AWS IAM Hands-on Implementation and Security Best Practices

Users
Policies
Roles

Credit((Guo, 2022))

Prepared By: Kripesh Acharya

01/03/2025

Abstract

AWS Identity and Access Management (IAM) is a crucial component in securing cloud environments by managing user access and permissions. This report documents my hands-on experience in configuring AWS IAM, covering key aspects such as user management, security policies, access controls, and best practices. The objective of this report is to demonstrate practical knowledge of IAM configurations that enhance security and streamline access management in AWS environments. Screenshots have been included to provide a visual representation of the steps performed.

Table of Contents

Introduction

AWS Identity and Access Management (IAM)

AWS Identity and Access Management (IAM) provides granular access control across all AWS services and resources. It enables users to define who can access specific services and resources, and under what conditions.

1.1 Authentication & Authorization

IAM operates with two key security functions:

- **Authentication:** Validates the identity of a user by verifying credentials such as usernames and passwords. Advanced authentication mechanisms like Multi-Factor Authentication (MFA) enhance security by requiring an additional verification step, such as a one-time code sent to a user's mobile device.
- **Authorization:** Determines what authenticated user is permitted to access. Authorization restricts access to applications, data sets, and AWS services based on defined permissions.

1.2 IAM Identities: Users, Groups, Roles

IAM defines different entities to manage permissions and access control:

Root User, IAM Users, and Groups

- **Root User:** The AWS root user is created when an AWS account is registered. It has unrestricted access to all AWS resources and services and is authenticated using the email and password used during account creation.
- **IAM Users:** These are individual entities created within an AWS account. IAM users can log in to the AWS Management Console or interact with AWS services via the API or CLI using long-term credentials.
- **User Groups:** A user group is a collection of IAM users with shared permissions. Groups simplify permission management by applying policies to multiple users at once.

IAM Roles

An IAM role is similar to a user but does not have long-term credentials. Instead, when a role is assumed, it grants temporary security credentials for a session. Roles can be assumed by IAM users, AWS services, or applications that require specific permissions without requiring permanent access credentials.

1.3 IAM Policies

IAM policies define permissions and control access within AWS.

- **Policies:** JSON-based documents that specify what actions are allowed or denied for a user, group, or role.
- **Permissions:** Policies determine whether a request to access a resource is approved or denied.
- **Resource-Based Policies:** These policies attach directly to AWS resources and control access at the resource level.

By leveraging IAM policies, organizations can ensure secure, controlled access to AWS resources, minimizing security risks.

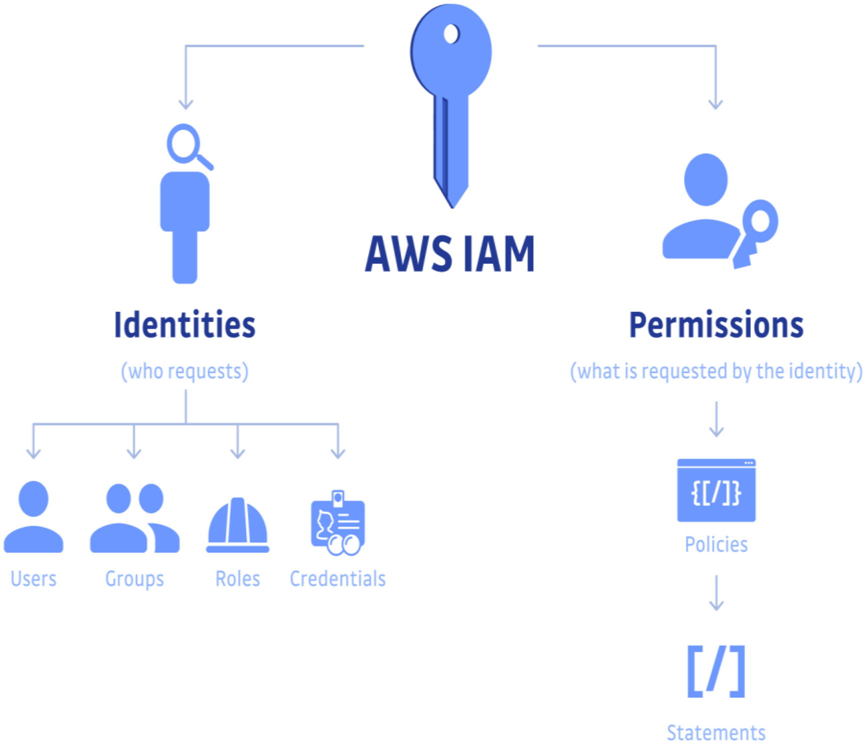
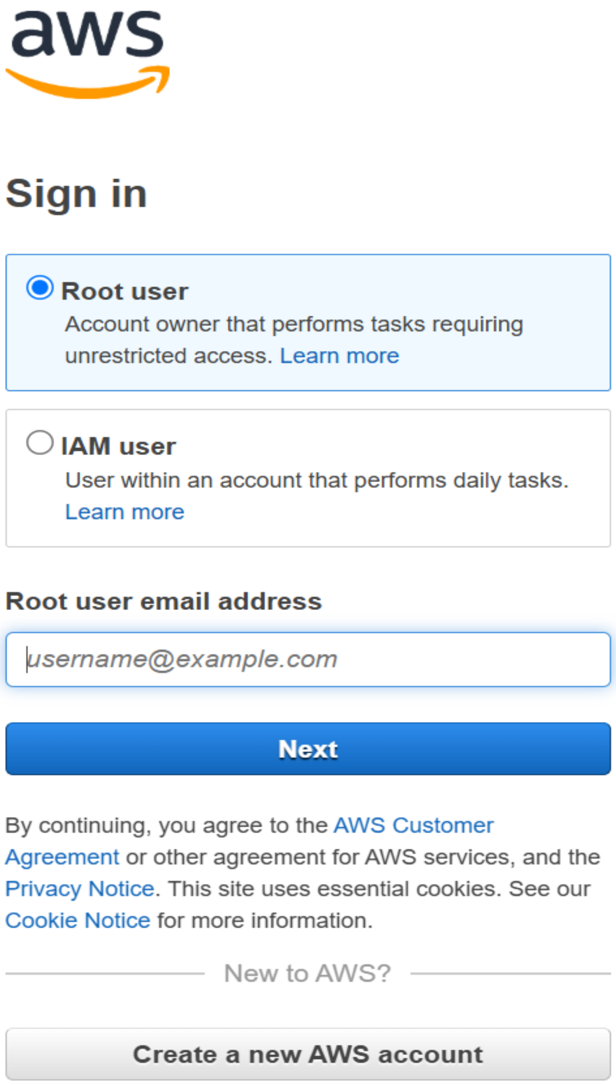


Fig: AWS IAM (Guo, 2022).

AWS IAM enables administrators to manage users, groups, roles, and policies securely. This report outlines my hands-on experience in implementing IAM security best practices, managing permissions, and ensuring a secure cloud environment. The subsequent sections detail the steps taken and the significance of each configuration.

2. Creating Admin Root User

To establish an AWS account, the root user is created with full administrative privileges. However, it is recommended to limit the use of the root account for security reasons.



Sign in

● Root user
Account owner that performs tasks requiring unrestricted access. Learn more

○ IAM user
User within an account that performs daily tasks. Learn more

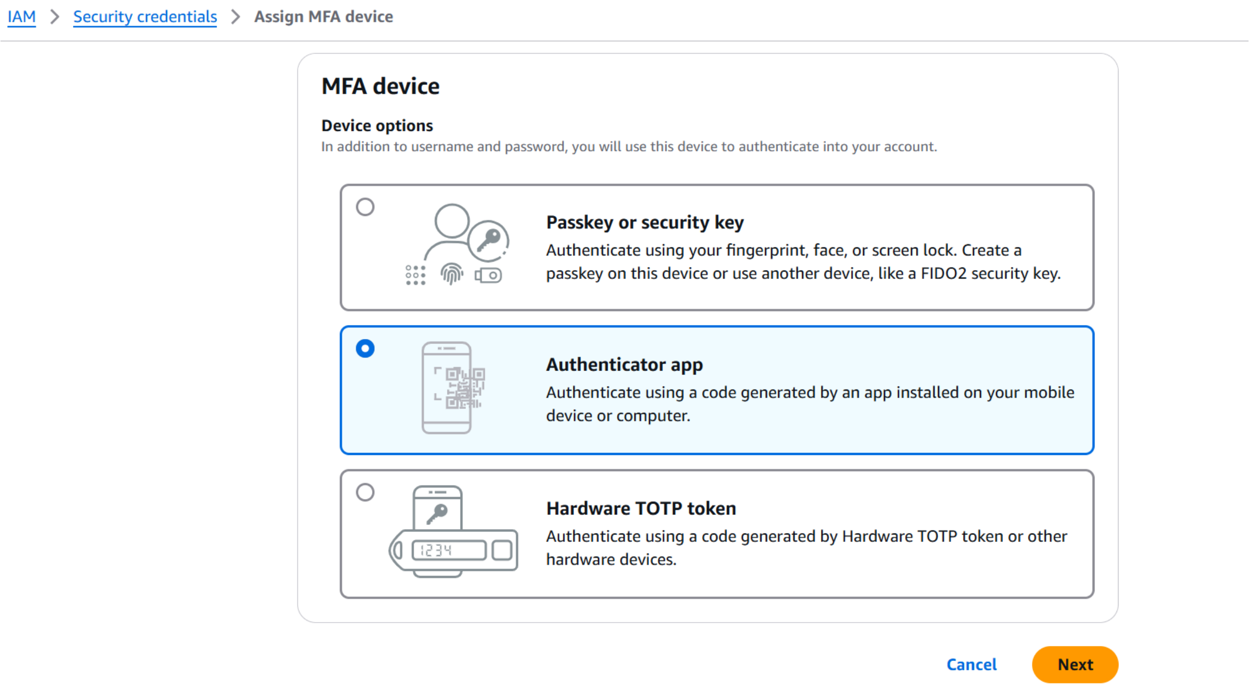
Root user email address

username@example.com

By continuing, you agree to the 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.

3. Setting Up Multi-Factor Authentication (MFA)

Multi-Factor Authentication (MFA) enhances security by requiring an additional verification step, such as a one-time password (OTP) from a mobile authenticator app. Securing the root account with MFA is a critical best practice, as an unsecured root account can be exploited by malicious actors, potentially leading to full control over the AWS environment. Implementing MFA significantly reduces the risk of unauthorized access and strengthens overall account security.



MFA device

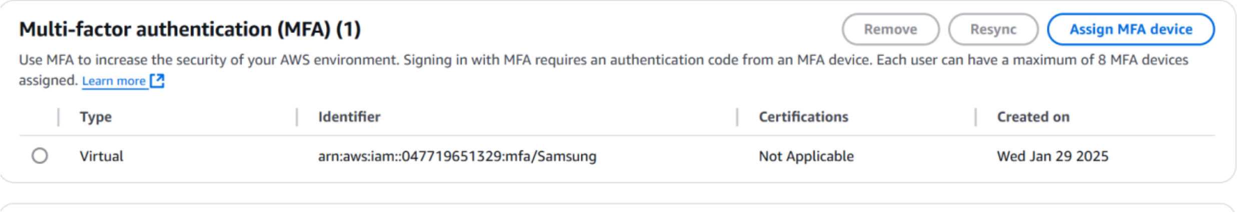
Device options

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.



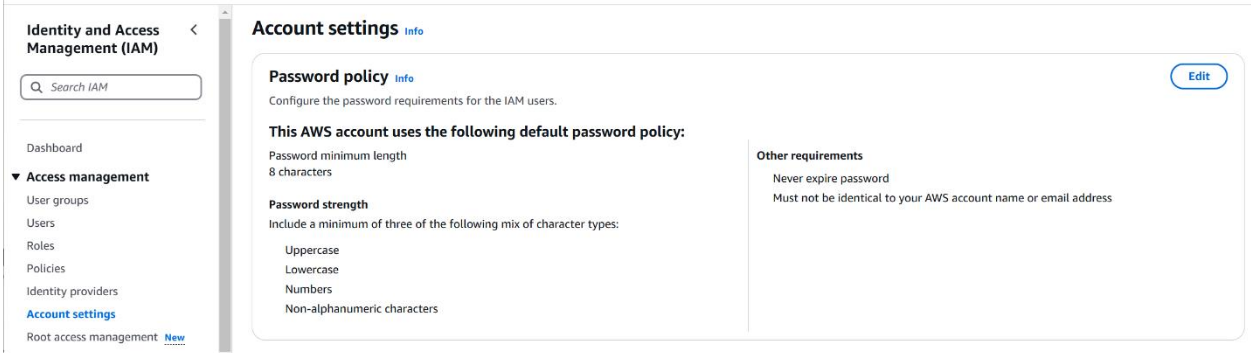
Multi-factor authentication (MFA) (1)

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::047719651329:mfa/Samsung	Not Applicable	Wed Jan 29 2025

4. Editing Custom Password Policy

A strong password policy enforces security best practices. I configured a custom policy to enforce complexity, expiration, and historical rules.



Account settings

Password policy

Configure the password requirements for the IAM users.

This AWS account uses the following default password policy:

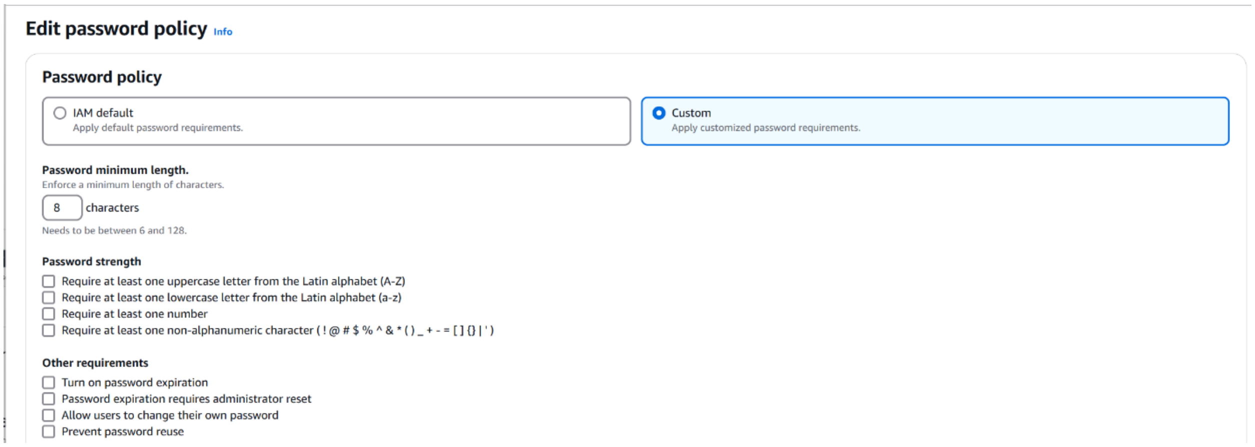
Password minimum length
8 characters

Password strength
Include a minimum of three of the following mix of character types:

- Uppercase
- Lowercase
- Numbers
- Non-alphanumeric characters

Other requirements

- Never expire password
- Must not be identical to your AWS account name or email address



Edit password policy

Password policy

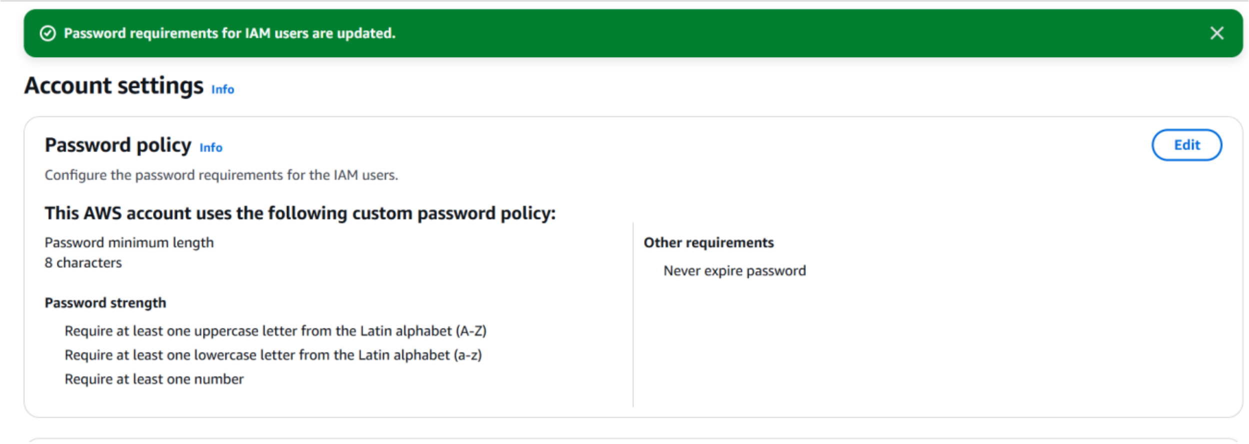
○ IAM default
Apply default password requirements.

● Custom
Apply customized password requirements.

Password minimum length.
Enforce a minimum length of characters.
8 characters
Needs to be between 6 and 128.

Password strength
□ Require at least one uppercase letter from the Latin alphabet (A-Z)
□ Require at least one lowercase letter from the Latin alphabet (a-z)
□ Require at least one number
□ Require at least one non-alphanumeric character (! @ # $ % ^ & * ( ) _ + - = [ ] { } | ')

Other requirements
□ Turn on password expiration
□ Password expiration requires administrator reset
□ Allow users to change their own password
□ Prevent password reuse



Password requirements for IAM users are updated.

Account settings

Password policy

Configure the password requirements for the IAM users.

This AWS account uses the following custom password policy:

Password minimum length
8 characters

Password strength

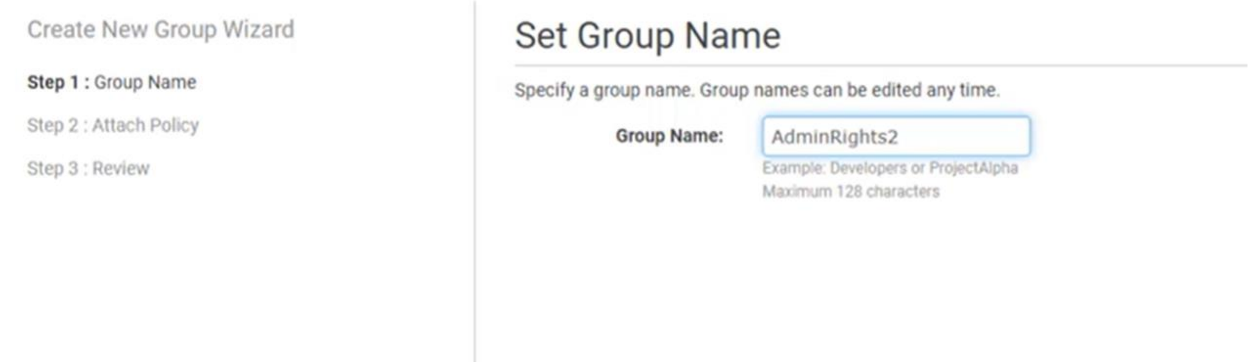
- Require at least one uppercase letter from the Latin alphabet (A-Z)
- Require at least one lowercase letter from the Latin alphabet (a-z)
- Require at least one number

Other requirements

- Never expire password

5. Creating a Group

IAM groups simplify permissions management by assigning policies at the group level instead of individual users.



Create New Group Wizard

Step 1 : Group Name

Step 2 : Attach Policy

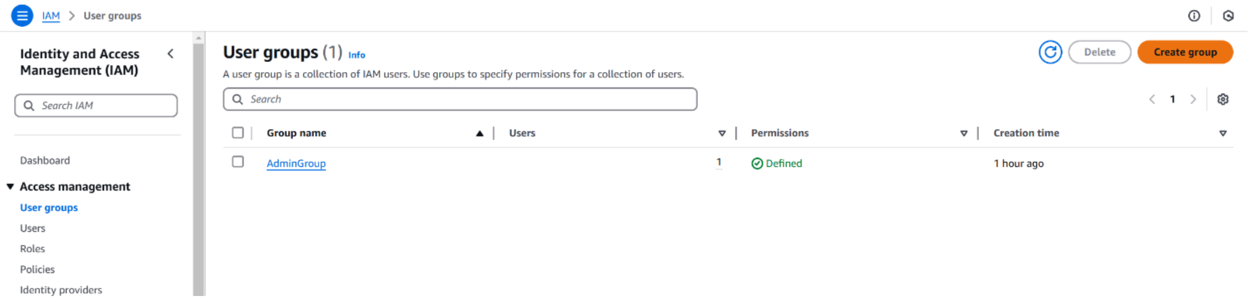
Step 3 : Review

Set Group Name

Specify a group name. Group names can be edited any time.

Group Name: AdminRights2

Example: Developers or ProjectAlpha
Maximum 128 characters



Identity and Access Management (IAM)

Search IAM

Dashboard

Access management
- User groups
- Users
- Roles
- Policies
- Identity providers

User groups (1) Info

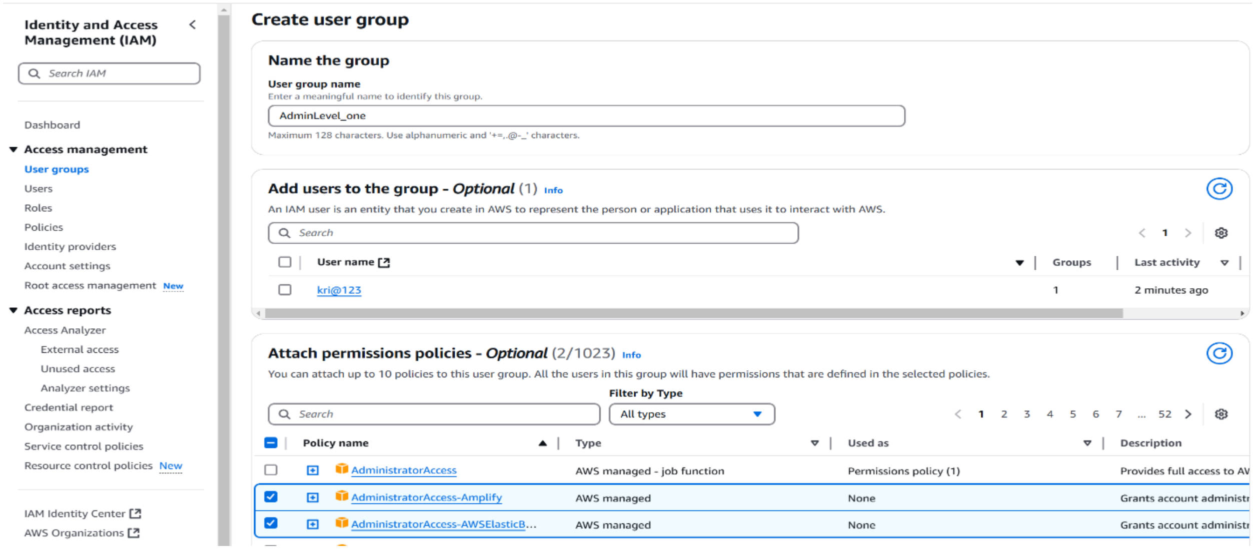
A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

Search

□	Group name	Users	Permissions	Creation time
□	AdminGroup	1	✓ Defined	1 hour ago

6. Attaching a Policy to the Group

Policies define the actions users or groups can perform. I assigned an AWS-managed policy to a group.



Identity and Access Management (IAM)

Search IAM

Dashboard

Access management
- User groups
- Users
- Roles
- Policies
- Identity providers
- Account settings
- Root access management New

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

IAM Identity Center
AWS Organizations

Create user group

Name the group

User group name
Enter a meaningful name to identify this group.
AdminLevel_one
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.

Add users to the group - *Optional* (1) Info

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

Search

□	User name	Groups	Last activity
□	kri@123	1	2 minutes ago

Attach permissions policies - *Optional* (2/1023) Info

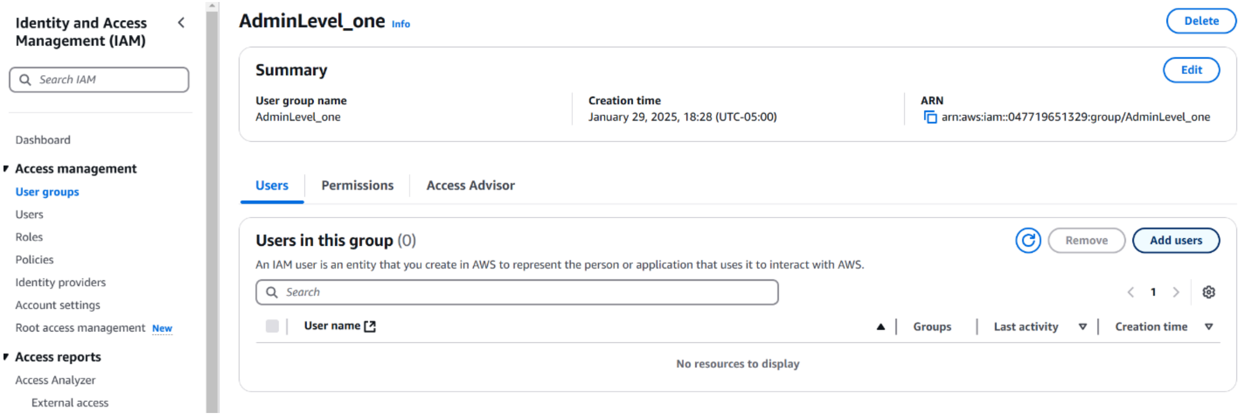
You can attach up to 10 policies to this user group. All the users in this group will have permissions that are defined in the selected policies.

Filter by Type: All types

	Policy name	Type	Used as	Description
□	AdministratorAccess	AWS managed - job function	Permissions policy (1)	Provides full access to AV
☑	AdministratorAccess-Amplify	AWS managed	None	Grants account administr
☑	AdministratorAccess-AWSElasticB...	AWS managed	None	Grants account administr

7. Adding a User to a Group

Users inherit permissions from the groups they belong to, ensuring streamlined access management.



Identity and Access Management (IAM)

Search IAM

- Dashboard
- Access management
  - User groups
  - Users
  - Roles
  - Policies
  - Identity providers
  - Account settings
  - Root access management New
- Access reports
  - Access Analyzer
    - External access

AdminLevel_one Info

Delete

Summary

Edit

User group name	Creation time	ARN
AdminLevel_one	January 29, 2025, 18:28 (UTC-05:00)	arn:aws:iam::047719651329:group/AdminLevel_one

Users | Permissions | Access Advisor

Users in this group (0)

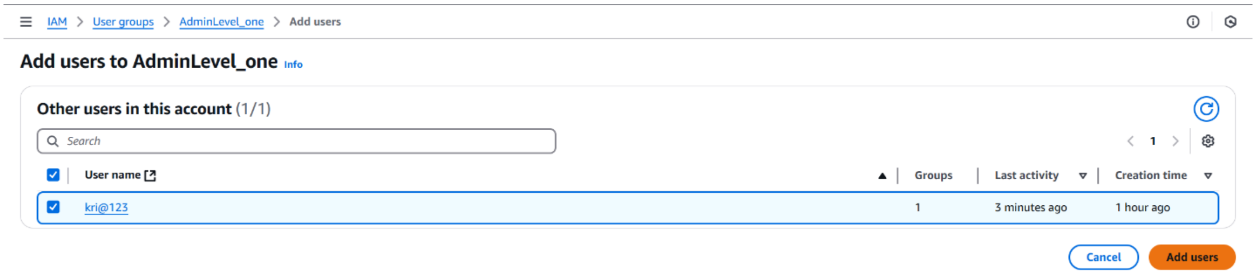
Remove | Add users

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

Search

□	User name	Groups	Last activity	Creation time

No resources to display



IAM > User groups > AdminLevel_one > Add users

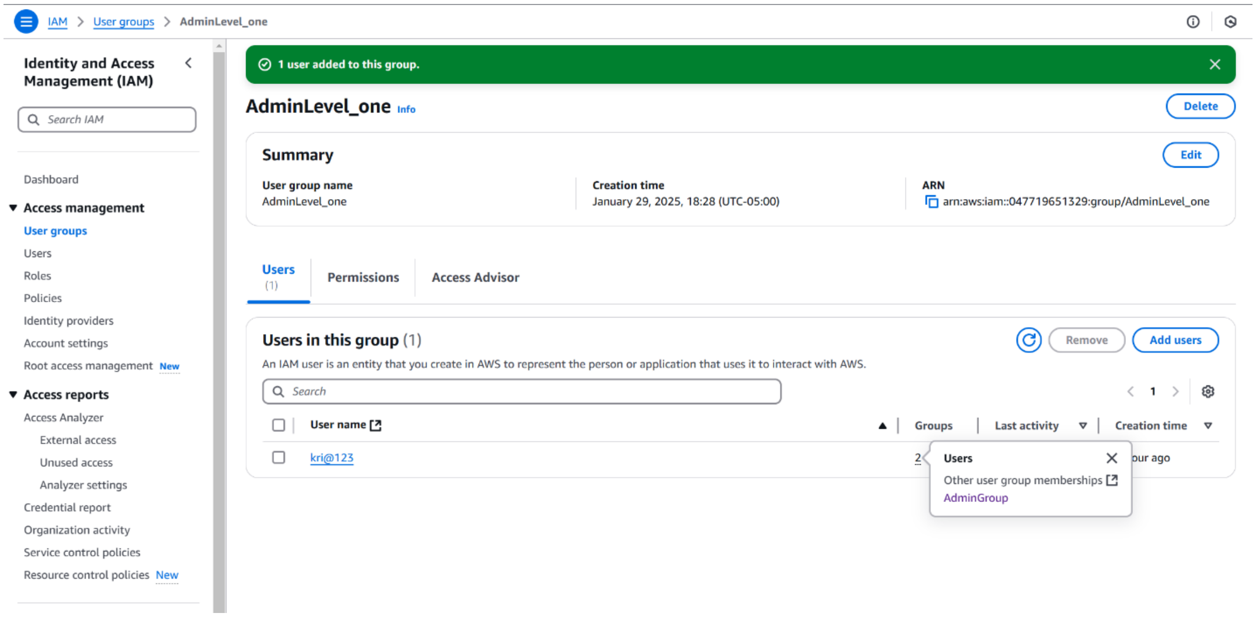
Add users to AdminLevel_one Info

Other users in this account (1/1)

Search

☑	User name	Groups	Last activity	Creation time
☑	kri@123	1	3 minutes ago	1 hour ago

Cancel | Add users



IAM > User groups > AdminLevel_one

1 user added to this group.

Identity and Access Management (IAM)

Search IAM

- Dashboard
- Access management
  - User groups
  - Users
  - Roles
  - Policies
  - Identity providers
  - Account settings
  - Root access management New
- Access reports
  - Access Analyzer
    - External access
    - Unused access
    - Analyzer settings
  - Credential report
  - Organization activity
  - Service control policies
  - Resource control policies New

AdminLevel_one Info

Delete

Summary

Edit

User group name	Creation time	ARN
AdminLevel_one	January 29, 2025, 18:28 (UTC-05:00)	arn:aws:iam::047719651329:group/AdminLevel_one

Users (1) | Permissions | Access Advisor

Users in this group (1)

Remove | Add users

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

Search

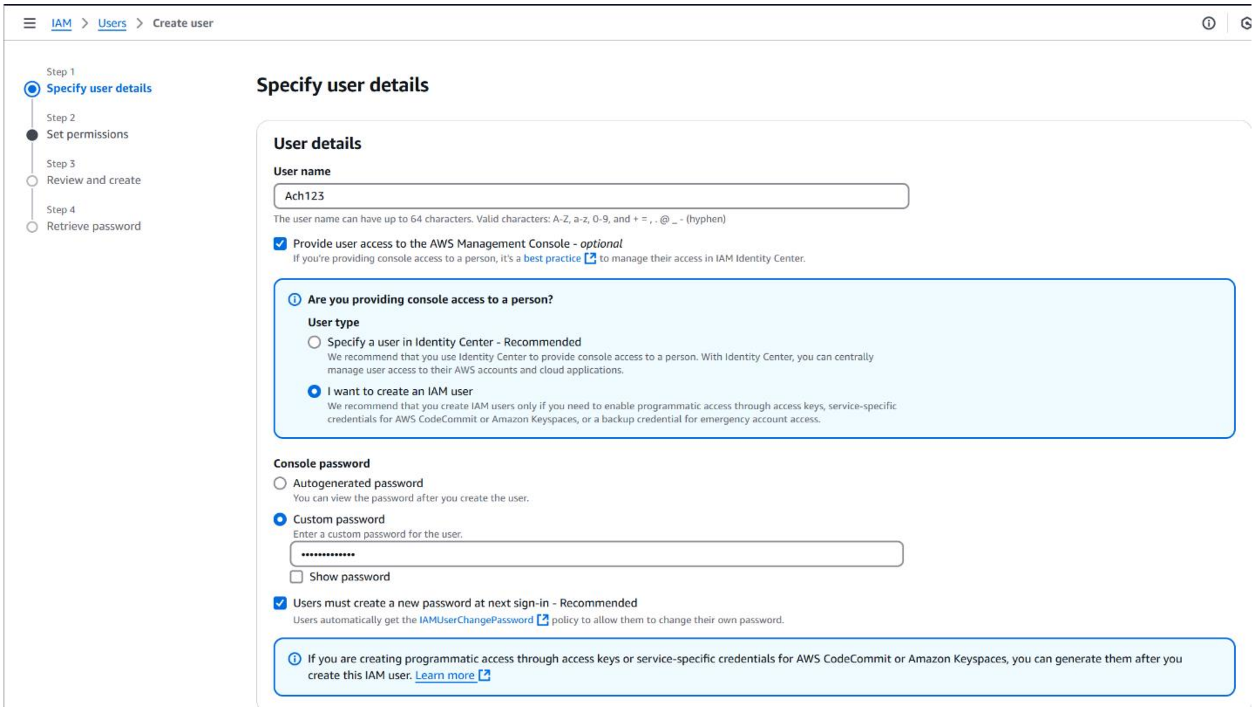
□	User name	Groups	Last activity	Creation time
□	kri@123	2		our ago

Users

Other user group memberships

AdminGroup

8. Creating Another User



Step 1
● Specify user details

Step 2
● Set permissions

Step 3
○ Review and create

Step 4
○ Retrieve password

Specify user details

User details

User name

Ach123

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.

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.
••••••••••••••

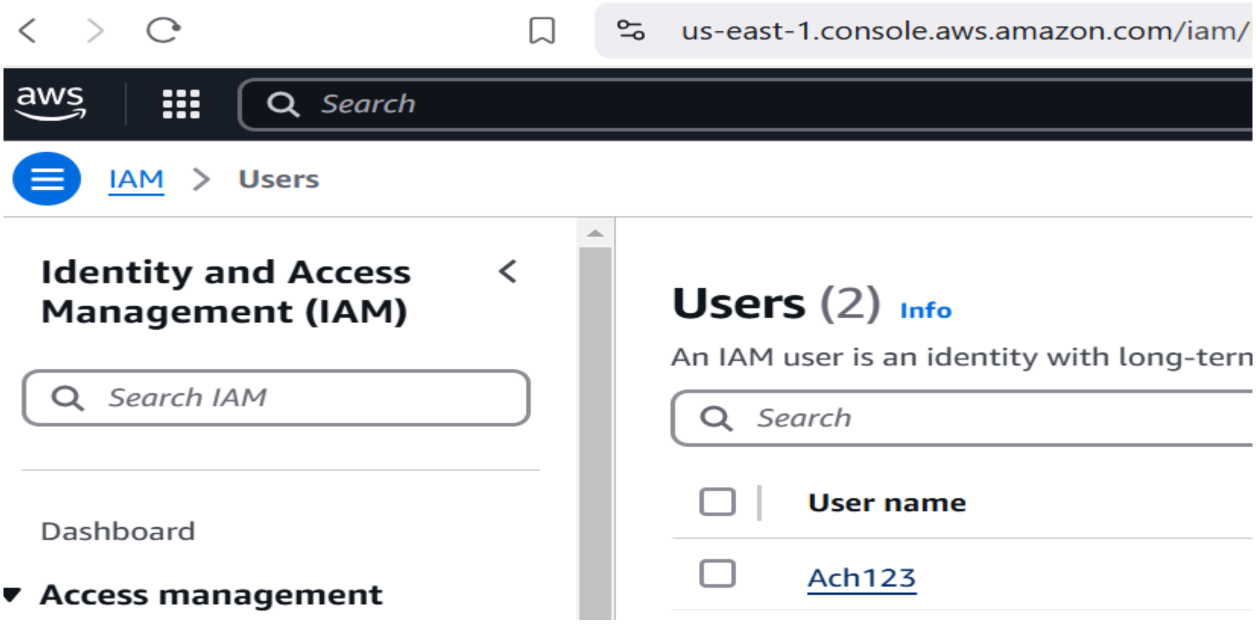
□ Show password

☑ 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

9. Attaching inline Policy to Specific Users

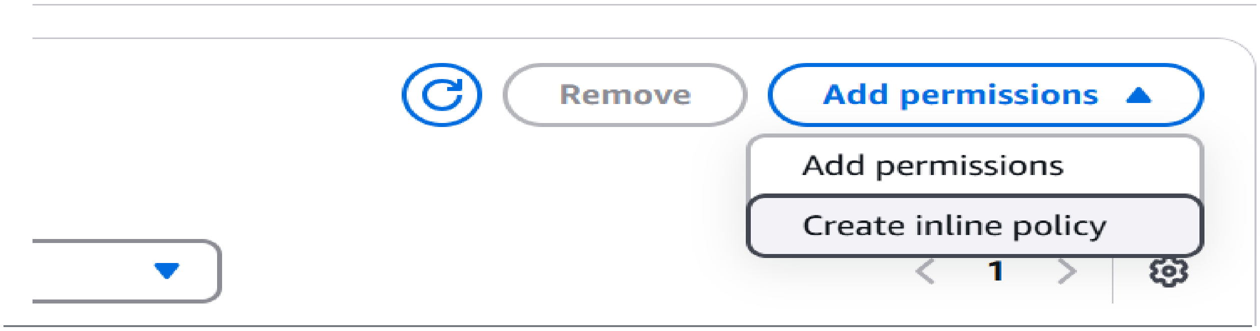
When additional permissions are needed beyond the group policies, they can be attached to individual users.

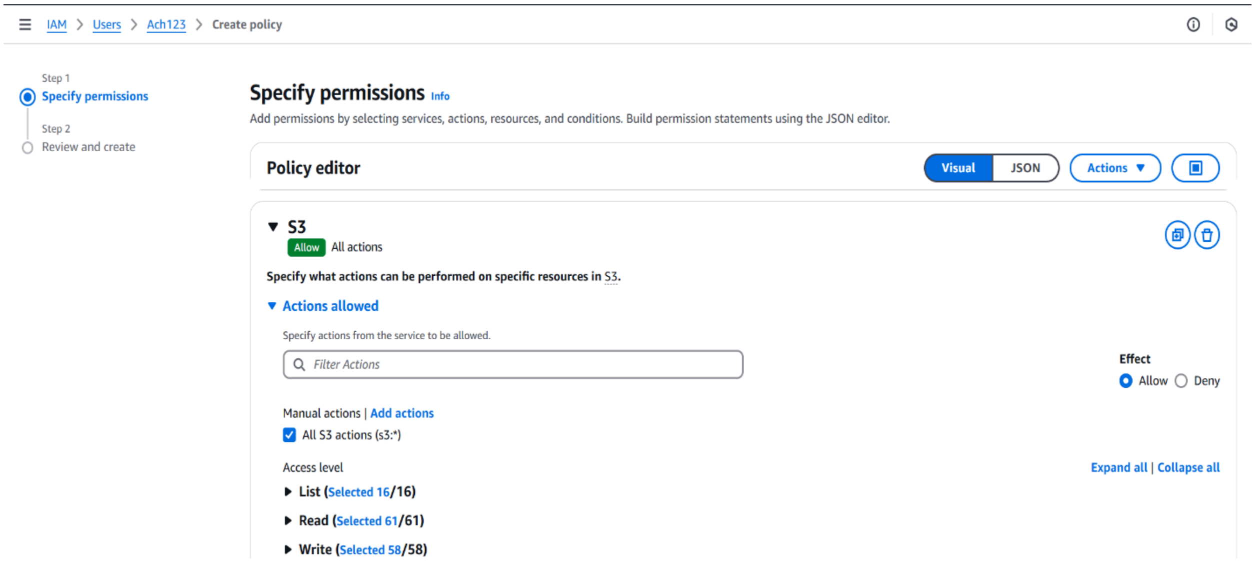

us-east-1.console.aws.amazon.com/iam/
aws
Search
IAM > Users
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management

Users (2) Info

An IAM user is an identity with long-tern

□	User name
□	Ach123





IAM > Users > Ach123 > Create policy

Step 1
Specify permissions

Step 2
Review and create

Specify permissions Info

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

Policy editor — Visual | JSON — Actions ▼

▼ S3
Allow All actions

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

▼ Actions allowed

Specify actions from the service to be allowed.

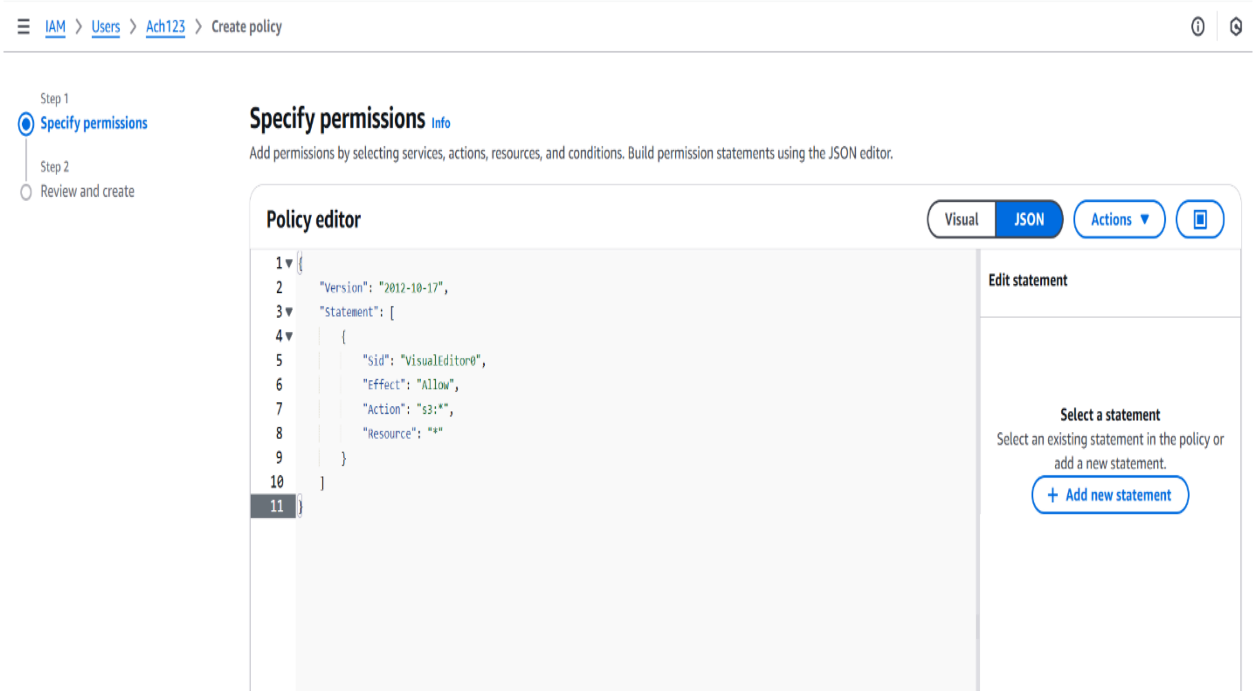
Filter Actions

Effect
● Allow ○ Deny

Manual actions | Add actions
☑ All S3 actions (s3:*)

Access level — Expand all | Collapse all

- ▶ List (Selected 16/16)
- ▶ Read (Selected 61/61)
- ▶ Write (Selected 58/58)



IAM > Users > Ach123 > Create policy

Step 1
Specify permissions

Step 2
Review and create

Specify permissions Info

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

Policy editor — Visual | JSON — Actions ▼

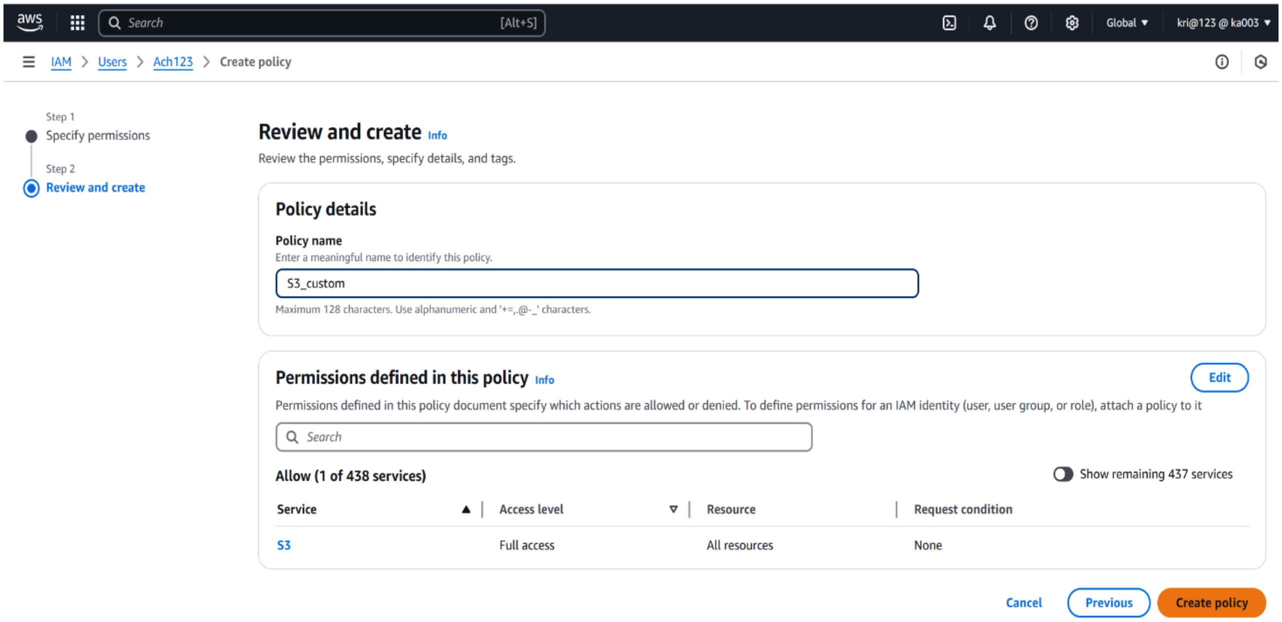
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": "*"
        }
    ]
}
```

Edit statement

Select a statement

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

+ Add new statement



Step 1
Specify permissions

Step 2
Review and create

Review and create Info

Review the permissions, specify details, and tags.

Policy details

Policy name
Enter a meaningful name to identify this policy.

S3_custom

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

Permissions defined in this policy Info

Edit

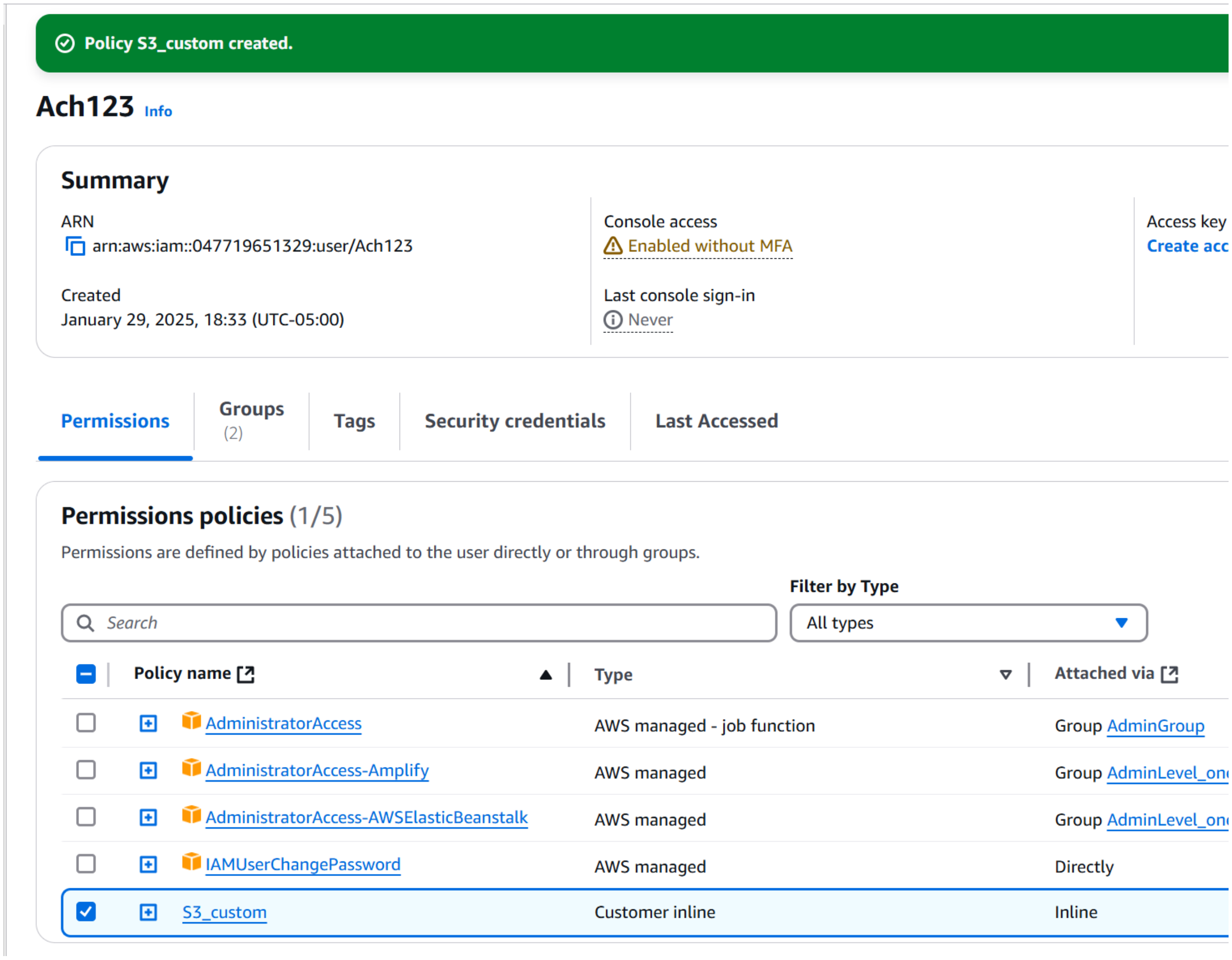
Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Search

Allow (1 of 438 services)

Show remaining 437 services

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



Policy S3_custom created.

Ach123 Info

Summary

ARN
arn:aws:iam::047719651329:user/Ach123

Created
January 29, 2025, 18:33 (UTC-05:00)

Console access
Enabled without MFA

Last console sign-in
Never

Access key
Create acc

Permissions policies (1/5)

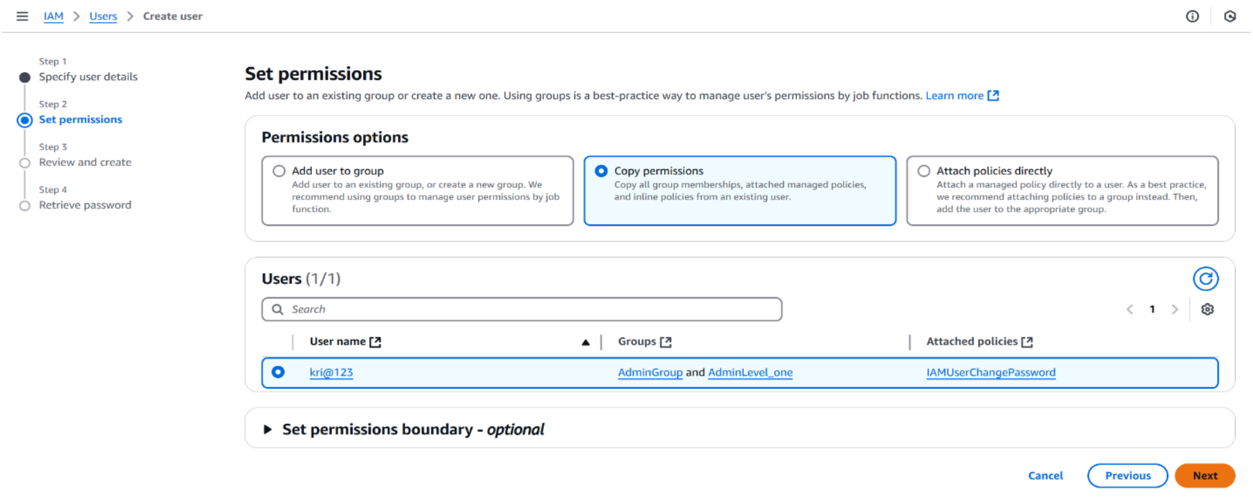
Permissions are defined by policies attached to the user directly or through groups.

Search

Filter by Type
All types

	Policy name ▲	Type ▽	Attached via
□	AdministratorAccess	AWS managed - job function	Group AdminGroup
□	AdministratorAccess-Amplify	AWS managed	Group AdminLevel_on
□	AdministratorAccess-AWSElasticBeanstalk	AWS managed	Group AdminLevel_on
□	IAMUserChangePassword	AWS managed	Directly
☑	S3_custom	Customer inline	Inline

10. Adding Same user to Multiple Groups



IAM > Users > Create user

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. 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.

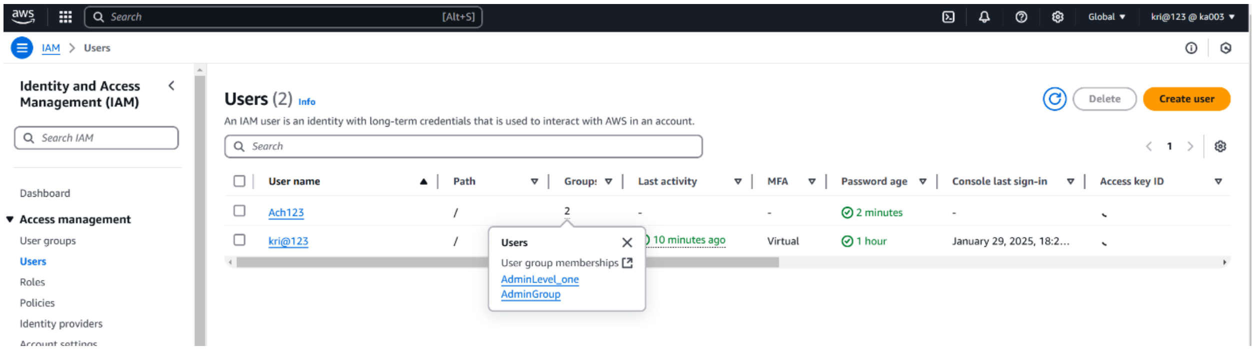
○ Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Users (1/1)

Search

	User name	Groups	Attached policies
●	kri@123	AdminGroup and AdminLevel_one	IAMUserChangePassword

▶ **Set permissions boundary - *optional***



Identity and Access Management (IAM)

Search IAM

Dashboard
▼ Access management
User groups
Users
Roles
Policies
Identity providers
Account settings

Users (2) Info

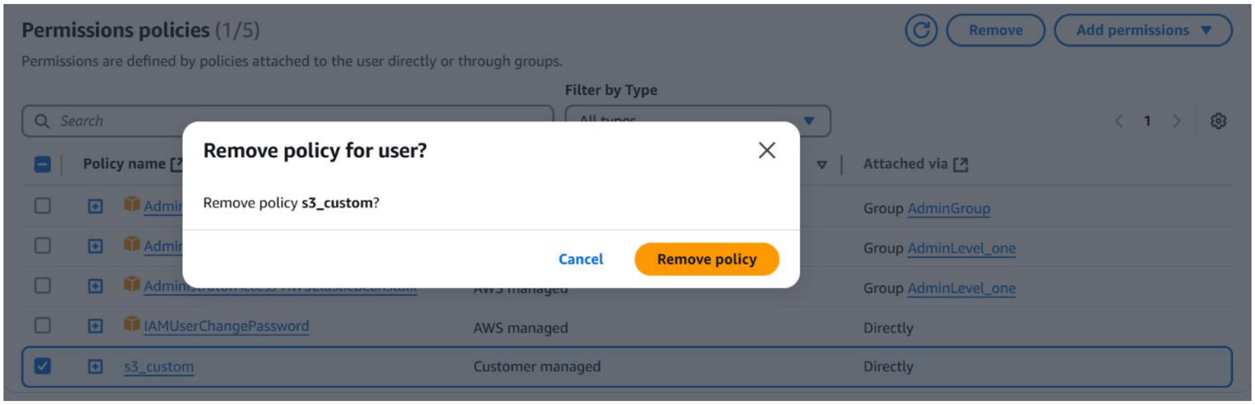
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

□	User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in	Access key ID
□	Ach123	/	2	-	-	✓ 2 minutes	-	
□	kri@123	/		10 minutes ago	Virtual	✓ 1 hour	January 29, 2025, 18:2...	

Users
User group memberships
AdminLevel_one
AdminGroup

11. Removing a Policy from a Specific Group

To enhance security, unnecessary policies should be removed from groups to follow the principle of least privilege.



Permissions policies (1/5)

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

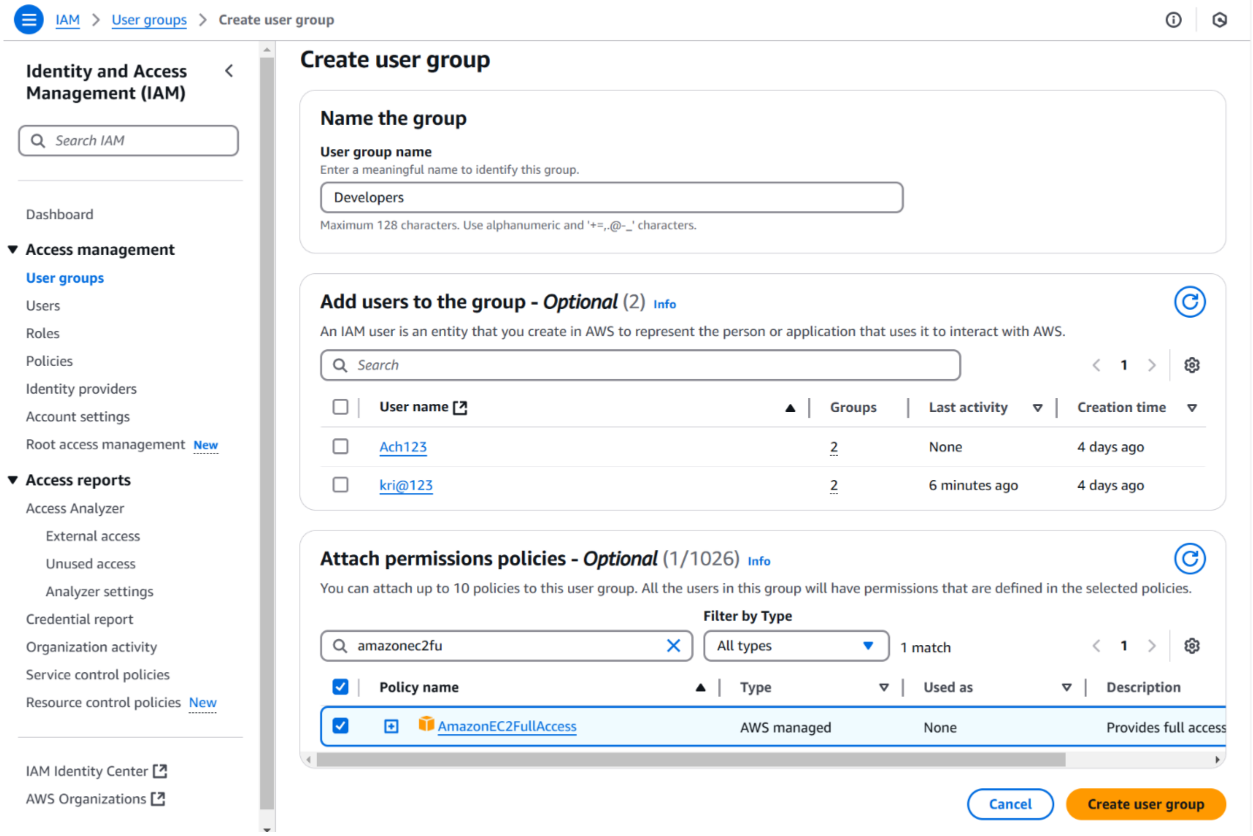
□	Policy name	Type	Attached via
□	Admir		Group AdminGroup
□	Admir		Group AdminLevel_one
□	Admin	AWS managed	Group AdminLevel_one
□	IAMUserChangePassword	AWS managed	Directly
☑	s3_custom	Customer managed	Directly

Remove policy for user?

Remove policy **s3_custom**?

11. Adding different New Users to different New Groups

User management is simplified by assigning users to different groups based on their roles and responsibilities.



Create user group

Name the group

User group name

Enter a meaningful name to identify this group.

Developers

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

Add users to the group - *Optional* (2) Info

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

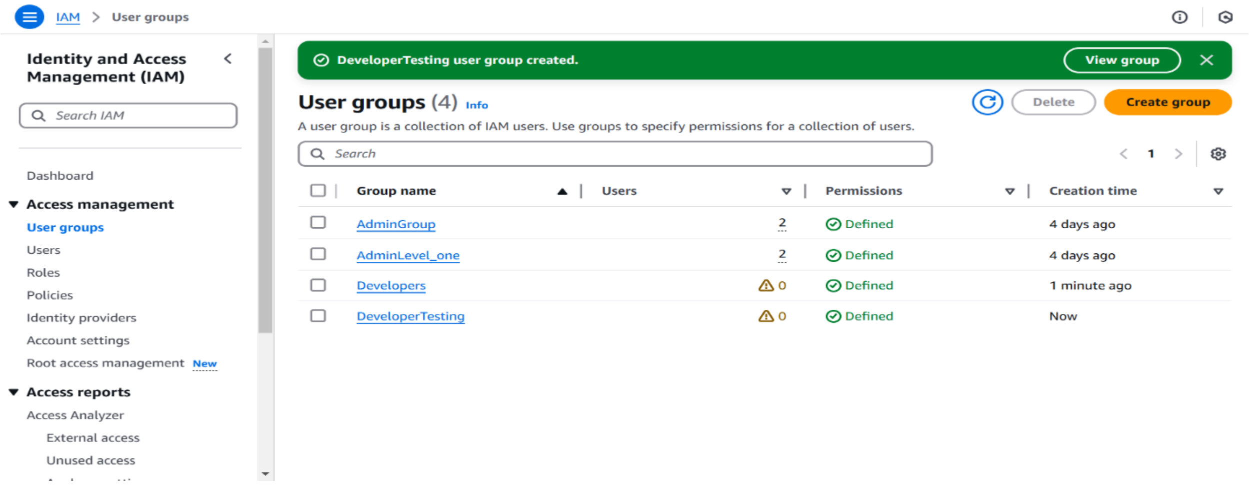
□	User name	Groups	Last activity	Creation time
□	Ach123	2	None	4 days ago
□	kri@123	2	6 minutes ago	4 days ago

Attach permissions policies - *Optional* (1/1026) Info

You can attach up to 10 policies to this user group. All the users in this group will have permissions that are defined in the selected policies.

Search: amazonec2fu — Filter by Type: All types — 1 match

☑	Policy name	Type	Used as	Description
☑	AmazonEC2FullAccess	AWS managed	None	Provides full access

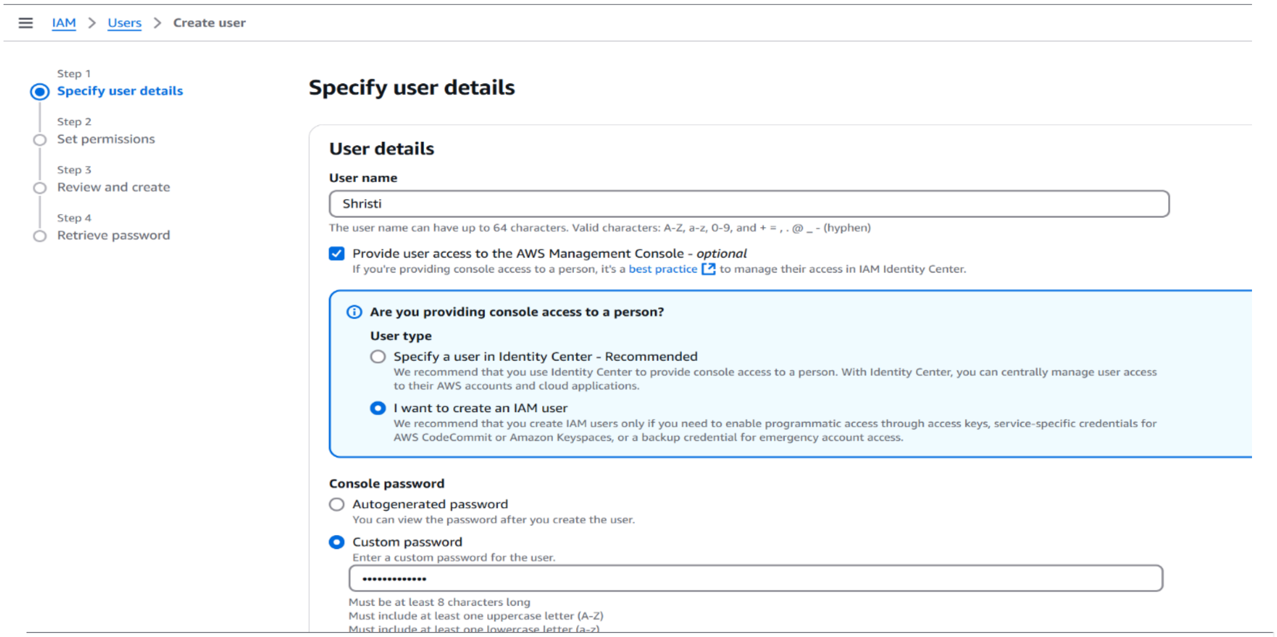


DeveloperTesting user group created.

User groups (4) 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
□	AdminGroup	2	✓ Defined	4 days ago
□	AdminLevel_one	2	✓ Defined	4 days ago
□	Developers	⚠ 0	✓ Defined	1 minute ago
□	DeveloperTesting	⚠ 0	✓ Defined	Now



IAM > Users > Create user

Step 1
● Specify user details

Step 2
○ Set permissions

Step 3
○ Review and create

Step 4
○ Retrieve password

Specify user details

User details

User name

Shristi

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.

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.

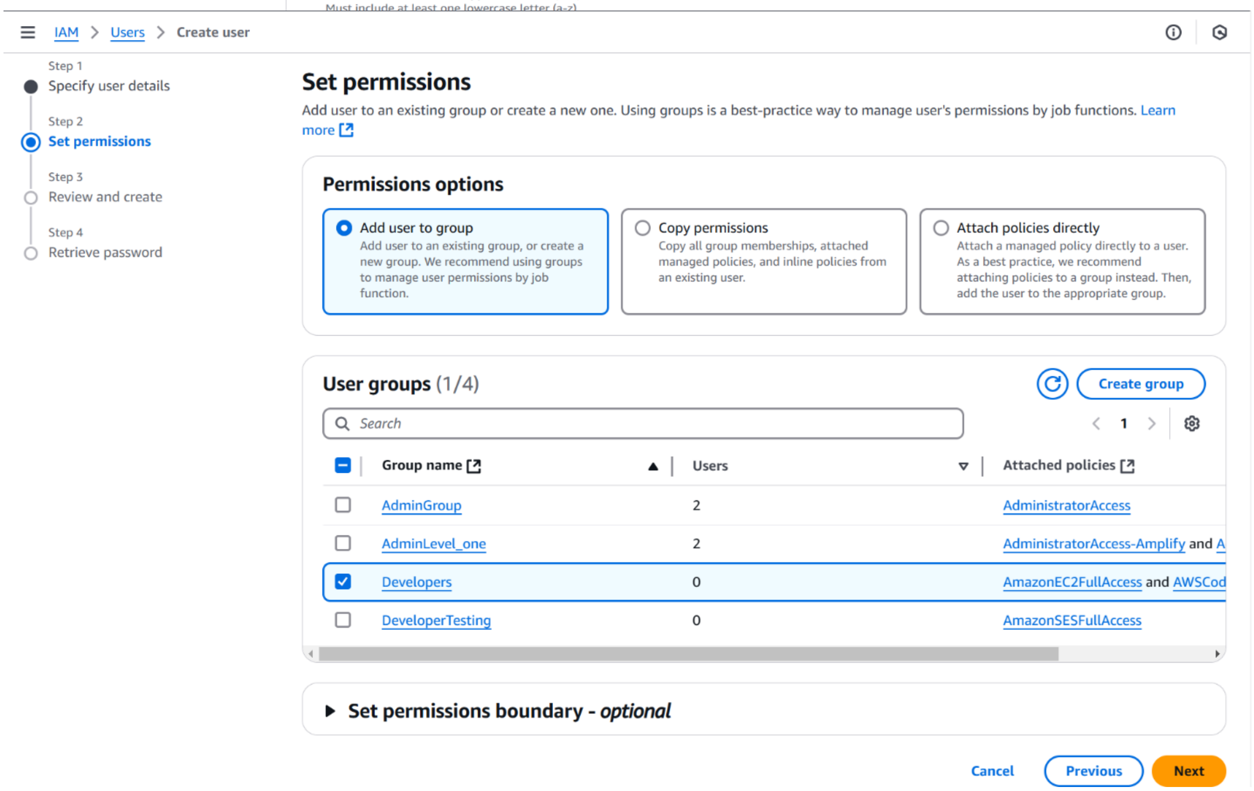
Console password

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

● Custom password
Enter a custom password for the user.

•••••••••••••

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)



IAM > Users > Create user

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. 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.

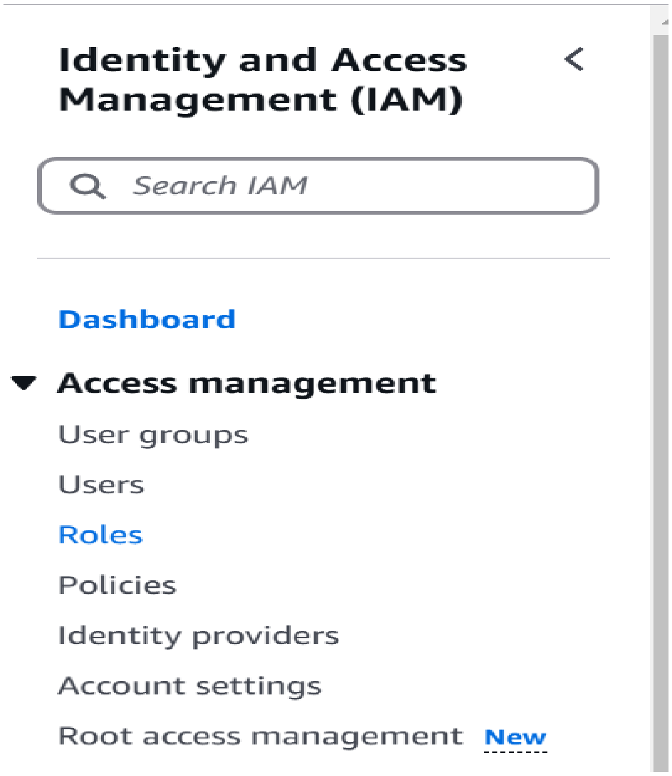
User groups (1/4)

▣	Group name ▲	Users	Attached policies
□	AdminGroup	2	AdministratorAccess
□	AdminLevel_one	2	AdministratorAccess-Amplify and A
☑	Developers	0	AmazonEC2FullAccess and AWSCod
□	DeveloperTesting	0	AmazonSESFullAccess

▸ Set permissions boundary - *optional*

12. Adding Roles

Roles grant permissions to AWS services and applications, eliminating the need for long-term credentials.



Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management

User groups

Users

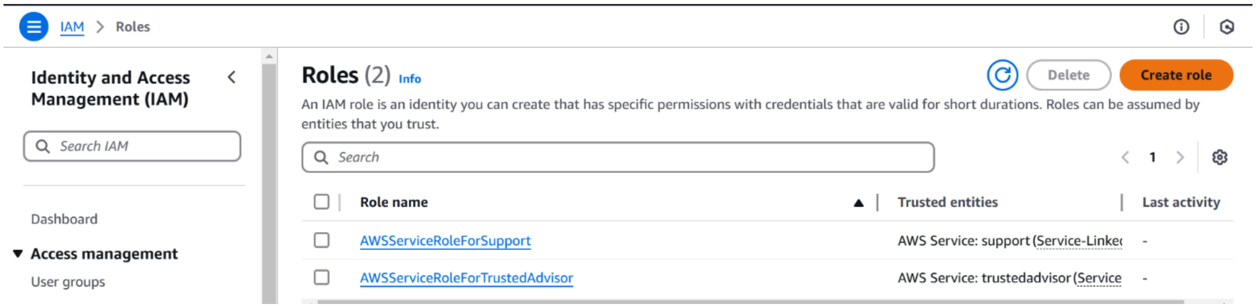
Roles

Policies

Identity providers

Account settings

Root access management New



IAM > Roles

Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management

User groups

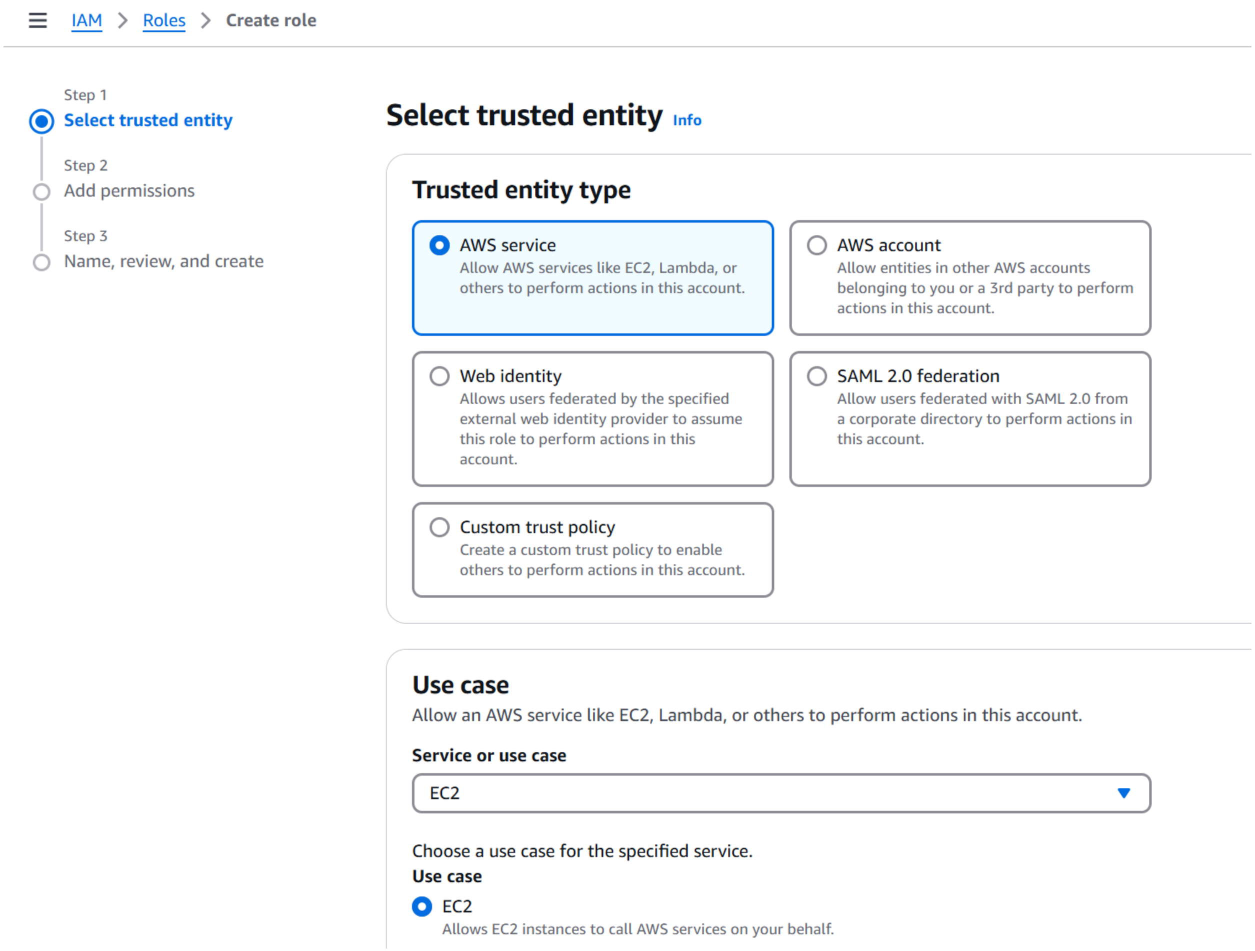
Roles (2) Info

Delete | Create role

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Search

□	Role name	Trusted entities	Last activity
□	AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-
□	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-



IAM > Roles > Create role

Step 1
● Select trusted entity

Step 2
○ Add permissions

Step 3
○ Name, review, and create

Select trusted entity Info

Trusted entity type

● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

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

Use case

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

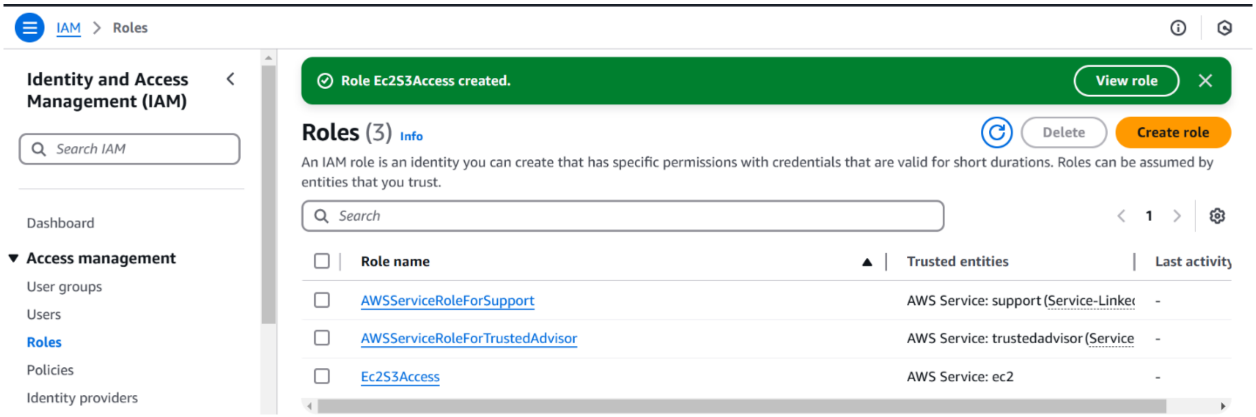
Service or use case

EC2 ▼

Choose a use case for the specified service.

Use case

● EC2
Allows EC2 instances to call AWS services on your behalf.



IAM > Roles

Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management
User groups
Users
Roles
Policies
Identity providers

Role Ec2S3Access created. View role ✕

Roles (3) Info   Delete   Create role

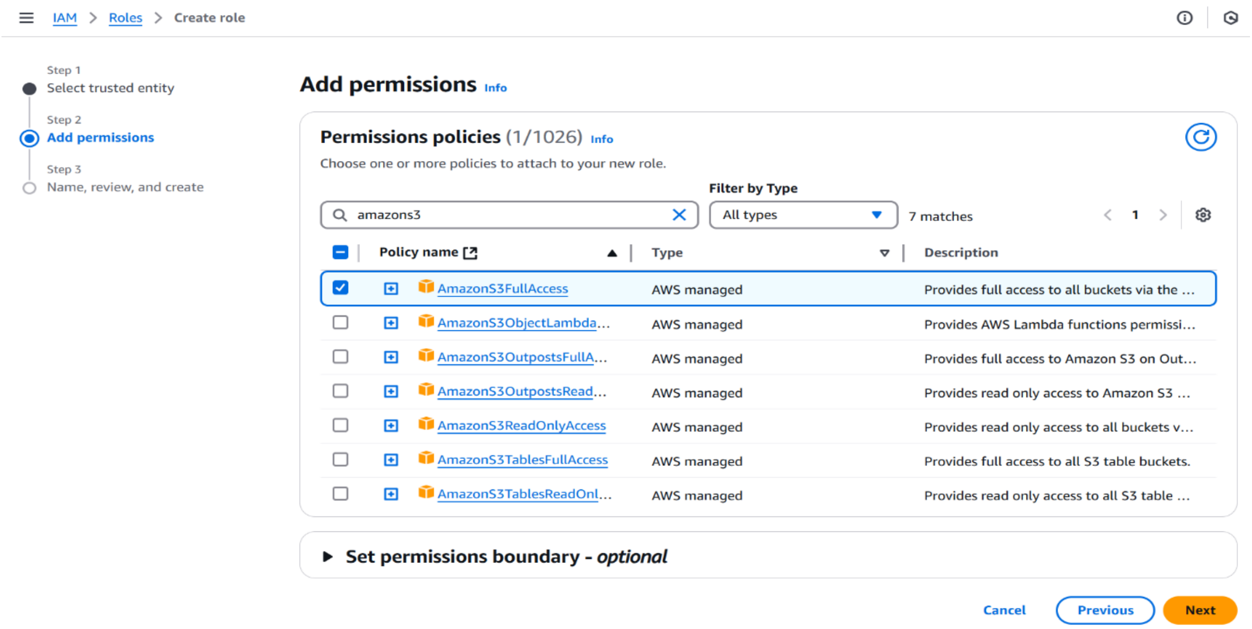
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Search   < 1 >

□	Role name ▲	Trusted entities	Last activity
□	AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-
□	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-
□	Ec2S3Access	AWS Service: ec2	-

13. Adding Resource-Based Policies into Roles

Resource-based policies define access permissions directly on AWS resources.



Add permissions

Permissions policies (1/1026)

Choose one or more policies to attach to your new role.

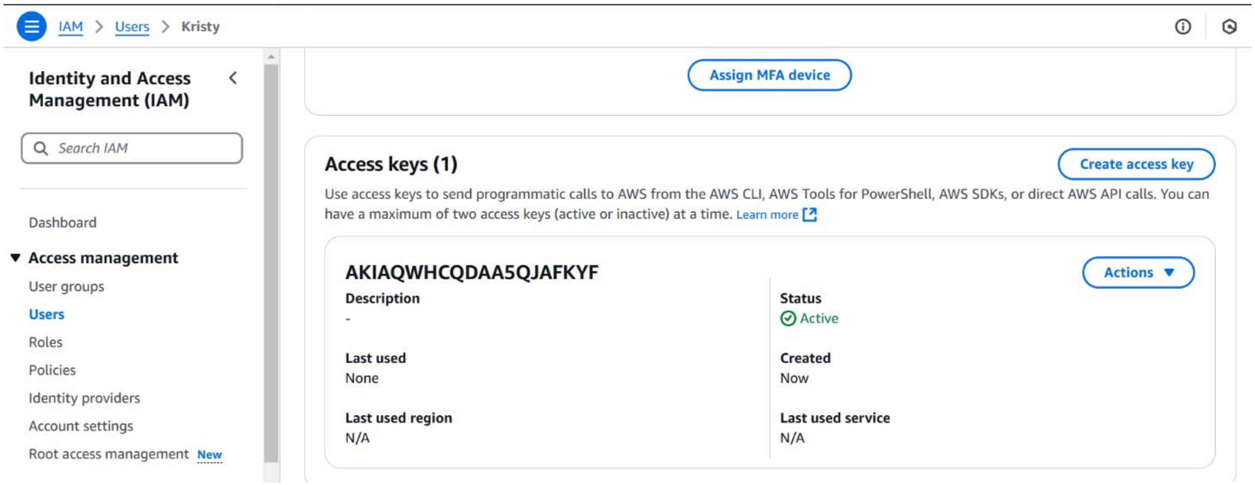
Search: amazons3

	Policy name	Type	Description
☑	AmazonS3FullAccess	AWS managed	Provides full access to all buckets via the …
□	AmazonS3ObjectLambda…	AWS managed	Provides AWS Lambda functions permissi…
□	AmazonS3OutpostsFullA…	AWS managed	Provides full access to Amazon S3 on Out…
□	AmazonS3OutpostsRead…	AWS managed	Provides read only access to Amazon S3 …
□	AmazonS3ReadOnlyAccess	AWS managed	Provides read only access to all buckets v…
□	AmazonS3TablesFullAccess	AWS managed	Provides full access to all S3 table buckets.
□	AmazonS3TablesReadOnl…	AWS managed	Provides read only access to all S3 table …

▶ Set permissions boundary - *optional*

14. Rotating Access Keys for Better Security

To minimize security risks, access keys should be rotated regularly to prevent unauthorized access.



Access keys (1)

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.

AKIAQWHCQDAA5QJAFKYF

Description: -

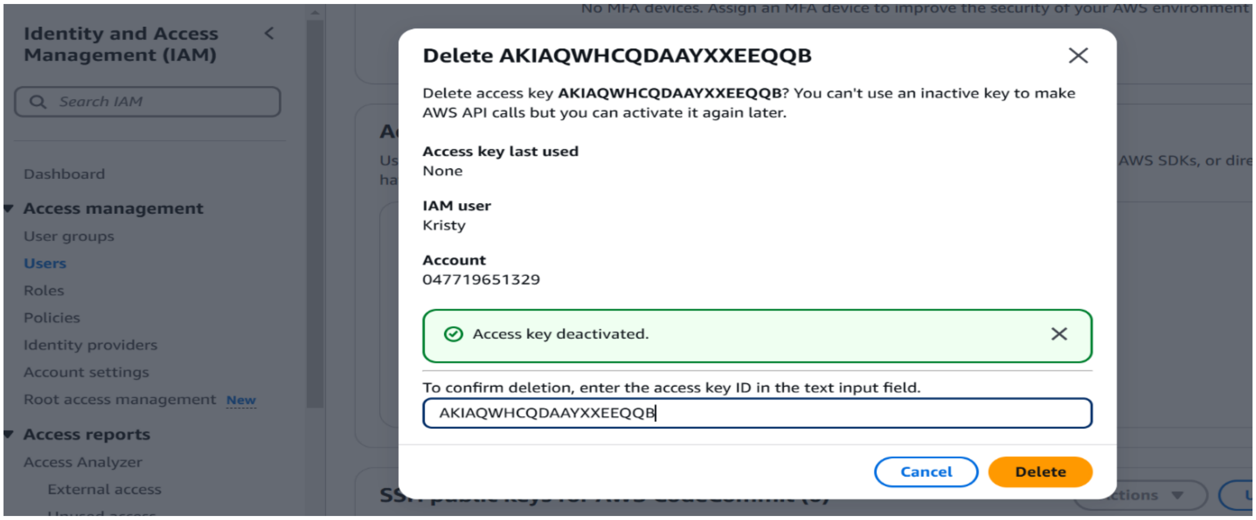
Status: Active

Last used: None

Created: Now

Last used region: N/A

Last used service: N/A



Delete AKIAQWHCQDAAYXXEEQQB

Delete access key **AKIAQWHCQDAAYXXEEQQB**? You can't use an inactive key to make AWS API calls but you can activate it again later.

Access key last used
None

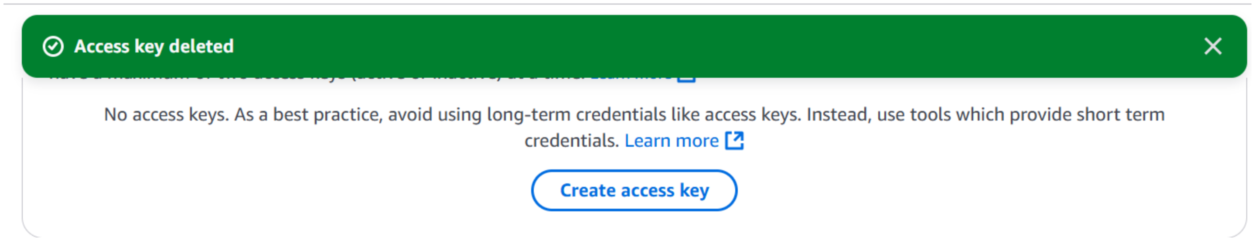
IAM user
Kristy

Account
047719651329

Access key deactivated.

To confirm deletion, enter the access key ID in the text input field.

AKIAQWHCQDAAYXXEEQQB

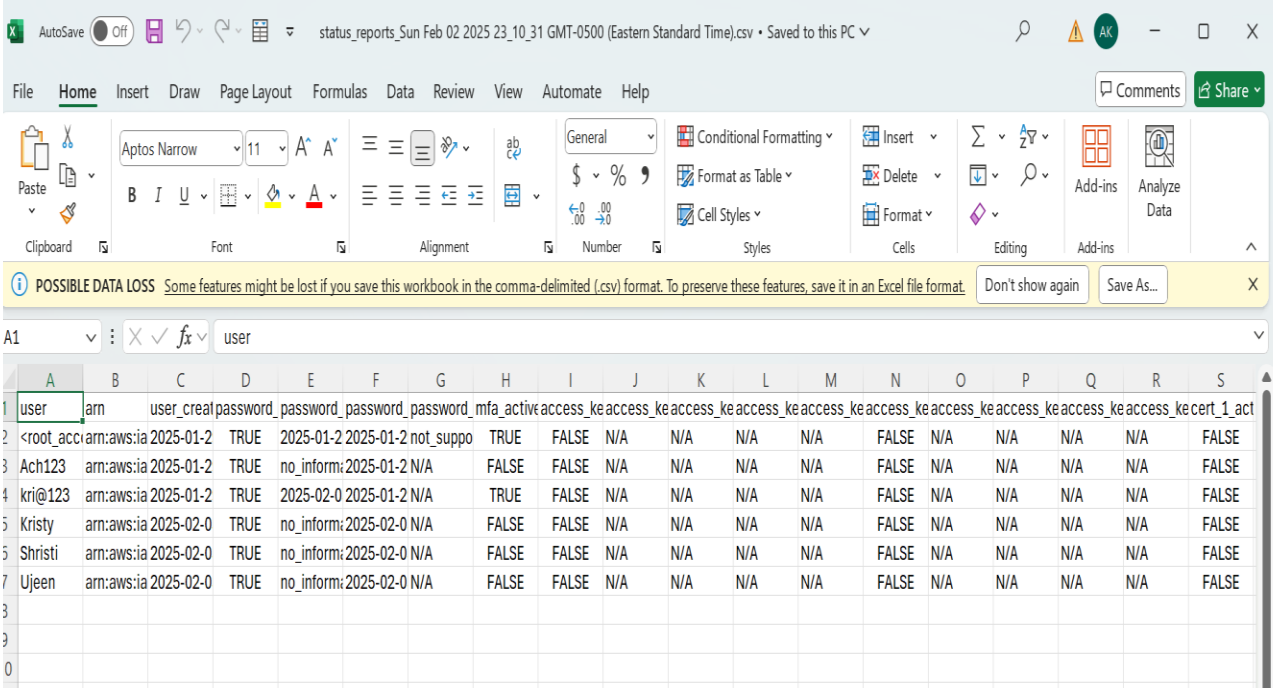


Access key deleted

No access keys. As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. Learn more

15. Creating a Credential Report for Analysis

A credential report provides insights into the security posture of IAM users, listing their access keys, password status, and MFA settings.



status_reports_Sun Feb 02 2025 23_10_31 GMT-0500 (Eastern Standard Time).csv • Saved to this PC

POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.

A1 user

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	user	arn	user_creat	password_	password_	password_	password_	mfa_active	access_ke	access_ke	access_ke	access_ke	access_ke	access_ke	access_ke	access_ke	access_ke	access_ke	cert_1_act
2	<root_acc	arn:aws:ia	2025-01-2	TRUE	2025-01-2	2025-01-2	not_suppo	TRUE	FALSE	N/A	N/A	N/A	N/A	FALSE	N/A	N/A	N/A	N/A	FALSE
3	Ach123	arn:aws:ia	2025-01-2	TRUE	no_inform	2025-01-2	N/A	FALSE	FALSE	N/A	N/A	N/A	N/A	FALSE	N/A	N/A	N/A	N/A	FALSE
4	kri@123	arn:aws:ia	2025-01-2	TRUE	2025-02-0	2025-01-2	N/A	TRUE	FALSE	N/A	N/A	N/A	N/A	FALSE	N/A	N/A	N/A	N/A	FALSE
5	Kristy	arn:aws:ia	2025-02-0	TRUE	no_inform	2025-02-0	N/A	FALSE	FALSE	N/A	N/A	N/A	N/A	FALSE	N/A	N/A	N/A	N/A	FALSE
6	Shristi	arn:aws:ia	2025-02-0	TRUE	no_inform	2025-02-0	N/A	FALSE	FALSE	N/A	N/A	N/A	N/A	FALSE	N/A	N/A	N/A	N/A	FALSE
7	Ujeen	arn:aws:ia	2025-02-0	TRUE	no_inform	2025-02-0	N/A	FALSE	FALSE	N/A	N/A	N/A	N/A	FALSE	N/A	N/A	N/A	N/A	FALSE

16. Conclusion

This hands-on experience has provided me with a deeper understanding of AWS IAM and security best practices. By implementing user roles, policies, and access controls, organizations can effectively manage permissions and enhance cloud security. This report serves as a comprehensive demonstration of my IAM configuration knowledge and its application in securing AWS environments.

References

Guo, T. (2022, June 8). *AWS IAM Security Best Practices*. Retrieved from Git guardian: https://blog.gitguardian.com/aws-iam-security-best-practices/