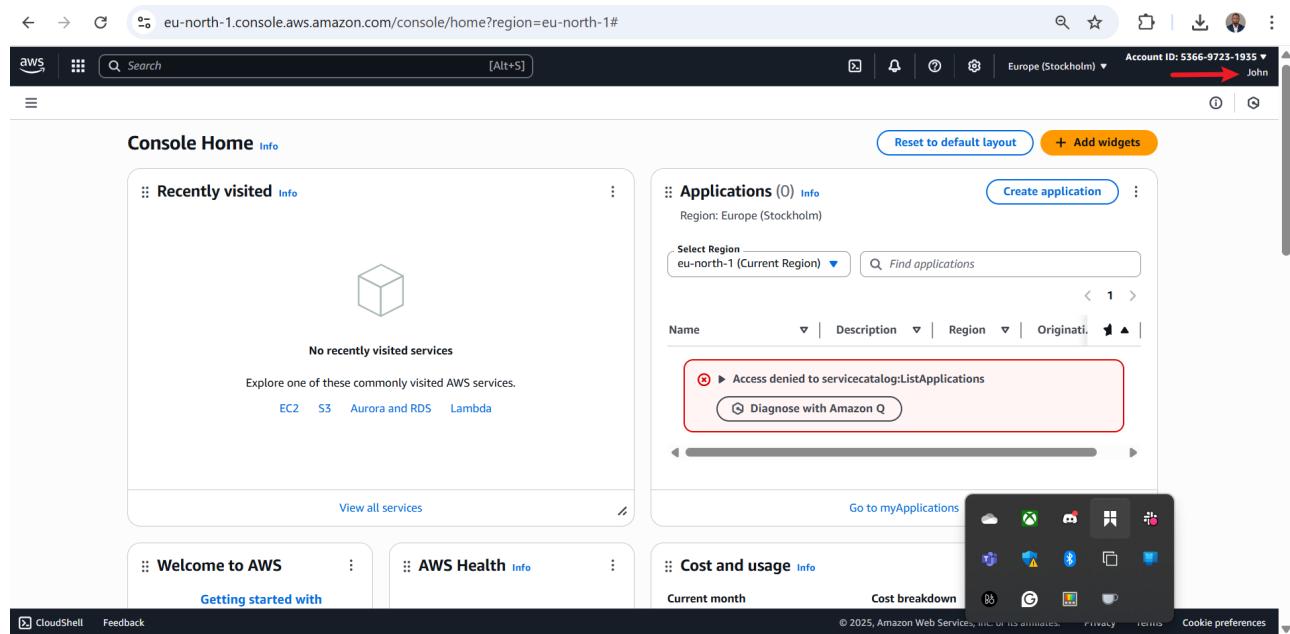
- Paste it on the browser.



- Sign in as IAM user using your login and account ID which is fetched by the link.



- Logged in as John.



- Search for EC2 and click on it.

The screenshot shows the AWS search results for 'ec2'. The 'Services' section is expanded, displaying several items. The first item, 'EC2 Virtual Servers in the Cloud', is highlighted with a red arrow. Other visible items include 'EC2 Image Builder' and 'EC2 Global View'. Below the services, the 'Features' section is shown, containing 'EC2 Instances', 'EC2 Resource Health', and 'Dashboard'. At the bottom left, there's a feedback section with 'Yes' and 'No' buttons. The top right corner shows the account ID '5366-9723-1935' and a user profile for 'John'. The bottom right corner includes links for 'CloudShell', 'Feedback', 'Cost breakdown', and copyright information.

- Successfully access EC2 dashboard, click launch instance.

The screenshot shows the EC2 home page. On the left, a sidebar menu lists 'EC2' (selected), 'Dashboard', 'EC2 Global View', 'Events', 'Instances' (selected), 'Images', 'Elastic Block Store', and 'Lifecycle Manager'. The main content area features the heading 'Amazon Elastic Compute Cloud (EC2)' and the sub-headline 'Create, manage, and monitor virtual servers in the cloud.' It also includes a paragraph about the service's offerings and a 'Benefits and features' section. On the right, there's a 'Launch a virtual server' section with 'Launch instance', 'View dashboard', and 'Get started walkthroughs' buttons. Below that is an 'Additional actions' section with 'View running instances' and 'Migrate a server' buttons. The bottom right corner includes links for 'CloudShell', 'Feedback', 'Cost breakdown', and copyright information.

- Launch an instance page appear.

The screenshot shows the AWS EC2 'Launch an instance' page. It includes fields for 'Name and tags' (with 'e.g. My Web Server' entered), 'Application and OS Images (Amazon Machine Image)' (with a search bar and a list of operating systems like Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian), and a 'Summary' section on the right showing 'Number of instances' (1) and a 'Launch' button. A message at the top says 'It seems like you may be new to launching instances in EC2. Take a walkthrough to learn about EC2, how to launch instances and about best practices'.

- Name server and set it up, also set key pairs and launch instance.

The screenshot shows the AWS EC2 'Launch an instance' page with the 'Ubuntu 24.04 LTS (HVM), SSD Volume Type' AMI selected. The 'Quick Start' section shows the AMI details: Canonical, Ubuntu, 24.04, amd64 noble image. The 'Summary' section on the right shows 'Number of instances' (1) and a 'Launch instance' button. A message at the top says 'It seems like you may be new to launching instances in EC2. Take a walkthrough to learn about EC2, how to launch instances and about best practices'.

- Instance successfully launched.

The screenshot shows the AWS EC2 Instances launch success page. At the top, a green banner displays the message "Successfully initiated launch of instance (i-0b61d2ccfb672cbb)". Below this, a "Launch log" button is visible. The main area is titled "Next Steps" and contains ten cards:

- Create billing usage alerts**: To manage costs and avoid surprise bills, set up email notifications for billing usage thresholds. Includes a "Create billing alerts" button.
- Connect to your instance**: Once your instance is running, log into it from your local computer. Includes a "Connect to instance" button.
- Connect an RDS database**: Configure the connection between an EC2 instance and a database to allow traffic flow between them. Includes a "Connect an RDS database" button.
- Create EBS snapshot policy**: Create a policy that automates the creation, retention, and deletion of EBS snapshots. Includes a "Create EBS snapshot policy" button.
- Manage detailed monitoring**: Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period. Includes a "Manage detailed monitoring" button.
- Create Load Balancer**: Create a application, network gateway or classic Elastic Load Balancer. Includes a "Create Load Balancer" button.
- Create AWS budget**: AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location. Includes a "Create AWS budget" button.
- Manage CloudWatch alarms**: Create or update Amazon CloudWatch alarms for the instance. Includes a "Manage CloudWatch alarms" button.
- Disaster recovery for your instances**: Recover the instances you just launched into a different Availability Zone or a different Region using AWS Elastic Disaster Recovery (DRS). Includes a "Disaster recovery for your instances" button.
- Monitor for suspicious runtime activities**: Amazon GuardDuty enables you to continuously monitor for malicious runtime activity and unauthorized behavior, with near real-time visibility into on-host activities occurring across your Amazon EC2 workloads. Includes a "Monitor for suspicious runtime activities" button.
- Get instance screenshot**: Capture a screenshot from the instance and view it as an image. This is useful for troubleshooting an unreachable instance. Includes a "Get instance screenshot" button.
- Get system log**: View the instance's system log to troubleshoot issues. Includes a "Get system log" button.

At the bottom right, there is a small icon bar with various AWS services: CloudWatch Metrics, CloudWatch Logs, CloudWatch Metrics Insights, CloudWatch Metrics Data Firehose, CloudWatch Metrics Data Pipeline, CloudWatch Metrics Data Processor, CloudWatch Metrics Data Transformer, CloudWatch Metrics Data Analytics, CloudWatch Metrics Data Exchange, CloudWatch Metrics Data Pipeline, CloudWatch Metrics Data Processor, CloudWatch Metrics Data Transformer, CloudWatch Metrics Data Analytics, CloudWatch Metrics Data Exchange, CloudWatch Metrics Data Pipeline, CloudWatch Metrics Data Processor, CloudWatch Metrics Data Transformer, CloudWatch Metrics Data Analytics, CloudWatch Metrics Data Exchange.

- Search for S3 and click on it.

The screenshot shows the AWS search results for "s3". A red arrow points to the search bar at the top left. Another red arrow points to the "S3 Scalable Storage in the Cloud" service card in the "Services" section. The search results include:

- Services**
 - S3 Scalable Storage in the Cloud
 - S3 Glacier Archive Storage in the Cloud
 - AWS Snow Family Large Scale Data Transport
- Features**
 - S3 on Outposts AWS Outposts feature
 - Exports to S3 DynamoDB feature
 - S3 Access Grants S3 feature
- Resources** / for a focused search
 - Introducing resource search To search for resources, Resource Explorer must be active in at least one AWS Region and you must have permission to use the default view in the account. [Learn more](#)

At the bottom right, there is a small icon bar with various AWS services: CloudWatch Metrics, CloudWatch Logs, CloudWatch Metrics Insights, CloudWatch Metrics Data Firehose, CloudWatch Metrics Data Pipeline, CloudWatch Metrics Data Processor, CloudWatch Metrics Data Transformer, CloudWatch Metrics Data Analytics, CloudWatch Metrics Data Exchange, CloudWatch Metrics Data Pipeline, CloudWatch Metrics Data Processor, CloudWatch Metrics Data Transformer, CloudWatch Metrics Data Analytics, CloudWatch Metrics Data Exchange, CloudWatch Metrics Data Pipeline, CloudWatch Metrics Data Processor, CloudWatch Metrics Data Transformer, CloudWatch Metrics Data Analytics, CloudWatch Metrics Data Exchange.

- Successfully access S3 dashboard, click create bucket.

The screenshot shows the Amazon S3 dashboard. On the right side, there is a prominent 'Create a bucket' button. The main content area features a section titled 'How it works' with a video thumbnail showing a bucket icon in the clouds.

(Note: Context indicates access attempt leading to denial.)

- Name bucket and set it up.

The screenshot shows the 'Create bucket' configuration page. Under 'General configuration', the 'Bucket name' field is filled with 'johns3bucket'. Under 'Bucket type', the 'General purpose' option is selected. In the 'Object Ownership' section, 'ACLs disabled (recommended)' is selected. At the bottom, the 'Bucket owner enforced' option is chosen.

- Scroll down and click create bucket.

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

[Add new tag](#)

You can add up to 50 tags.

Default encryption [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)
 Server-side encryption with AWS Key Management Service keys (SSE-KMS)
 Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable
 Enable

Advanced settings

ⓘ After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

[Cancel](#) [Create bucket](#)

- Failed to create bucket.

eu-north-1.console.aws.amazon.com/s3/bucket/create?region=eu-north-1&bucketType=ge...

Default encryption [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)
 Server-side encryption with AWS Key Management Service keys (SSE-KMS)
 Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable
 Enable

Advanced settings

ⓘ After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Failed to create bucket

To create a bucket, the s3:CreateBucket permission is required.

View your permissions in the [IAM console](#). [Identity and Access Management in Amazon S3](#)

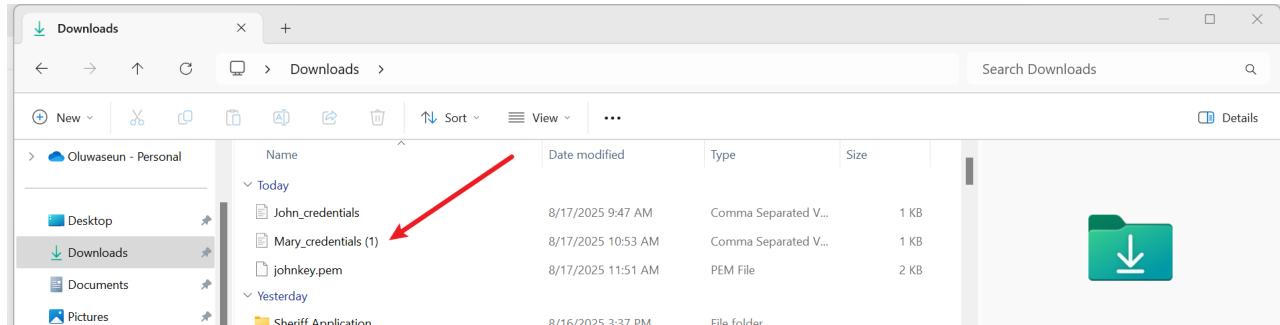
API response

[Cancel](#) [Create bucket](#)

- Logged in as Mary.
- Accessed **S3 Dashboard** .
- Attempted to access **EC2 Dashboard** (denied).

Steps:

- Locate and open Mary downloaded CSV file.

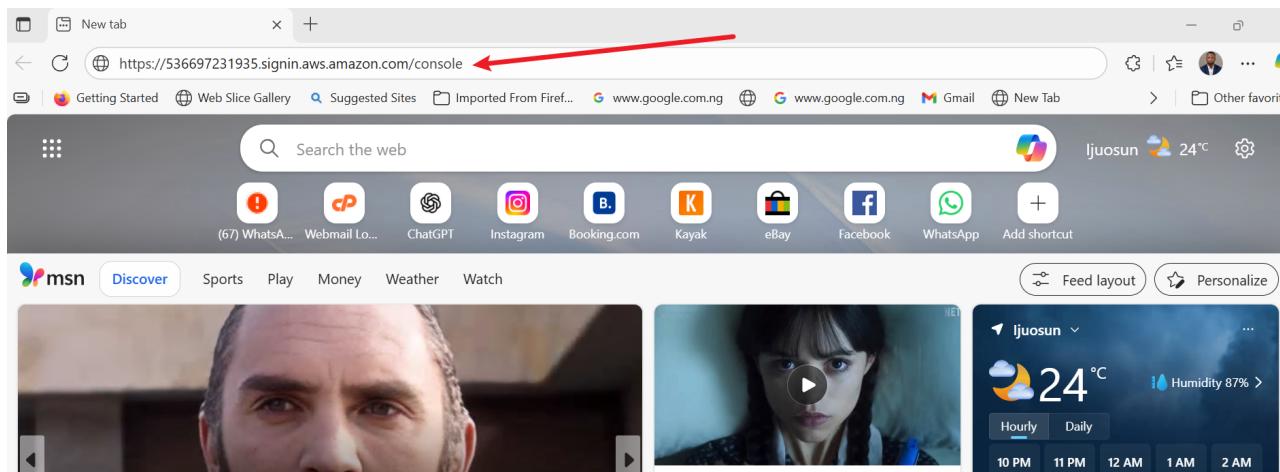


(Note: Filename references John, but context indicates Mary's file.)

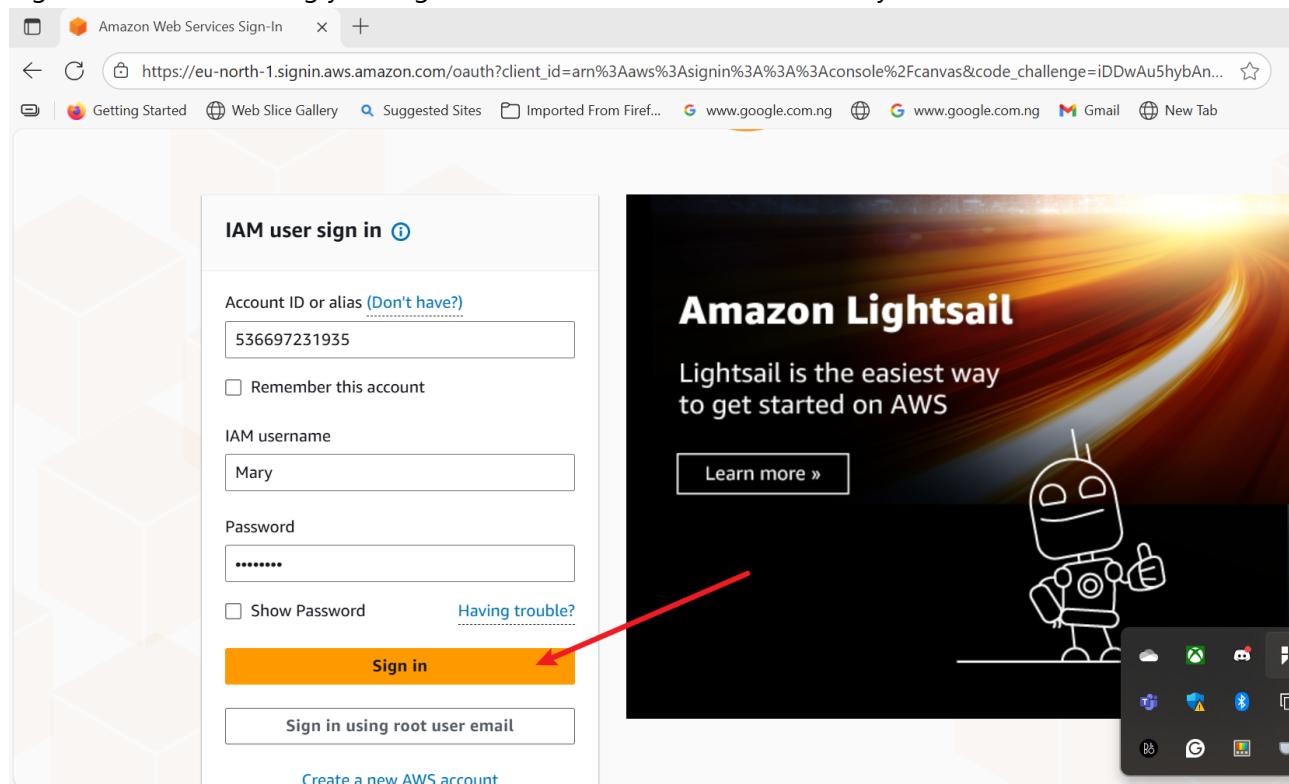
- Copy the HTTPS link.



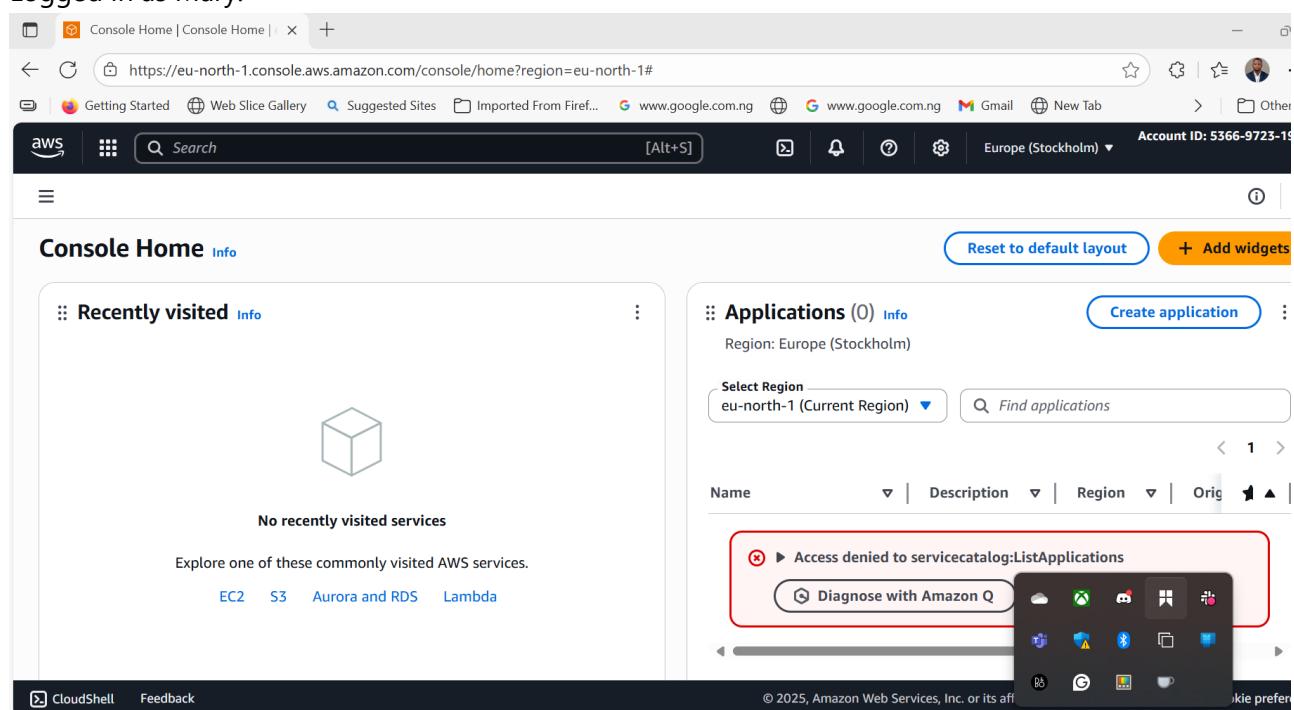
- Paste it on the browser.



- Sign in as IAM user using your login and account ID which is fetched by the link.



- Logged in as Mary.



- Search for S3 and click on it.

The screenshot shows the AWS console search results for 'S3'. The search bar at the top contains 's3'. Below the search bar, there are tabs for 'Services', 'Features', 'Resources', 'Documentation', 'Knowledge articles', and 'Marketplace'. The 'Services' tab is selected. The results list three services: 'S3 Scalable Storage in the Cloud', 'S3 Glacier Archive Storage in the Cloud', and 'AWS Snow Family Large Scale Data Transport'. Each service entry has a star icon and an upward arrow icon. At the bottom of the results page, there are buttons for 'Were these results helpful?' with 'Yes' and 'No' options, and links for 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences'. A copyright notice at the bottom reads '© 2025, Amazon Web Services, Inc. or its affiliates.'

- Successfully access S3 dashboard, click create bucket.

The screenshot shows the 'Amazon S3' get-started page. The main heading is 'Amazon S3' with the subtext 'Store and retrieve any amount of data from anywhere'. Below this, a paragraph states: 'Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.' To the right, a callout box titled 'Create a bucket' contains the text: 'Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.' A red arrow points to the 'Create bucket' button, which is highlighted with an orange background. At the bottom of the page, there are links for 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences', along with a copyright notice: '© 2025, Amazon Web Services, Inc. or its affiliates.'

- Name bucket and set it up.

The screenshot shows the 'Create bucket' page in the AWS Management Console. The 'General configuration' section is visible, featuring fields for 'Bucket name' (set to 'marys3bucket1234') and 'Bucket type'. A 'General purpose' bucket is selected, with a note explaining it's recommended for most use cases. A 'Directory' option is also shown with its own description. Other tabs like 'Copy settings from existing bucket - optional' and 'Object Ownership' are present but not active.

- Scroll down and click create bucket.

The screenshot shows the 'Create bucket' page with the 'Advanced settings' section expanded. A note at the top says 'After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.' A red arrow points to the 'Create bucket' button at the bottom right of the page.

- Bucket successfully created.

The screenshot shows the AWS S3 console interface. At the top, there's a success message: "Successfully created bucket 'marys3bucket1234'. To upload files and folders, or to configure additional bucket settings, choose View details." Below this, there are two tabs: "General purpose buckets" (selected) and "All AWS Regions". On the left, under "General purpose buckets (1)", there's a table with one row: Name (marys3bucket1234), AWS Region (Europe (Stockholm) eu-north-1), and Creation date (August 17, 2025, 21:32:49 (UTC+01:00)). There are buttons for "Copy ARN", "Empty", "Delete", and "Create bucket". To the right, there are sections for "Account snapshot" and "External access summary - new". The bottom of the page includes standard AWS footer links like CloudShell, Feedback, and Copyright information.

- Search for EC2 and click on it.

The screenshot shows the AWS search results for the query "ec2". A red arrow points to the search bar at the top. Another red arrow points to the "EC2" service card in the "Services" section. The search results include:

- EC2**: Virtual Servers in the Cloud
- EC2 Image Builder**: A managed service to automate build, customize and deploy OS images
- EC2 Global View**: EC2 Global View provides a global dashboard and search functionality that lets you ...
- EC2 Instances**: CloudWatch feature
- EC2 Resource Health**: CloudWatch feature
- Dashboard**: EC2 feature

 The bottom of the page includes standard AWS footer links like CloudShell, Feedback, and Copyright information.

- Successfully access EC2 dashboard, click launch instance.

The screenshot shows the EC2 home page with a sidebar containing navigation links like Dashboard, EC2 Global View, Events, Instances, Images, and Elastic Block Store. The main content area features the title "Amazon Elastic Compute Cloud (EC2)" and a sub-section titled "Benefits and features" with a box about scalability. On the right, there's a callout box with four buttons: "Launch a virtual server" (highlighted with a red arrow), "View dashboard", "Get started walkthroughs", and "Get started tutorial". Below this is another callout box for "Additional actions" with "View running instances" and "Migrate a server". The bottom right of the main content area has a "© 2025, Amazon Web Services, Inc. or its affiliates." footer.

(Note: Context indicates access attempt leading to denial.)

- Name server and set it up, also set key pairs and launch instance.

The screenshot shows the "Launch an instance" wizard. The first step, "Name and tags", has a text input field with "maryec2server1234" and a "Add additional tags" button. The second step, "Application and OS Images (Amazon Machine Image)", includes a search bar and a "Quick Start" section with buttons for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. The third step, "Summary", shows a summary of the configuration: 1 instance, Software Image (AMI) ami-0b83c7f5e2823d1f4, Virtual server type (instance type) t3.micro, Firewall (security group) New security group, and Storage (volumes). It includes "Cancel" and "Launch instance" buttons, with the latter highlighted by a red arrow. The bottom right of the summary section has a "Preview code" link.

- Mary not authorized to create key pair.

The screenshot shows the AWS EC2 Instances 'Launch an instance' page. A modal window titled 'Create key pair' is open. In the 'Key pair name' field, the value 'marykeypair' is entered and highlighted with a red box. Below the field, an error message is displayed: 'You are not authorized to perform this operation. User: arnaws:iam::536697231935:user/Mary is not authorized to perform: ec2:CreateKeyPair on resource: arn:aws:ec2:eu-north-1:536697231935:key-pair/marykeypair because no identity-based policy allows the ec2:CreateKeyPair action. Encoded authorization failure message: YN0VrcplcRumBeP3OQADGVfhlJ-nBdMycDRvQlwrmRK4MRGVKKii8-ytc4eRVBirk1CIUWVLWLj9176cErn0GLhEfYaP4HUB505FaCn28i2cKBbgYrjqboIWQ1-YnqFFxU976nXLArNc37ZgrSpd_JAMwshHgth5Ae8vA9Yt8svV2GKz7JX0896kmn1nTGNcxeKoQXCsk36Br2hN7x6aHmB4IDsfzrvJ3g5k49gSyetTr0656FUB65P5mkKGWn9L_d7dvVXupohxYKV5XM4BE_p6r0o-lbpvdEhbqx1nV1MBccsze35kcE6APMT7LCV45zlpUvy83GG9TonOjC4PdpkjOP72Gj8BLd55L0KRE0mNMMI9jBeScKbfdfHFcqPBsogk3UDm8G2rPJhWWVrfHdtCHRZlw-laj-QKatolPhMjcuz289Ru-dFTYHxlObiWusnUJZDriDCGSw_wa_bMReD6mmhWlfnaVSpXO_1ho'Eet5KFw2HyXjnRSvWJruOY3rCcAoMvBP_3a-d0dMAdUEc5IovPVF7stTz1nBfH3uV'.

This confirmed that users had **role-specific permissions** aligned with the principle of least privilege.

8. Enabling Multi-Factor Authentication (MFA)

John (Backend Developer)

- Enabled MFA.
- Linked with **Google Authenticator**.
- Tested successful login with MFA.

Steps:

- Search IAM to set MFA for John.

The screenshot shows the AWS S3 console with the 'Create bucket' wizard open. On the left, there's a sidebar with options like 'General', 'AWS Regions', 'Bucket type', 'Bucket name', 'Copy settings', 'Object controls', and 'ACLs'. The 'ACLs' section is selected. At the bottom of this sidebar is a 'Were these results helpful?' poll with 'Yes' and 'No' buttons. In the main content area, there's a 'Services' section with 'IAM' highlighted by a red arrow. Below it are sections for 'Features' (IAM Access analyzer for S3, Groups, Roles) and 'Resources' (Introducing resource search). The top right corner shows account information: Account ID: 5366-9723-1935, Europe (Stockholm), and a user profile for John.

- Login as IAM admin and navigate to IAM dashboard.

The screenshot shows the AWS IAM dashboard. The left sidebar includes 'Identity and Access Management (IAM)', 'Access management' (User groups, Users, Roles, Policies, Identity providers, Account settings, Root access management), 'Access reports' (Access Analyzer, Resource analysis, Unused access, Analyzer settings, Credential report, Organization activity, Service control policies, Resource control policies), and 'IAM Identity Center'. The main dashboard features a summary of IAM resources: User groups (2), Users (3), Roles (6), Policies (10), and Identity providers (0). Below this is a 'What's new' section with a list of recent changes, including the introduction of API keys for Amazon Bedrock, support for annotations in AWS Service Reference Information, expanded resource control policies (RCPs) for two additional services, and the enforcement of MFA for root users across all account types. To the right, there are boxes for 'AWS Account' (Account ID: 536697231935, Sign-in URL: https://536697231935.signin.aws.amazon.com/console), 'Quick Links' (My security credentials, Manage access keys, MFA, other credentials), and 'Tools' (Policy simulator, which evaluates policies for actions specified).

- Click on users.

The screenshot shows the AWS IAM Dashboard. On the left sidebar, under 'Access management', the 'Users' link is highlighted with a red arrow. The main content area displays 'IAM resources' with counts: 2 User groups, 3 Users, 6 Roles, 10 Policies, and 0 Identity providers. Below this is a 'What's new' section listing recent changes in IAM. To the right are three boxes: 'AWS Account' (Account ID: 536697231935), 'Quick Links' (My security credentials, Manage access keys), and 'Tools' (Policy simulator).

- Click on John.

The screenshot shows the 'Users' page in the AWS IAM console. The 'Users' link in the sidebar is highlighted with a red arrow. The main table lists three users: jiro, John, and Mary. The 'John' row is highlighted with a red arrow. The table columns include User name, Path, Group, Last activity, MFA, Password age, and Console last sign.

User name	Path	Group	Last activity	MFA	Password age	Console last sign
jiro	/	0	20 days ago	-	-	
John	/	1	10 hours ago	-	12 hours	August 17, 2025,
Mary	/	1	49 minutes ago	-	11 hours	August 17, 2025,

- Click security credential and assign MFA.

Screenshot of the AWS IAM User Details page for user 'John'. The 'Security credentials' tab is selected. A red arrow points from the 'Assign MFA device' button at the top right of the 'Multi-factor authentication (MFA)' section to the 'Assign MFA device' button at the bottom of the same section.

- Set name and choose authenticator app and click next.

Screenshot of the 'Assign MFA device' step in the AWS IAM process. The 'Device name' input field contains 'John Authenticator'. The 'Authenticator app' option is selected in the 'MFA device' section. Red arrows point to both the 'Device name' field and the 'Authenticator app' option.

- Show QR, scan it and supply code as instructed.

The screenshot shows the 'Assign MFA device' step 2 page in the AWS IAM console. The steps are:

- Install a compatible application such as Google Authenticator, Duo Mobile, or Authy app on your mobile device or computer.
- Open your authenticator app, choose **Show QR code** on this page, then use the app to scan the code. Alternatively, you can type a secret key. [Show secret key](#)
- Type two consecutive MFA codes below

Below the steps, there are input fields for the first and second MFA codes, and a note to wait 30 seconds between entries. At the bottom are 'Cancel', 'Previous', and 'Add MFA' buttons.

- MFA successfully added.

The screenshot shows the 'Users' page in the AWS IAM console for the user 'John'. A green success message at the top states: 'MFA device assigned' with a note about registering up to 8 MFA devices. Below is the 'Summary' section:

ARN arn:aws:iam::536697231935:user/John	Console access Enabled with MFA	Access key 1 Create access key
Created August 17, 2025, 09:43 (UTC+01:00)	Last console sign-in Today	

Below the summary are tabs for 'Permissions', 'Groups (1)', 'Tags', 'Security credentials' (which is selected), and 'Last Accessed'. The 'Console sign-in' section shows a 'Console sign-in link' (https://536697231935.signin.aws.amazon.com/console) and a 'Console password' updated 12 hours ago. The 'Multi-factor authentication (MFA)' section shows one MFA device assigned (Virtual, Identifier: arn:aws:iam::536697231935:mfa/John-Authenticator, Created on: Sun Aug 17 2025). Buttons for 'Remove', 'Resync', and 'Assign MFA device' are available.

- Sign-in as John.

The screenshot shows a web browser window with the URL https://eu-north-1.signin.aws.amazon.com/oauth?client_id=arn%3Aaws%3Asignin%3A%3A%3Aconsole%2Fcanvas&code_challenge=91C0jWQYSgX3.... The page title is "IAM user sign in". It contains fields for "Account ID or alias" (536697231935), "Remember this account" (unchecked), "IAM username" (John), "Password" (redacted), "Show Password" (unchecked), and "Having trouble?". A red arrow points to the "Sign in" button. Below it is a link "Sign in using root user email". To the right of the sign-in form is a promotional banner for "Amazon Lightsail" with the text "Lightsail is the easiest way to get started on AWS" and a "Learn more »" button. The banner features a cartoon robot character and a blurred background of orange and yellow streaks. The browser's address bar and tab bar are visible at the top.

- Supply required MFA and login.

The screenshot shows a web browser window with the same URL as the previous screenshot. The page title is "Additional verification required". It states "Your account is protected with multi-factor authentication (MFA)." and instructs the user to enter the code from their MFA device. A red arrow points to the "Sign in" button. Below it are links "Sign in to a different account" and "Trouble signing in?". The right side of the screen is identical to the previous screenshot, showing the "Amazon Lightsail" banner with the robot character and blurred background. The browser's address bar and tab bar are visible at the top.

© 2025 Amazon Web Services, Inc. or its affiliates. All rights reserved.

- Successfully logged in.

The screenshot shows the AWS Console Home page. At the top right, it displays the account ID '5366-9723-1935' and the user name 'John'. A red arrow points from the text 'Successfully logged in.' to this user information. Below the header, there are sections for 'Recently visited' (IAM, S3, EC2), 'Applications' (Info, Region: Europe (Stockholm)), 'Welcome to AWS' (Getting started with AWS), 'AWS Health' (Info), and 'Cost and usage' (Info). The bottom of the page includes a feedback link, copyright information (© 2025, Amazon Web Services, Inc. or its affiliates.), and cookie preferences.

Mary (Data Analyst)

- Enabled MFA.
- Linked with **Microsoft Authenticator**.
- Tested successful login with MFA.

Steps:

- Click users to repeat the same step for Mary.

The screenshot shows the AWS IAM User details page for the user 'John'. On the left, a navigation menu is open under 'Access management', with the 'Users' option highlighted by a red arrow. The main content area shows a summary of the user's access status, including ARN, console access (enabled with MFA), and access keys. It also displays the last console sign-in (today) and a 'Console sign-in' section with a sign-in link and password information. At the bottom, there is a 'Multi-factor authentication (MFA)' section with a table showing one assigned MFA device (Virtual, Identifier: arn:aws:iam::536697231935:mfa/John-Authenticator). The page includes standard AWS footer links like CloudShell and Feedback.

- Click on Mary.

The screenshot shows the AWS IAM 'Users' page. On the left, there's a sidebar with 'Identity and Access Management (IAM)' and various management options like 'Access management', 'Access reports', and 'CloudShell'. The main area displays a table of users:

User name	Path	Group	Last activity	MFA	Password age	Console last sign
jiro	/	0	20 days ago	-	-	-
John	/	1	10 hours ago	-	12 hours	August 17, 2025,
Mary	/	1	1 hour ago	-	11 hours	August 17, 2025,

- Click security credential and assign MFA.

The screenshot shows the AWS IAM 'User details' page for 'Mary'. The left sidebar is identical to the previous screenshot. The main area has tabs for 'Permissions', 'Groups', 'Tags', and 'Security credentials'. The 'Security credentials' tab is selected, indicated by a blue underline. Below it, there's a 'Console sign-in' section with a 'Console sign-in link' and a 'Console password' section. At the bottom, there's a 'Multi-factor authentication (MFA) (0)' section with a 'Remove' and 'Resync' button, and a prominent 'Assign MFA device' button highlighted with a red arrow.

- Set name and choose authenticator app and click next.

MFA device name

Device name
This name will be used within the identifying ARN for this device.

Maximum 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

MFA device

Device options
In addition to username and password, you will use this device to authenticate into your account.

- Passkey or security key**
Authenticate using your fingerprint, face, or screen lock. Create a passkey on this device or use another device, like a FIDO2 security key.
- Authenticator app**
Authenticate using a code generated by an app installed on your mobile device or computer.
- Hardware TOTP token**
Authenticate using a code generated by Hardware TOTP token or other hardware devices.

Cancel **Next**

- Show QR, scan it and supply code as instructed.

Step 1
 Select MFA device
 Set up device

Set up device Info

Authenticator app
A virtual MFA device is an application running on your device that you can configure by scanning a QR code.

- Install a compatible application such as Google Authenticator, Duo Mobile, or Authy app on your mobile device or computer.
[See a list of compatible applications](#)
-
- Type two consecutive MFA codes below

Enter a code from your virtual app below

Wait 30 seconds, and enter a second code entry.

Cancel **Previous** **Add MFA**

- MFA successfully added.

MFA device assigned

You can register up to 8 MFA devices of any combination of the currently supported MFA types with your AWS account root and IAM user. With multiple MFA devices, you only need one MFA device to sign in to the AWS console or create a session through the AWS CLI with that user.

Summary

ARN arn:aws:iam::536697231935:user/Mary	Console access Enabled with MFA	Access key 1 Create access key
Created August 17, 2025, 10:53 (UTC+01:00)	Last console sign-in Today	

Security credentials

Console sign-in

Multi-factor authentication (MFA) (1)

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

Type	Identifier	Certifications	Created on
Virtual	arn:aws:iam::536697231935:mfa/Mary-Authenticator	Not Applicable	Sun Aug 17 2025

9. Project Reflection

9.1 Role of IAM in AWS

IAM defines **who** can access AWS and **what actions** they can perform. It enhances security, compliance, and efficient resource management.

9.2 Difference Between IAM Users and Groups

- IAM Users** → Individual identities (e.g., John, Mary).
- IAM Groups** → Collections of users with shared permissions (e.g., Development-Team, Analyst-Team). Groups simplify permission management at scale.

9.3 Process of Creating IAM Policies

- Identify required service.
- Select allowed actions.
- Define resources.
- Save as **custom policy**.
- Attach policy to a user or group.

9.4 Significance of Principle of Least Privilege

Granting users only the access they need minimizes risks and prevents accidental or malicious misuse of AWS resources.

9.5 Scenario with John and Mary

- **John** → User in Development-Team → EC2 access only.
 - **Mary** → User in Analyst-Team → S3 access only.
 - Both secured with **MFA**.
 - This setup mirrors real-world fintech security practices.
-

10. Conclusion

This project demonstrated how AWS IAM helps enforce **secure, role-based access control** for Zappy e-Bank. By applying IAM concepts such as **users, groups, policies, and MFA**, the project successfully built a secure environment that aligns with fintech compliance requirements and the **principle of least privilege**.