Lab-1
Implement user authentication in AWS using IAM.

Group-1
(AP25122050002,AP25122050004,AP25122050005)

Introduction

This document describes the implementation of user authentication in AWS using IAM with mandatory Multi-Factor Authentication (MFA) enforced for all users.
The implementation uses a laptop-based passkey (FIDO2 security key / platform authenticator) as the second authentication factor.

The objective is to prevent unauthorized access by ensuring that no IAM user can access AWS resources without MFA.

Scope of Implementation

The scope of this implementation includes:

- Creation of IAM users and groups
- Configuration of passkey-based MFA
- Enforcement of MFA using IAM policies
- Secure authentication flow validation
- Submission of implementation to a Git repository

System Architecture

AWS IAM is used to manage authentication and authorization. MFA enforcement is implemented using a deny-based IAM policy applied at the group level.

Architecture Components:

- IAM Users - “Alice”, “Bob”, “Cipher”
- IAM Group - “`MFA-Users`”
- IAM MFA Enforcement Policy
- FIDO2 Passkey (Laptop Platform Authenticator)

Authentication Mechanism

Primary Authentication Factor

Username and password (IAM user credentials)

Secondary Authentication Factor

Passkey (FIDO2 / WebAuthn)

Stored securely on the user’s laptop

Unlocked using device PIN or biometric authentication

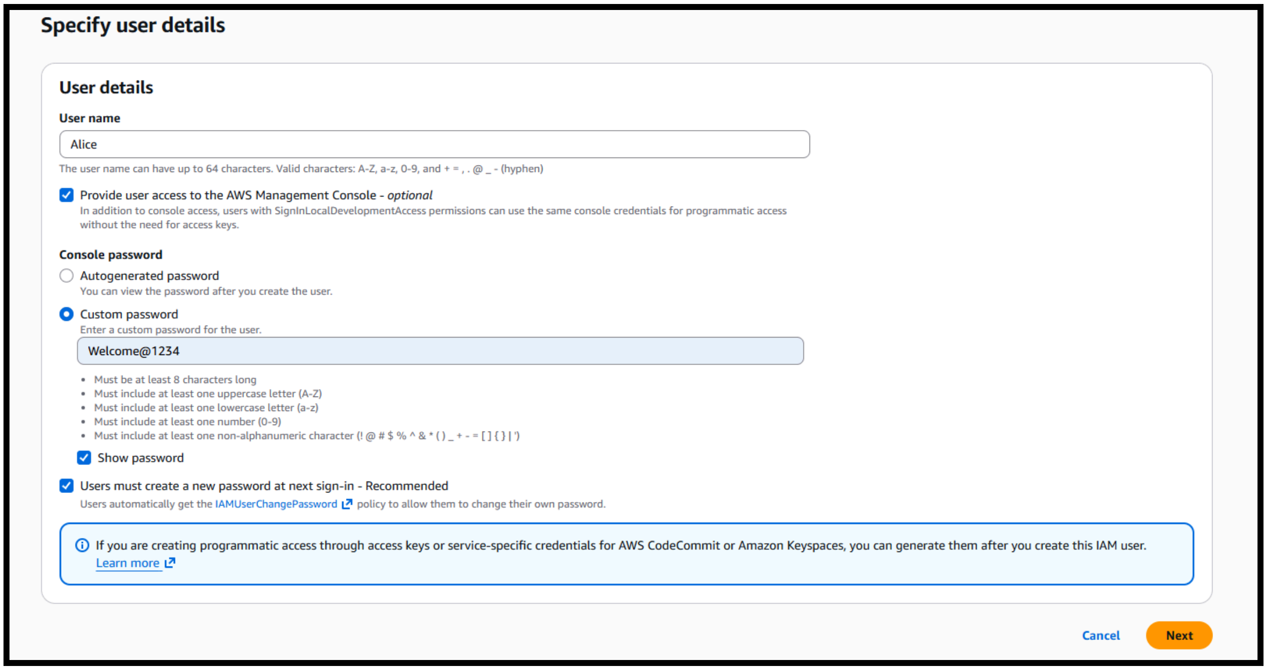
Implementation Procedure:

IAM User Creation:

IAM users were created for individual users requiring access to AWS services.

Procedure:

- Navigate to IAM → Users
- Create user with console access
- Configure password authentication



Specify user details

User details

User name

Alice

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*
In addition to console access, users with SignInLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys.

Console password

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

● Custom password
Enter a custom password for the user.

Welcome@1234

- Must be at least 8 characters long
- Must include at least one uppercase letter (A-Z)
- Must include at least one lowercase letter (a-z)
- Must include at least one number (0-9)
- Must include at least one non-alphanumeric character (! @ # $ % ^ & * ( ) _ + - = [ ] { } | ')

☑ 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.

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

IAM Group Configuration:

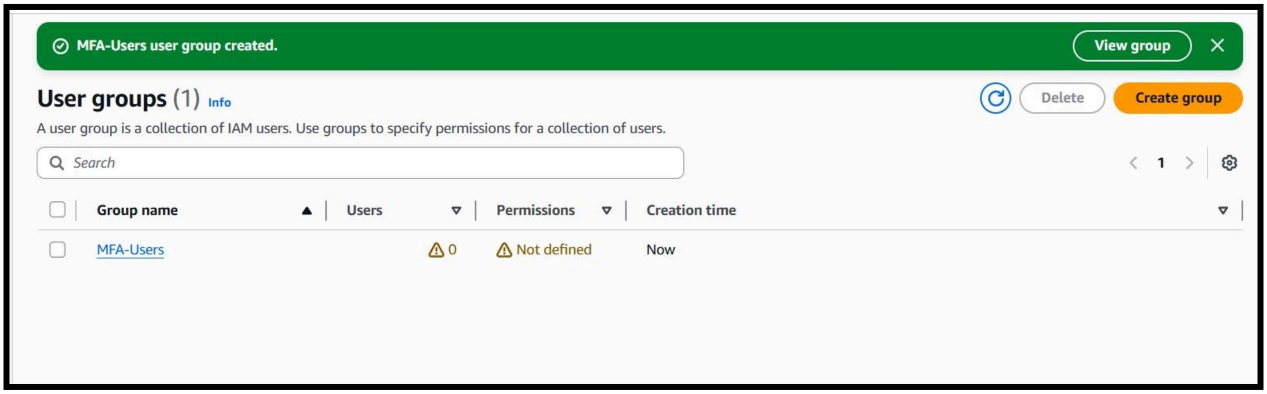
A group named “`MFA-Users`” was created to manage permissions centrally.

Purpose:

- Simplified policy management

- Consistent security enforcement

All IAM users were added to this group.



MFA-Users user group created.

User groups (1) Info

A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

☐	Group name	Users	Permissions	Creation time
☐	MFA-Users	⚠ 0	⚠ Not defined	Now

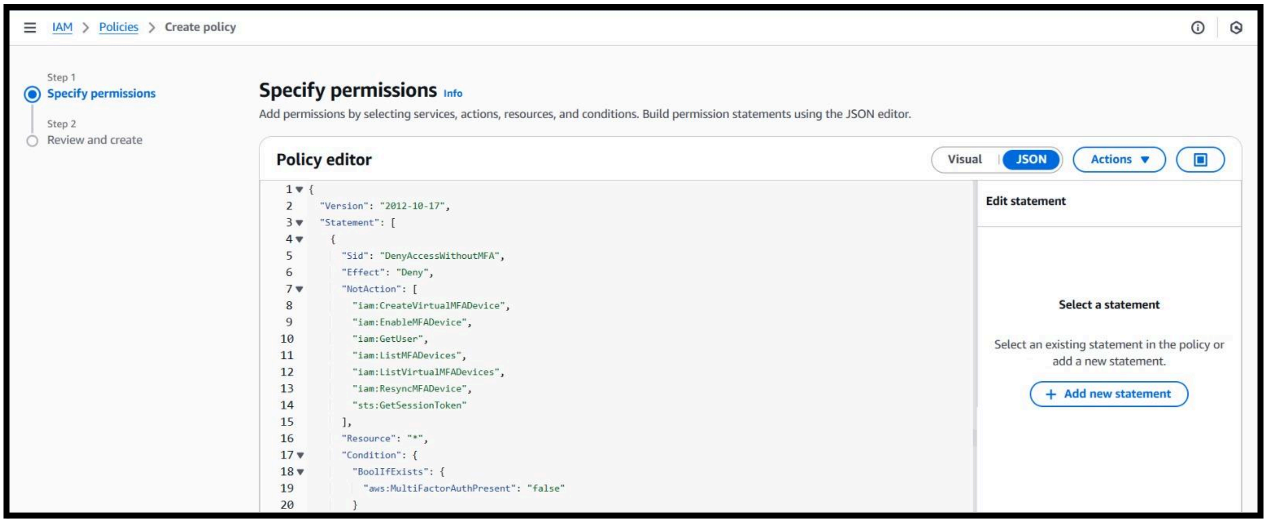
MFA Enforcement Policy:

A custom IAM policy was created to deny all AWS actions unless MFA is present.

MFA Enforcement Policy (JSON)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAccessWithoutMFA",
      "Effect": "Deny",
      "NotAction": [
        "iam:CreateVirtualMFADevice",
        "iam:EnableMFADevice",
        "iam:GetUser",
        "iam:ListMFADevices",
        "iam:ListVirtualMFADevices",
        "iam:ResyncMFADevice",
        "sts:GetSessionToken"
      ],
      "Resource": "*",
      "Condition": {
        "BoolIfExists": {
          "aws:MultiFactorAuthPresent": "false"
        }
      }
    }
  ]
}
```

This policy was attached to the “`MFA-Users`” group.



Specify permissions Info

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

Policy editor

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAccessWithoutMFA",
      "Effect": "Deny",
      "NotAction": [
        "iam:CreateVirtualMFADevice",
        "iam:EnableMFADevice",
        "iam:GetUser",
        "iam:ListMFADevices",
        "iam:ListVirtualMFADevices",
        "iam:ResyncMFADevice",
        "sts:GetSessionToken"
      ],
      "Resource": "*",
      "Condition": {
        "BoolIfExists": {
          "aws:MultiFactorAuthPresent": "false"
        }
```

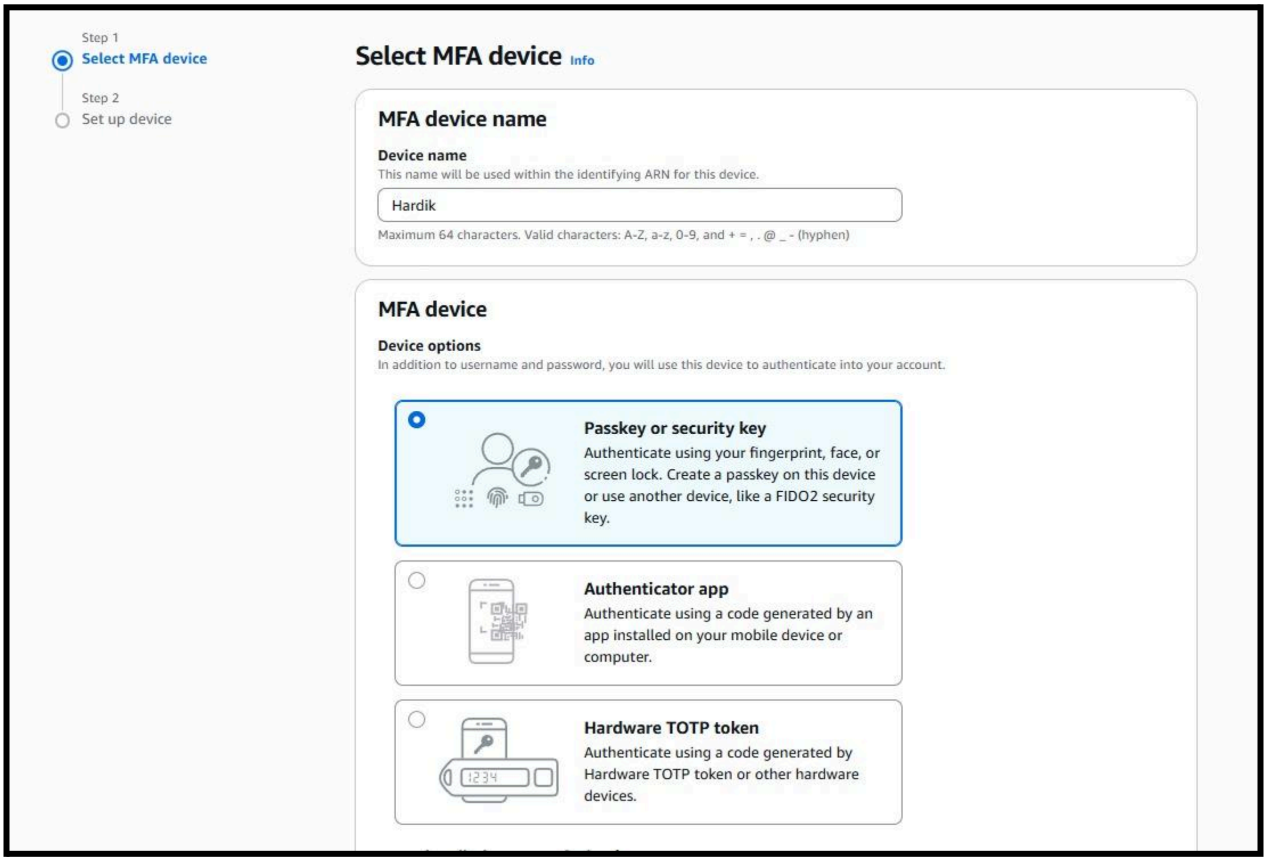
Passkey (FIDO2) MFA Configuration:

Each IAM user was assigned a passkey-based MFA device.

Procedure:

- IAM → User → Security credentials
- Assign MFA device
- Select **Security key (FIDO2)**
- Register laptop passkey

- Verify using device authentication



Select MFA device

MFA device name

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

Hardik

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

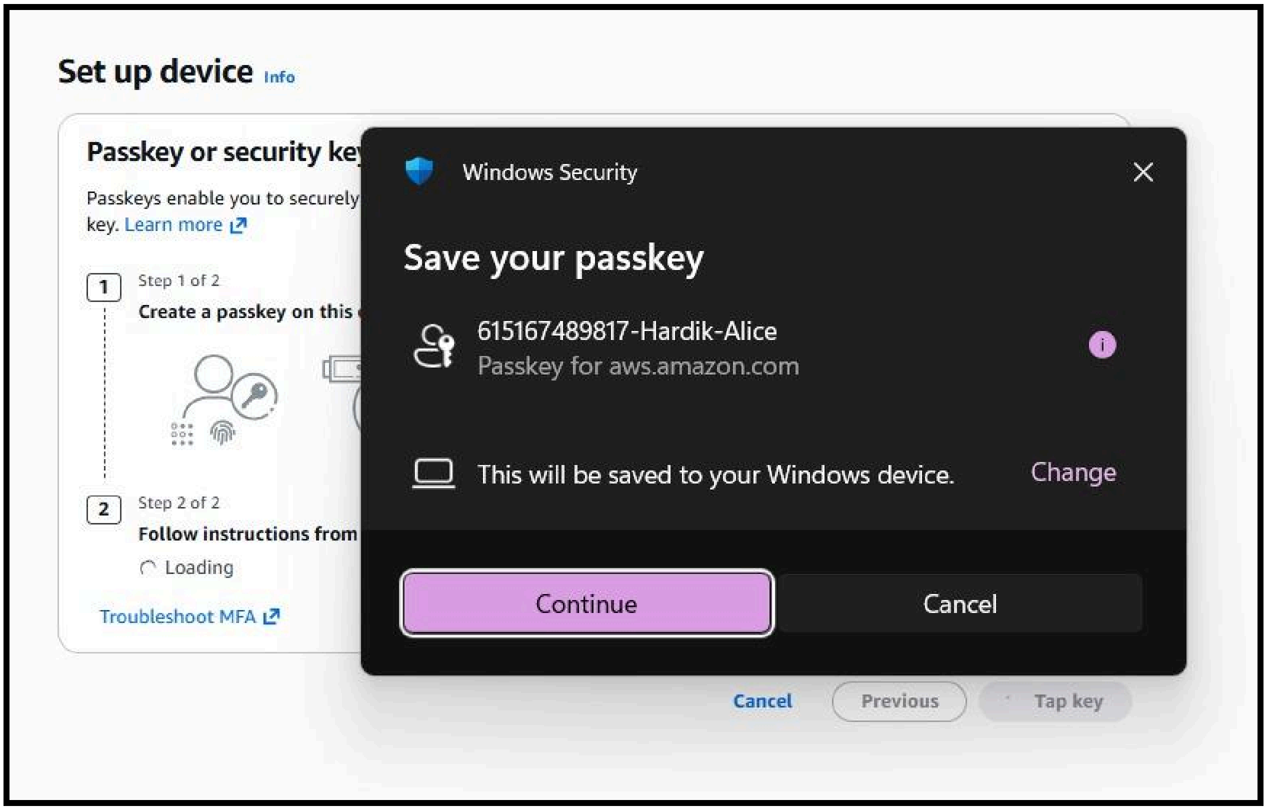
MFA device

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.



Set up device Info

Passkey or security ke

Passkeys enable you to securely

key. Learn more

1 Step 1 of 2

Create a passkey on this

2 Step 2 of 2

Follow instructions from

Loading

Troubleshoot MFA

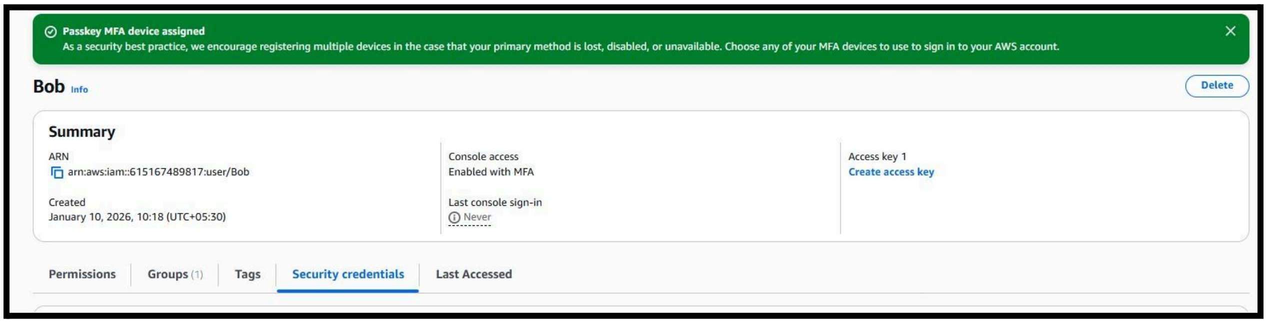
Windows Security

Save your passkey

615167489817-Hardik-Alice

Passkey for aws.amazon.com

This will be saved to your Windows device.



Passkey MFA device assigned

As a security best practice, we encourage registering multiple devices in the case that your primary method is lost, disabled, or unavailable. Choose any of your MFA devices to use to sign in to your AWS account.

Bob Info

Summary

ARN
arn:aws:iam::615167489817:user/Bob

Created
January 10, 2026, 10:18 (UTC+05:30)

Console access
Enabled with MFA

Last console sign-in
Never

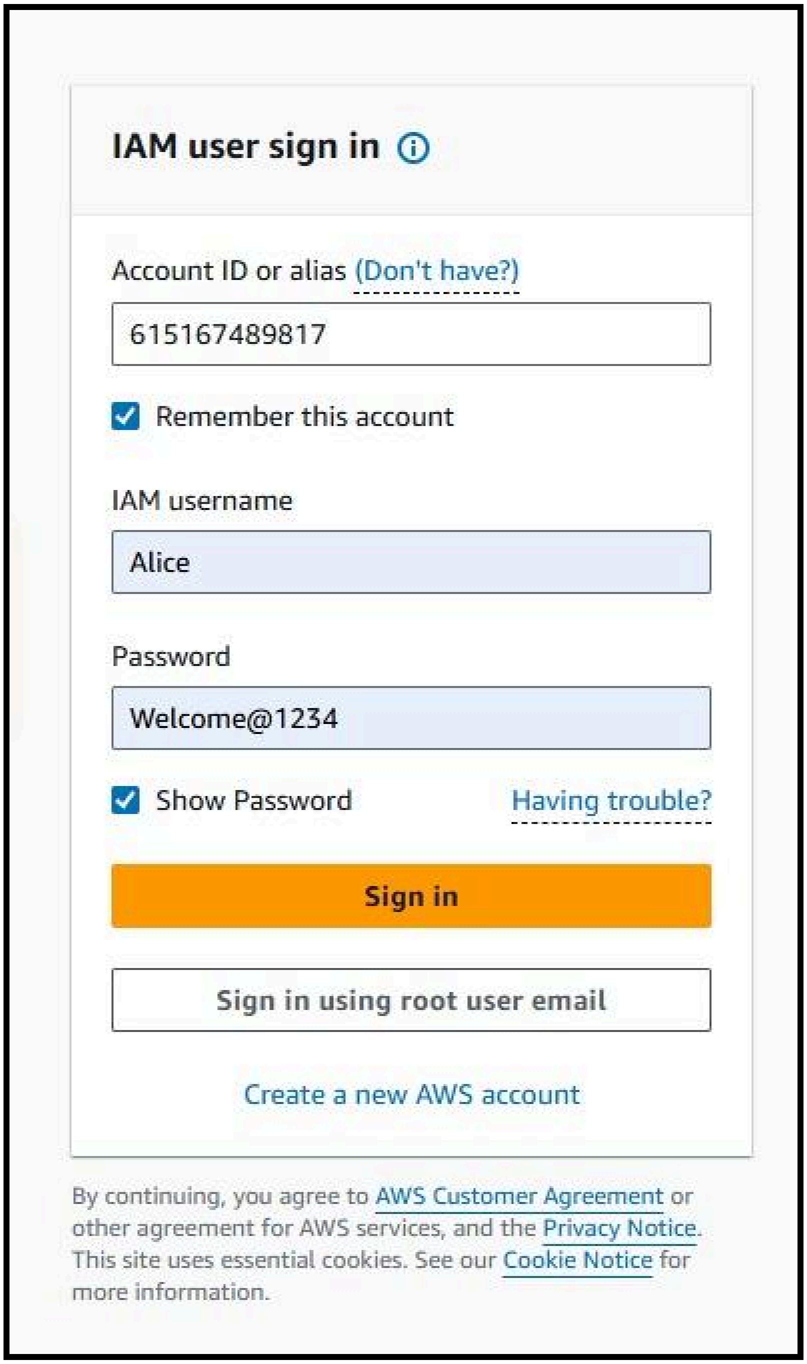
Access key 1
Create access key

Authentication Flow

Flow Description:

1. User enters IAM username and password
2. AWS validates credentials

3. Passkey challenge is triggered
4. User authenticates using laptop PIN/biometric
5. IAM policy verifies MFA presence
6. Access is granted



IAM user sign in ⓘ

Account ID or alias (Don't have?)

615167489817

☑ Remember this account

IAM username

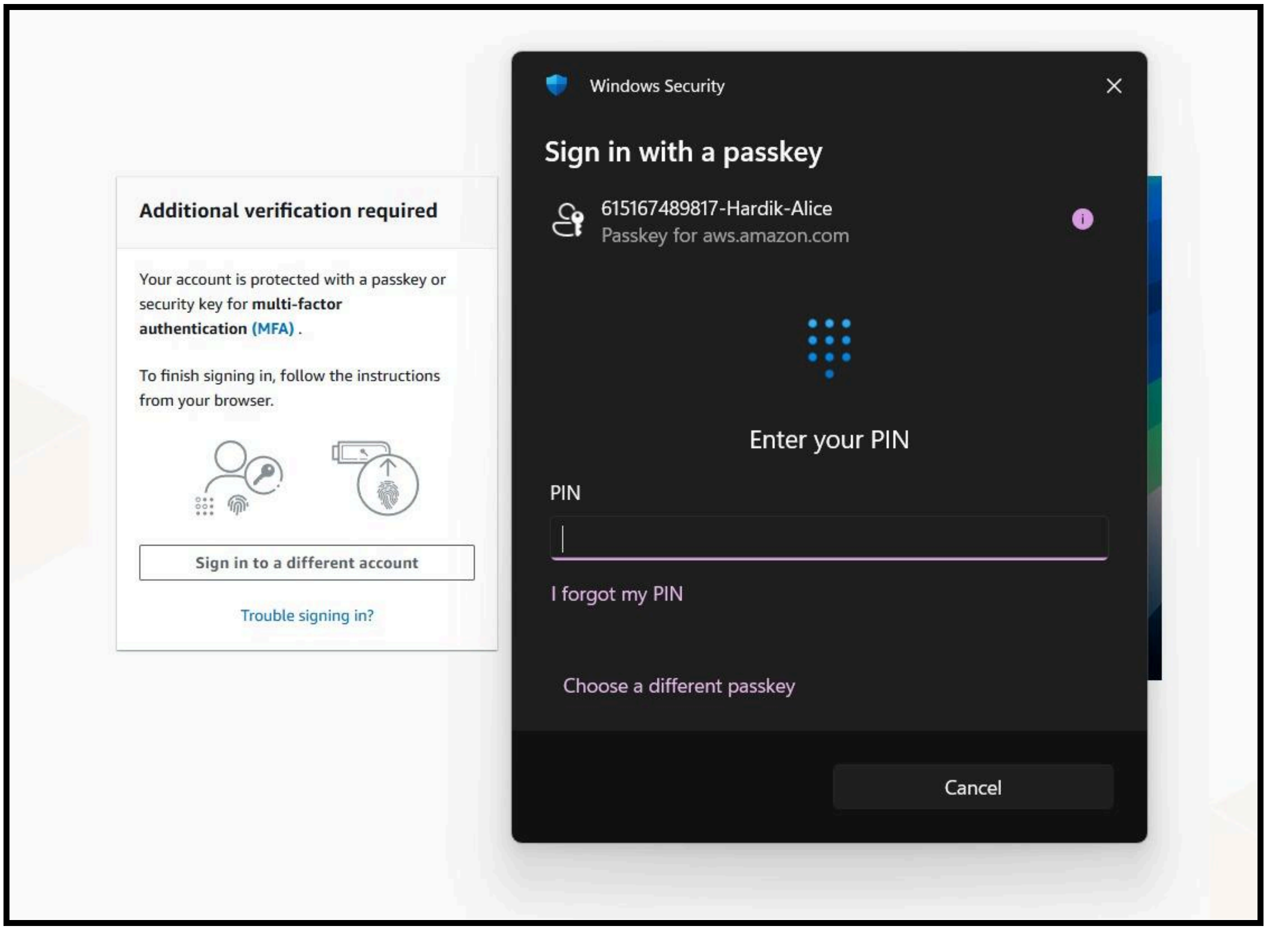
Alice

Password

Welcome@1234

☑ Show Password

By continuing, you agree to AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice. This site uses essential cookies. See our Cookie Notice for more information.



Additional verification required

Your account is protected with a passkey or security key for **multi-factor authentication** (MFA) .

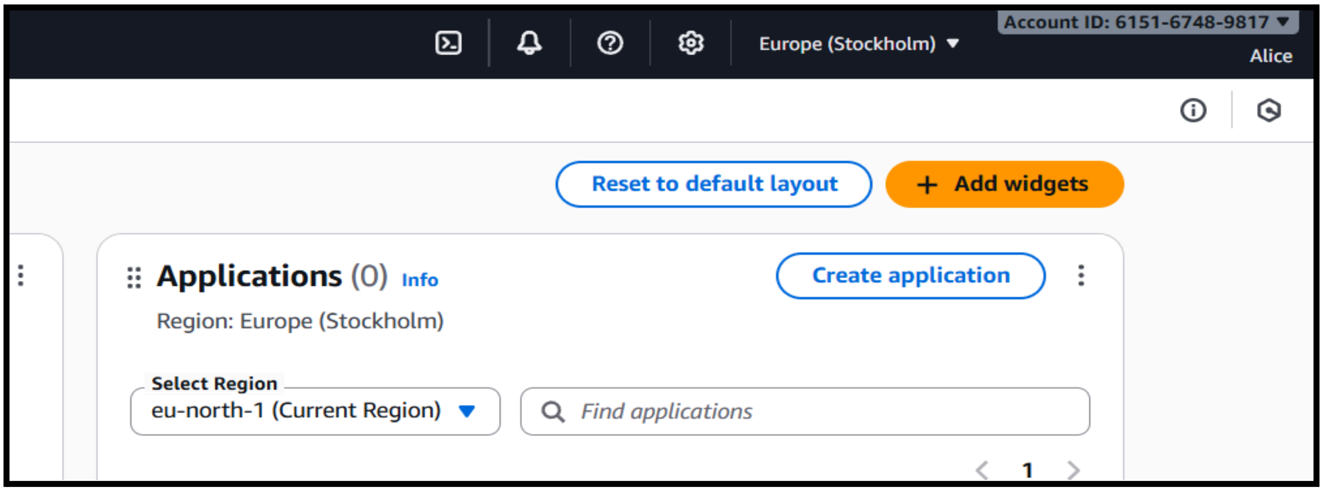
To finish signing in, follow the instructions from your browser.

Sign in with a passkey

615167489817-Hardik-Alice
Passkey for aws.amazon.com

Enter your PIN

PIN



Account ID: 6151-6748-9817 ▼
Alice

Applications (0) Info

Region: Europe (Stockholm)

Select Region
eu-north-1 (Current Region) ▼

Security Analysis

Scenario	Result
Login without MFA	Access denied
Login with passkey MFA	Access granted

Security Benefits

- Phishing-resistant authentication
- No OTP interception risk
- Hardware-backed credential storage
- Compliance with AWS security best practices

Conclusion

This implementation successfully enforces secure user authentication in AWS using IAM with passkey-based MFA.

By combining IAM authentication, FIDO2 passkeys, and deny-based policies, the system ensures strong access control and improved cloud security.