

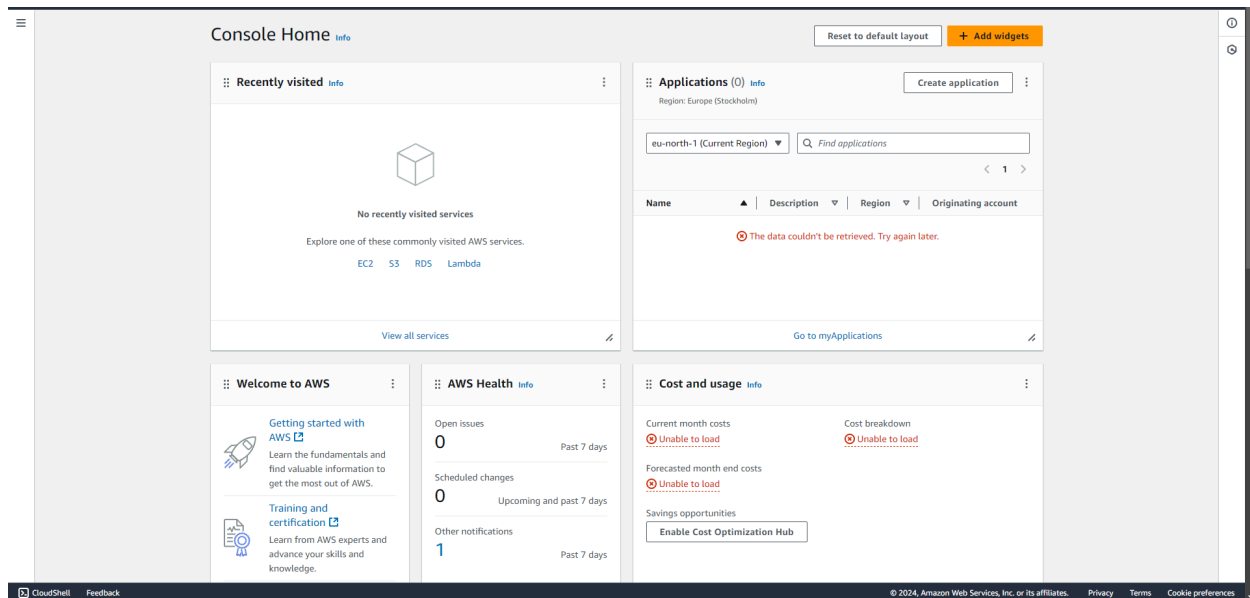
Name of Student: Pushkar Sane		
Roll Number: 45		Lab Assignment Number: 7
Title of Lab Assignment: Identity Access Management		
DOP: 21-10-2024		DOS: 21-10-2024
CO Mapped: CO5	PO Mapped: PO1, PO2, PO3, PO7, PO9, PSO1	Signature:

Practical No. 7

Aim: Identity Access Management

AWS

Amazon Web Services (AWS)



Amazon Web Services (AWS) is a comprehensive and widely adopted cloud computing platform offered by Amazon. It provides a variety of cloud services, enabling businesses and individuals to access computing power, storage, and other functionalities on-demand via the internet.

AWS Security

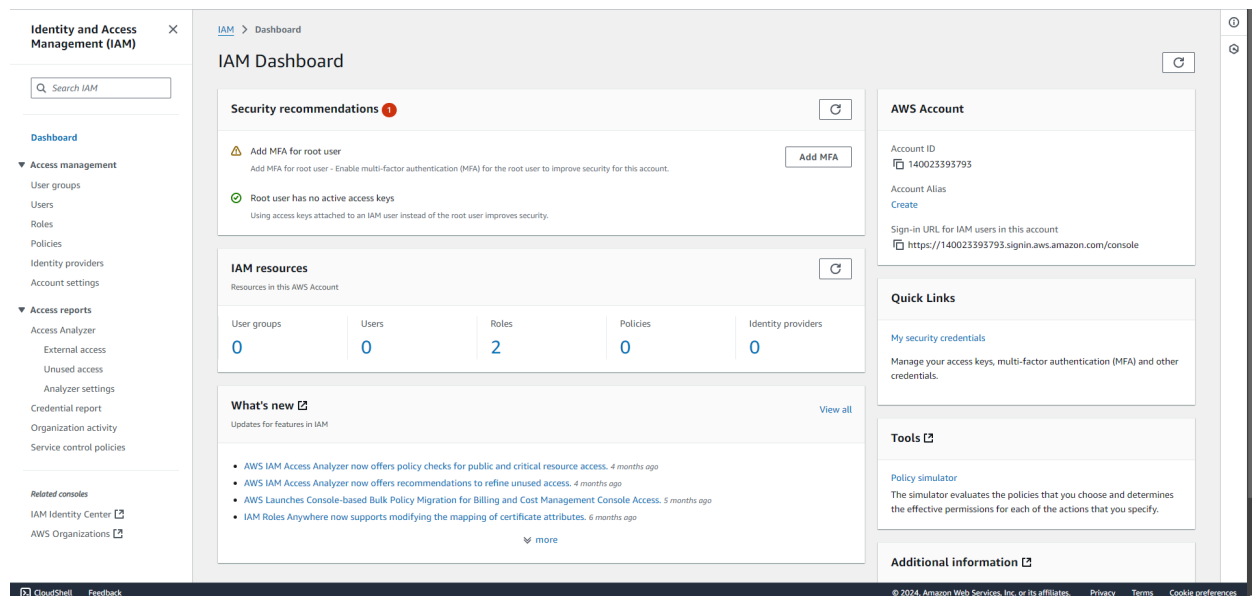
AWS Security is a comprehensive framework that includes tools, policies, and best practices to protect customer data, applications, and services within the AWS cloud environment. AWS adopts a **Shared Responsibility Model**, where:

- **AWS's Responsibilities:** AWS manages the security of the cloud infrastructure, which includes hardware, software, networking, and facilities. This involves maintaining the physical security of data centers, patching underlying infrastructure, and ensuring compliance with various security standards.
- **Customer's Responsibilities:** Customers are responsible for securing their applications, managing access to their data, configuring security controls (like IAM), and ensuring compliance with relevant regulations.

AWS provides various security services and features, including:

- **Identity and Access Management (IAM):** Manage access to AWS services and resources.
- **AWS Shield and AWS WAF:** Protect applications against Distributed Denial of Service (DDoS) attacks and web exploits.
- **AWS Key Management Service (KMS):** Manage cryptographic keys for data encryption.
- **Amazon CloudTrail:** Track user activity and API usage for auditing purposes.

What is IAM?



Identity and Access Management (IAM) is a critical service within AWS that allows administrators to control user access to AWS services and resources securely. IAM provides the following capabilities:

- **User Management:** Create and manage AWS users and their associated access permissions.
- **Access Control:** Define who can access what resources and under what conditions.
- **Policy Management:** Create policies that dictate permissions for actions on specific resources.

IAM ensures that users have the minimum level of access necessary to perform their jobs (principle of least privilege), which helps reduce the risk of unauthorized access.

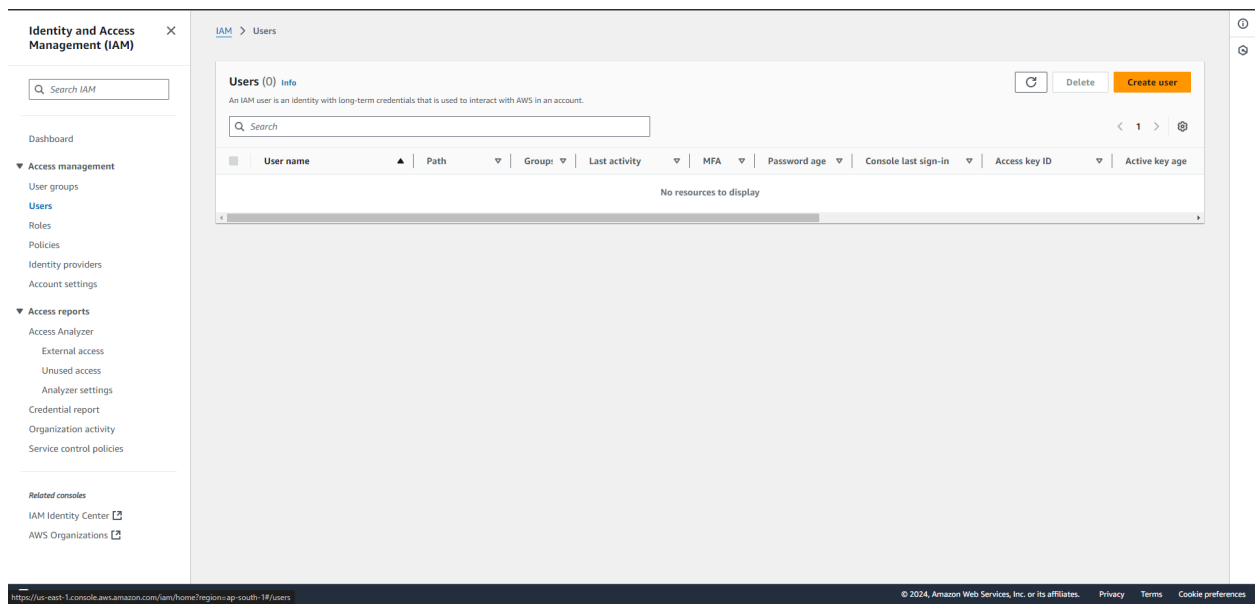
How Does IAM Work?

IAM operates through several key processes and concepts:

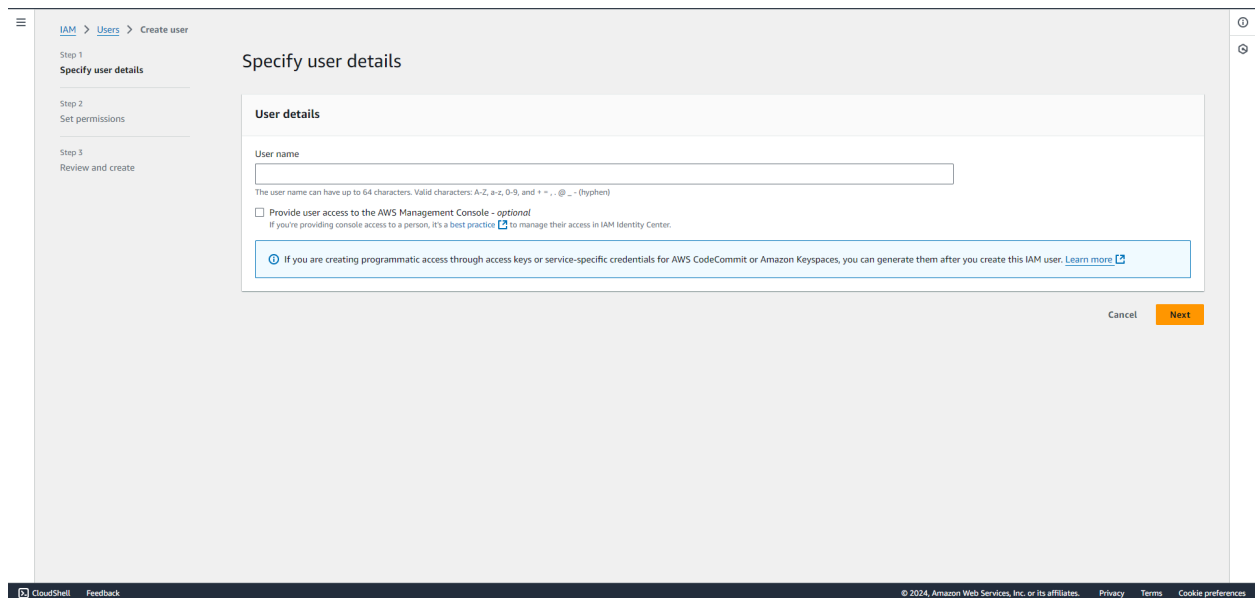
1. **User Creation:** Administrators can create IAM users, each with unique security credentials (username and password) to access AWS resources.
2. **Groups:** Users can be organized into groups to simplify permission management. Instead of assigning permissions to each user individually, permissions can be assigned to a group, which then applies to all users in that group.
3. **Policies:** IAM uses policies to grant or restrict access to resources. Policies are JSON documents that specify:
 - **Effect:** Whether the policy allows or denies access.
 - **Action:** The specific actions (e.g., s3:PutObject, ec2:StartInstances) the policy permits or denies.
 - **Resource:** The specific AWS resources (e.g., Amazon S3 buckets, EC2 instances) to which the actions apply.
4. **Roles:** Roles are IAM identities that have specific permissions but are not associated with a specific user. Instead, roles can be assumed by users, applications, or AWS services, enabling temporary access without needing to manage long-term credentials.
5. **Authentication and Authorization:** When a user or application requests access to an AWS resource, IAM first authenticates the identity (verifying the provided credentials). Then, it checks the permissions associated with that identity against the requested action and resource to determine if access should be granted or denied.

Create a New IAM User

1. **Select Users:** In the left navigation pane, click on **Users**.



2. **Add User:** Click the **Create user** button.



3. **Enter User Details:**

- **User Name:** Provide a unique username for the new user.
- **Access Type:** Choose the type of access:
 - **Programmatic access:** Provides an Access Key ID and Secret Access Key for AWS CLI, SDK, and APIs.
 - **AWS Management Console access:** Allows the user to sign in to the console. If selected, set a password (either autogenerate or create a custom one).

4. **Click Next:** After filling in the details, click the **Next: Permissions** button.

Step 4: Set Permissions for the User

1. Attach Existing Policies:

- Choose from existing policies to assign specific permissions. You can filter policies based on service or type.
- Select the appropriate policies based on the user's role (e.g., Administrator, Read-only, etc.).

2. Create Group (Optional):

- You can create a group and assign policies to that group, then add the user to the group.

- To do this, click on **Create group**, enter a group name, select policies, and click **Create group**.
- Select the group you created to add the user to that group.

Group1 user group created.

IAM > Users > Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

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 (1/1)

Search

<input checked="" type="checkbox"/>	Group name	Users	Attached policies	Created
<input checked="" type="checkbox"/>	Group1	0	AlexaForBusinessGatewayExecution, AlexaForB...	2024-10-21 (Now)

► **Set permissions boundary - optional**

Cancel Previous **Next**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create user group

Create a user group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

User group name
Enter a meaningful name to identify this group.
Group1
Maximum 128 characters. Use alphanumeric and "+-@._" characters.

Permissions policies (957)

Search Filter by Type All type

<input type="checkbox"/>	Policy name	Type	Use...	Description
<input type="checkbox"/>	AdministratorAccess	AWS managed ...	None	Provides full access to AWS services
<input type="checkbox"/>	AdministratorAcce...	AWS managed	None	Grants account administrative perm
<input type="checkbox"/>	AdministratorAcce...	AWS managed	None	Grants account administrative perm
<input type="checkbox"/>	AlexaForBusinessD...	AWS managed	None	Provide device setup access to Alex
<input type="checkbox"/>	AlexaForBusinessF...	AWS managed	None	Grants full access to AlexaForBusin
<input type="checkbox"/>	AlexaForBusinessG...	AWS managed	None	Provide gateway execution access t
<input type="checkbox"/>	AlexaForBusinessL...	AWS managed	None	Provide access to Lifesize AVS devic
<input type="checkbox"/>	AlexaForBusinessP...	AWS managed	None	Provide access to Poly AVS devices
<input type="checkbox"/>	AlexaForBusinessR...	AWS managed	None	Provide read only access to AlexaFc
<input type="checkbox"/>	AmazonAPIGateway...	AWS managed	None	Provides full access to create/edit/i
<input type="checkbox"/>	AmazonAPIGateway...	AWS managed	None	Provides full access to invoke APIs i

Cancel **Create user group**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

3. Add Inline Policy (Optional):

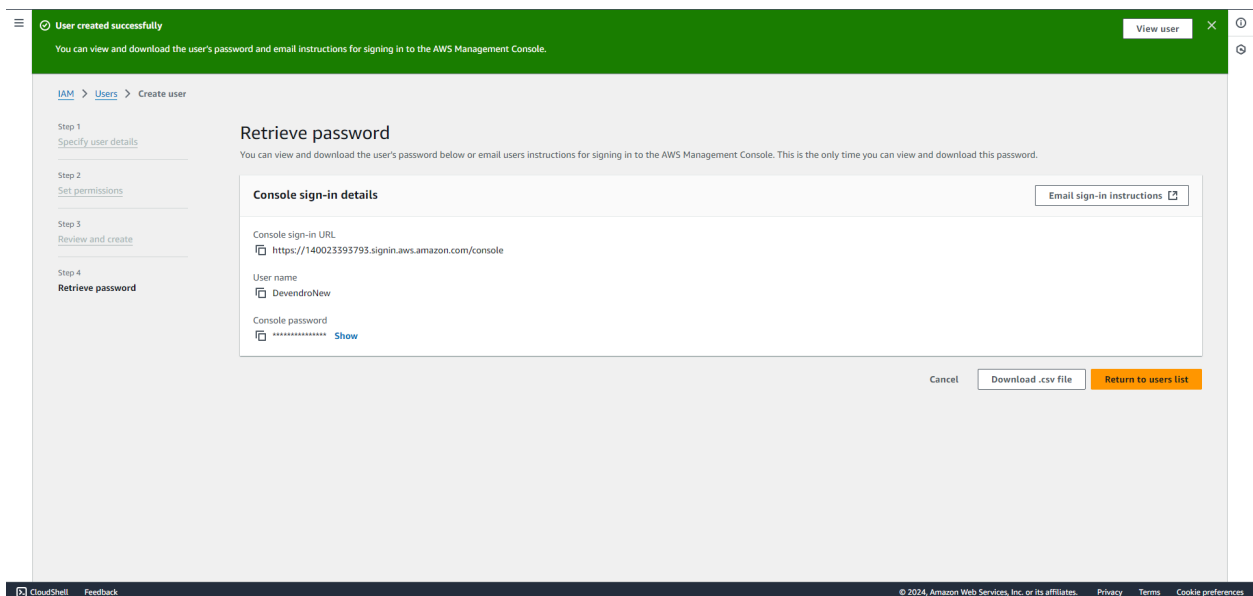
- If you need specific permissions not covered by existing policies, you can create an inline policy.
- Click **Next: Tags** to proceed.

Step 5: (Optional) Add Tags

1. **Add Tags:** Tags are key-value pairs used for organizing and managing IAM resources. You can add tags to the user for easier identification.
2. **Click Next:** After adding tags (if any), click **Next: Review**.

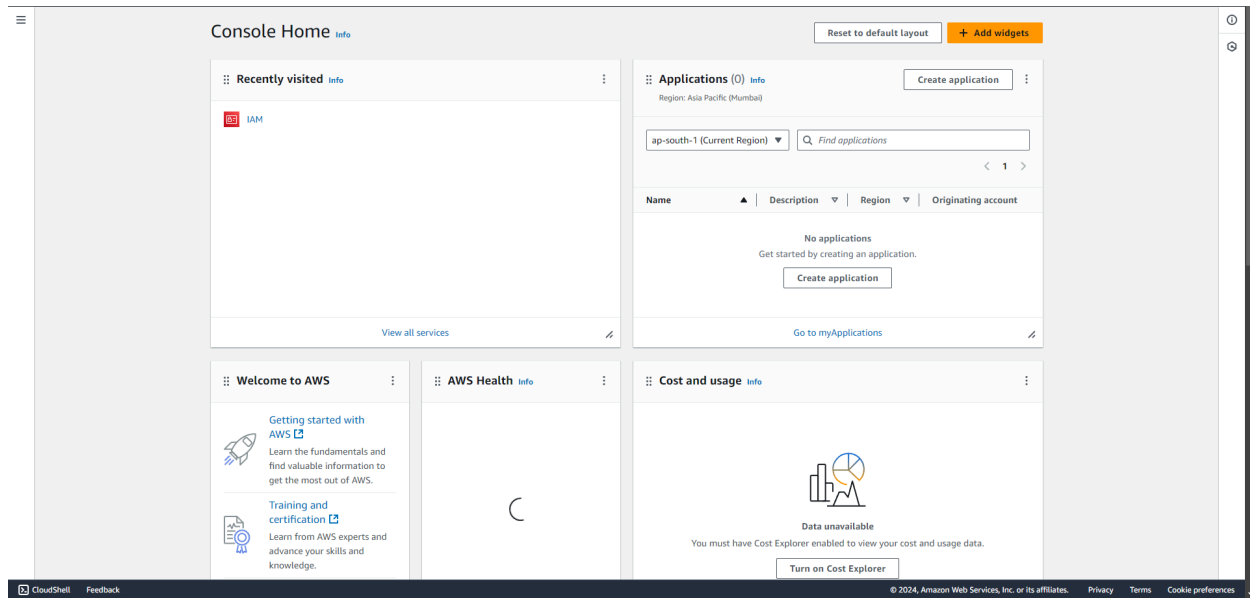
Step 6: Review and Create User

1. **Review Settings:** Check the user details, permissions, and tags.
2. **Create User:** Click the **Create user** button.



Step 7: Sign In as the New User

1. **Sign-In Link:** To log in to the AWS Management Console as the new user, use the sign-in link provided on the confirmation page or generate it from the IAM dashboard.
2. **Enter Credentials:** Use the new username and password (if console access was enabled) to sign in.



Features of IAM

IAM offers numerous features that enhance security and streamline management:

1. **Granular Permissions:** IAM policies allow administrators to specify detailed permissions at various levels (individual actions on specific resources), providing fine-grained control over access.
2. **Centralized Management:** IAM provides a unified interface to manage users, groups, roles, and permissions across all AWS services.
3. **Role-Based Access Control:** The ability to create roles simplifies permission management, especially in scenarios where temporary access is needed (e.g., granting AWS Lambda functions access to S3).
4. **Multi-Factor Authentication (MFA):** MFA can be enforced to enhance security, requiring users to authenticate using a second factor in addition to their password.
5. **Temporary Credentials:** IAM roles allow the provision of temporary security credentials for users and applications, reducing the risks associated with long-term credentials.
6. **Policy Versioning:** IAM supports versioning of policies, enabling administrators to update permissions while preserving older versions for auditing or rollback purposes.
7. **Audit and Monitoring:** AWS CloudTrail can track IAM actions, allowing organizations to monitor access and changes to IAM settings, providing visibility into who accessed what resources and when.

Multi-Factor Authentication (MFA)

Multi-Factor Authentication (MFA) is a crucial security feature in IAM that adds an extra layer of protection. MFA requires users to provide two or more verification factors when logging in, significantly reducing the risk of unauthorized access.

How MFA Works:

1. MFA Device Options:

- **Virtual MFA:** Users can install a virtual MFA application (e.g., Google Authenticator, AWS MFA) on their mobile device. The app generates a time-based one-time password (TOTP).
- **Hardware MFA:** Physical devices (e.g., YubiKey) can be used to generate authentication codes.
- **SMS MFA:** Users can receive a one-time code via SMS. While convenient, this method is less secure than the others.

2. Enabling MFA: Administrators can enforce MFA for specific users or groups. When MFA is enabled for an IAM user, the user must configure their MFA device and use it during the login process.

3. Authentication Process:

- During login, the user enters their username and password.
- After successful password entry, they are prompted to enter the MFA code generated by their MFA device.
- The code is time-sensitive (typically valid for 30 seconds), requiring users to enter the latest code.

4. Role Assumption with MFA: IAM roles can also require MFA for access. When a user assumes a role, they must provide the MFA code along with their request, ensuring that only authorized users can access sensitive resources.

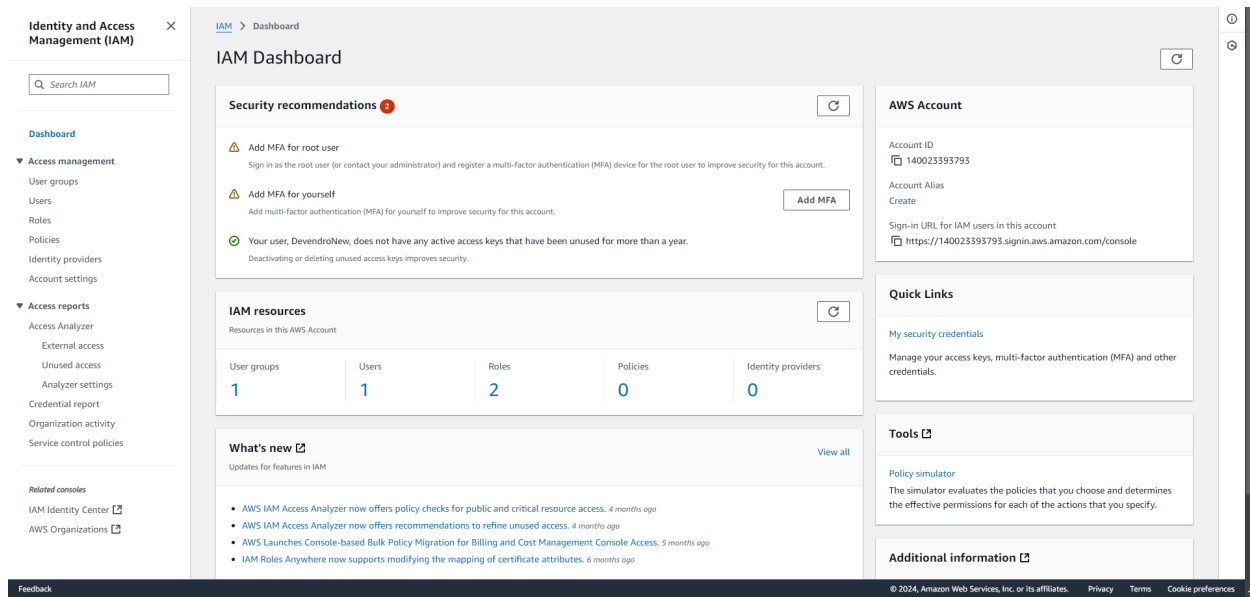
5. Use Cases for MFA:

- Accessing AWS Management Console.
- Using AWS CLI or SDKs for programmatic access.
- Assuming IAM roles for sensitive operations.

How to Enable MFA

Step 1: Access the IAM Dashboard

1. **Navigate to IAM:** In the AWS Management Console, type "IAM" in the search bar and select **IAM** from the results.
2. **IAM Dashboard:** You will see the IAM dashboard where you can manage users, groups, roles, and policies.



Step 2: Select the User

1. **Choose Users:** In the left navigation pane, click on **Users**.
2. **Select the User:** Click on the name of the user for whom you want to enable MFA.

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users**
- Roles
- Policies
- Identity providers
- Account settings

Access reports

- Access Analyzer
- External access
- Unused access
- Analyzer settings
- Credential report
- Organization activity
- Service control policies

Related consoles

- IAM Identity Center
- AWS Organizations

DevendroNew info Delete

Summary

ARN: `arn:aws:iam::140023393793:user/DevendroNew`

Created: October 21, 2024, 03:01 (UTC+05:30)

Console access: Enabled without MFA

Last console sign-in: Today

Access key 1: [Create access key](#)

Permissions | Groups (1) | Tags | **Security credentials** | Last Accessed

Permissions policies

Permissions are defined by policies attached to the user directly or through groups.

Search: Filter by Type: All types

Policy name	Type	Attached via
Loading policies		

► **Permissions boundary** (not set)

▼ **Generate policy based on CloudTrail events**

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Step 3: Enable MFA

- Security Credentials Tab:** Navigate to the **Security credentials** tab for the selected user.
- Manage MFA Device:** Find the **Multi-Factor Authentication (MFA)** section and click on **Manage** next to it.

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users**
- Roles
- Policies
- Identity providers
- Account settings

Access reports

- Access Analyzer
- External access
- Unused access
- Analyzer settings
- Credential report
- Organization activity
- Service control policies

Related consoles

- IAM Identity Center
- AWS Organizations

DevendroNew info Delete

Summary

ARN: `arn:aws:iam::140023393793:user/DevendroNew`

Created: October 21, 2024, 03:01 (UTC+05:30)

Console access: Enabled without MFA

Last console sign-in: Today

Access key 1: [Create access key](#)

Permissions | Groups (1) | Tags | **Security credentials** | Last Accessed

Permissions policies (11)

Permissions are defined by policies attached to the user directly or through groups.

Search: Filter by Type: All types

Policy name	Type	Attached via
AdministratorAccess	AWS managed - job function	Group Group1
AdministratorAccess-Amplify	AWS managed	Group Group1
AdministratorAccess-AWSElasticBeanstalk	AWS managed	Group Group1
AlexaForBusinessDeviceSetup	AWS managed	Group Group1
AlexaForBusinessFullAccess	AWS managed	Group Group1
AlexaForBusinessGatewayExecution	AWS managed	Group Group1

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

IAM > Users > DevendraNew > Assign MFA device

Step 1
Select MFA device

Step 2
Set up device

Select MFA device [Info](#)

MFA device name

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

Device name

Maximum 64 characters. Use alphanumeric and '+', '-', '@', '.', '_' characters.

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.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 4: Choose MFA Device Type

- Select MFA Device Type:** You will be presented with different MFA device options. Choose one of the following:
 - Virtual MFA device:** This option allows you to use an authenticator app (like Google Authenticator, Authy, or AWS Virtual MFA).
 - U2F security key:** If you have a hardware security key (like YubiKey), you can select this option.
 - SMS MFA:** You can receive a one-time code via SMS to your mobile phone (less recommended due to security concerns).
- Click Next:** After selecting the MFA device type, click **Next**.

Step 1
Select MFA device

Step 2
Set up device

Select MFA device [Info](#)

MFA device name

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

Device name
Maximum 64 characters. Use alphanumeric and '+', '-', '.', '@', '_', '-' characters.

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

cloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 5: Configure the MFA Device

For Virtual MFA Device:

- 1. Install Authenticator App:** Ensure you have an authenticator app installed on your smartphone.
- 2. Scan QR Code:**
 - You will see a QR code on the screen. Open your authenticator app and scan the QR code.
 - Alternatively, you can enter the provided secret key manually into your app.
- 3. Enter MFA Codes:**
 - In the IAM console, enter two consecutive one-time passwords (OTPs) generated by your authenticator app.
 - These codes must be entered within a short time frame (usually within 30 seconds).
- 4. Click Assign MFA:** Once both codes are validated, click **Assign MFA** to complete the setup.

The screenshot shows the 'Set up device' page in the AWS IAM console for the user 'DevendroNew'. The page is divided into two steps: Step 1 'Select MFA device' and Step 2 'Set up device'. Step 2 is active and contains three numbered instructions: 1. Install a compatible application (Google Authenticator, Duo Mobile, or Authy). 2. Open the authenticator app and scan the QR code. 3. Type two consecutive MFA codes. A 'Show QR code' button is visible. At the bottom, there are 'Cancel', 'Previous', and 'Add MFA' buttons.

IAM > Users > DevendroNew > Assign MFA device

Step 1
[Select MFA device](#)

Step 2
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.

- 1 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](#)
- 2 [Show QR code](#)
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](#)
- 3 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**

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 6: Confirmation

- **Success Message:** Once MFA is successfully enabled, you'll see a confirmation message indicating that MFA has been configured for the user.

The screenshot shows the 'DevendroNew' user page in the AWS IAM console. A green banner at the top states 'MFA device assigned'. The page has a left sidebar with navigation links. The main content area shows a 'Summary' section with user details and a 'Security credentials' tab. The 'Security credentials' tab shows 'Console sign-in' and 'Multi-factor authentication (MFA) (1)'. The 'MFA' section shows 'Remove', 'Resync', and 'Assign MFA device' buttons.

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users**
- Roles
- Policies
- Identity providers
- Account settings

Access reports

- Access Analyzer
- External access
- Unused access
- Analyzer settings
- Credential report
- Organization activity
- Service control policies

Related consoles

- IAM Identity Center
- AWS Organizations

IAM > Users > DevendroNew

DevendroNew info [Delete](#)

Summary

ARN arn:aws:iam::140023393793:user/DevendroNew	Console access Enabled with MFA	Access key 1 Create access key
Created October 21, 2024, 03:01 (UTC+05:30)	Last console sign-in Today	

Permissions Groups (1) Tags **Security credentials** Last Accessed

Console sign-in

[Manage console access](#)

Console sign-in link
[https://140023393793.signin.aws.amazon.com/console](#)

Console password
Updated 6 minutes ago (2024-10-21 03:03 GMT+5:30)

Last console sign-in
[6 minutes ago \(2024-10-21 03:02 GMT+5:30\)](#)

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](#)

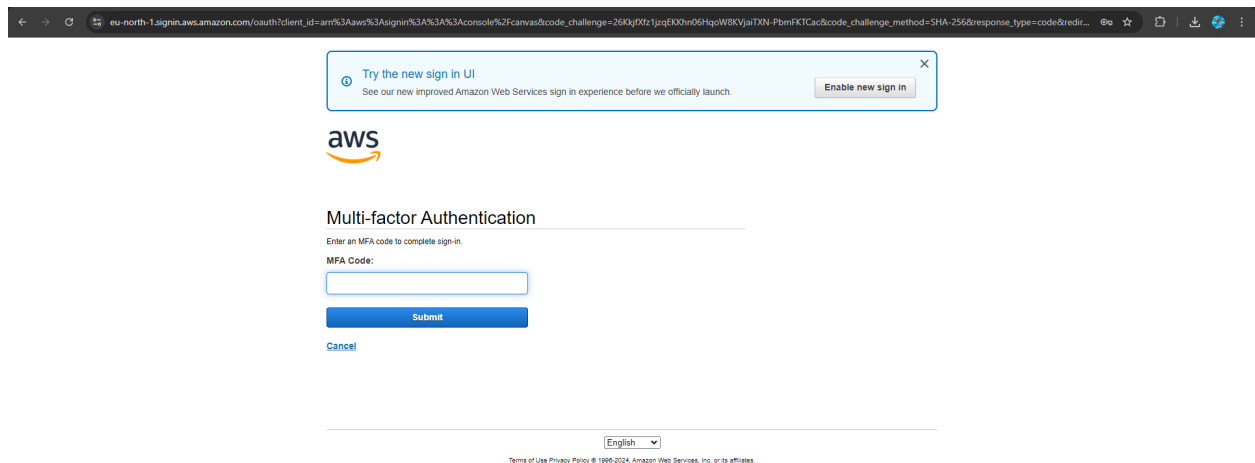
[Remove](#) [Resync](#) [Assign MFA device](#)

Type Identifier Certifications Created on

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 7: Sign In Using MFA

1. **Sign Out and Sign In Again:** Log out of the AWS Management Console and attempt to sign in again with the IAM user.
2. **Enter MFA Code:** After entering the username and password, you will be prompted to enter the MFA code generated by your authenticator app or received via SMS.
3. **Access AWS Resources:** Once the correct MFA code is entered, you will gain access to the AWS Management Console.



eu-north-1.signin.aws.amazon.com/oauth?client_id=arn%3Aaws%3A%3A%3A%3A%3Aconsole%2Fcanvas&code_challenge=26KxjMz1jqEKXh06HqoW8KVaiTXN-PbmFKTCac&code_challenge_method=SHA-256&response_type=code&redir...

Try the new sign in UI
See our new improved Amazon Web Services sign in experience before we officially launch. [Enable new sign in](#)

aws

Multi-factor Authentication

Enter an MFA code to complete sign-in.

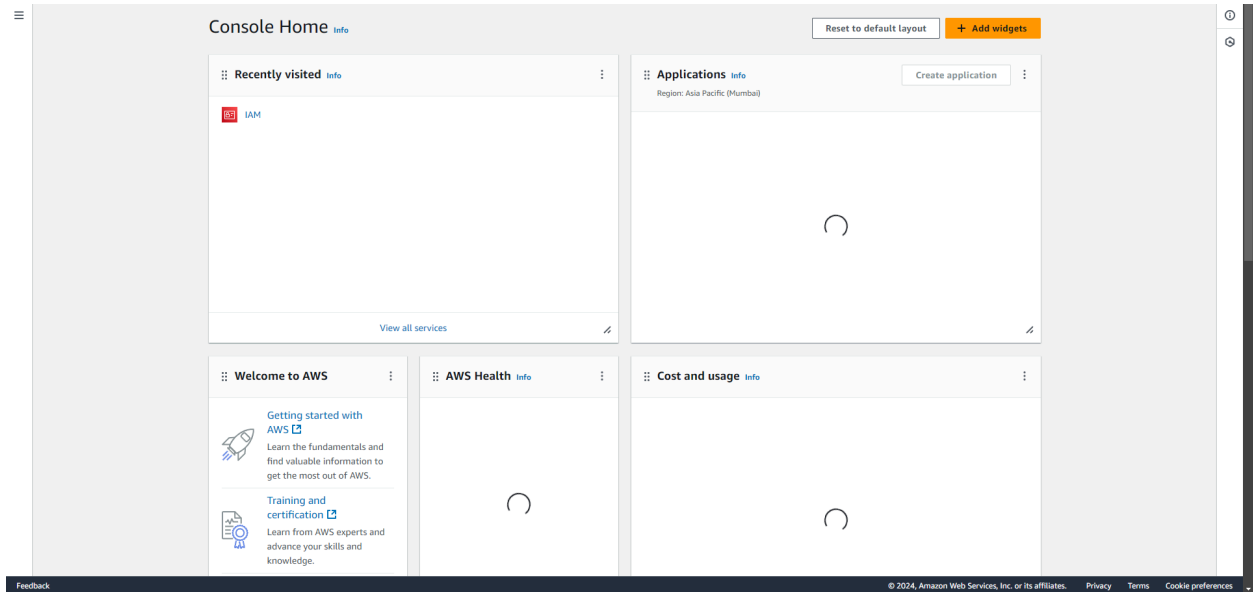
MFA Code:

[Submit](#)

[Cancel](#)

English

[Terms of Use](#) [Privacy Policy](#) © 1996-2024 Amazon Web Services, Inc. or its affiliates



Conclusion:

AWS Security, particularly through IAM, plays a vital role in protecting cloud resources. By leveraging IAM's features, organizations can implement stringent access controls, enhance security with Multi-Factor Authentication, and ensure that users have appropriate permissions based on their roles. This robust framework not only helps safeguard sensitive data but also streamlines the management of user access across the AWS ecosystem, aligning with best practices for cloud security.