

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

testemp1

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

☒ Enable console access - optional

Enables a password that allows users to sign in to the AWS Management Console.

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 three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # \$ % ^ & * () _ + - (hyphen) = [] { } | ' "

☐ 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.

☒ For programmatic access, you can generate access keys after you create the user. [Learn more](#)

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.

Permissions policies (1/1038)

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

Filter distributions by text, property or value

ec2

38 matches

Clear filters

	Policy name	Type	Attached entities
<input type="checkbox"/>	AmazonEC2ContainerRegistryFullA...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerRegistryPow...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServiceEvent...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServiceforEC...	AWS managed	0
<input type="checkbox"/>	AmazonEC2ContainerServiceRole	AWS managed	0
<input type="checkbox"/>	AmazonEC2FullAccess	AWS managed	0
<input checked="" type="checkbox"/>	AmazonEC2ReadOnlyAccess	AWS managed	0
<input type="checkbox"/>	AmazonEC2RoleforAWSCodeDeploy	AWS managed	0
<input type="checkbox"/>	AmazonEC2RoleforAWSCodeDeplo...	AWS managed	0
<input type="checkbox"/>	AmazonEC2RoleforDataPipelineRole	AWS managed	0
<input type="checkbox"/>	AmazonEC2RoleforSSM	AWS managed	0
<input type="checkbox"/>	AmazonEC2RolePolicyForLaunchW...	AWS managed	0
<input type="checkbox"/>	AmazonEC2SpotFleetAutoscaleRole	AWS managed	0
<input type="checkbox"/>	AmazonEC2SpotFleetTaggingRole	AWS managed	0

<input type="checkbox"/>		AmazonEC2ContainerServiceEvent...	AWS managed	0
<input type="checkbox"/>		AmazonEC2ContainerServiceforEC...	AWS managed	0
<input type="checkbox"/>		AmazonEC2ContainerServiceRole	AWS managed	0
<input type="checkbox"/>		AmazonEC2FullAccess	AWS managed	0
<input checked="" type="checkbox"/>		AmazonEC2ReadOnlyAccess	AWS managed	0
<input type="checkbox"/>		AmazonEC2RoleforAWSCodeDeploy	AWS managed	0
<input type="checkbox"/>		AmazonEC2RoleforAWSCodeDeplo...	AWS managed	0
<input type="checkbox"/>		AmazonEC2RoleforDataPipelineRole	AWS managed	0
<input type="checkbox"/>		AmazonEC2RoleforSSM	AWS managed	0
<input type="checkbox"/>		AmazonEC2RolePolicyForLaunchW...	AWS managed	0
<input type="checkbox"/>		AmazonEC2SpotFleetAutoscaleRole	AWS managed	0
<input type="checkbox"/>		AmazonEC2SpotFleetTaggingRole	AWS managed	0

Step 3

Review and create

Step 4

Retrieve password

User name

testemp1

Console password type

Autogenerated

Require password reset

Yes

Permissions summary

< 1 >

Name	Type	Used as
AmazonEC2ReadOnlyAccess	AWS managed	Permissions policy
IAMUserChangePassword	AWS managed	Permissions policy

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

User created successfully

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

View user

IAM > Users > Create user

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

Email sign-in instructions

Console sign-in URL

https://900437100429.signin.aws.amazon.com/console

User name

testemp1

Console password

Show

Download .csv file

Return to users list

cli.....

IAM > Users > Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Specify user details

User details

User name

kishandli

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

☐ Enable console access - optional
Enables a password that allows users to sign in to the AWS Management Console.

For programmatic access, you can generate access keys after you create the user. [Learn more](#)

Cancel

Next

Step 2
Set permissions

Step 3
Review and create

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 groups

< 1 > ⚙

<input checked="" type="checkbox"/>	Group name	Users	Attached policies	Created
<input checked="" type="checkbox"/>	devops	0	AmazonEC2FullAccess, AutoSca...	2023-01-28 (3 minutes ago)

Permissions boundary - optional

Set a permissions boundary to control the maximum permissions for this user. Use this advanced feature used to delegate permission management to others. [Learn more](#)

Cancel

Previous

Next

IAM > Users

Users (2) Info

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

Find users by username or access key

< 1 > ⚙

<input type="checkbox"/>	User name	Groups	Last activity	MFA	Password age	Active key age
<input type="checkbox"/>	kishandli	devops	Never	None	None	-
<input type="checkbox"/>	testemp1	devops	✔ 18 minutes ago	None	✔ 17 minutes ago	-

IAM > Users > kishancli

kishancli

Delete

Summary

ARN arn:aws:iam::900437100429:user/kishancli	Console access Disabled	Access key 1 Not enabled
Created January 28, 2023, 14:40 (UTC+05:30)	Last console sign-in -	Access key 2 Not enabled

Permissions

Groups (1)

Tags

Security credentials

Access Advisor

Console sign-in

Enable console access

Console sign-in link https://900437100429.signin.aws.amazon.com/console	Console password Not enabled
--	---------------------------------

Access keys (0)

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. [Learn more](#)

Create access key

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](#)

Create access key

SSH public keys for AWS CodeCommit (0)

IAM > Users > kishancli > Create access key

Step 1

Access key best practices & alternatives

Step 2 - optional

Set description tag

Step 3

Retrieve access keys

Access key best practices & alternatives

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

Command Line Interface (CLI)

You plan to use this access key to enable the AWS CLI to access your AWS account.

Local code

You plan to use this access key to enable application code in a local development environment to access your AWS account.

Application running on an AWS compute service

You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

Third-party service

You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

Application running outside AWS

You plan to use this access key to enable an application running on an on-premises host, or to use a



Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

Step 3
Retrieve access keys

Retrieve access keys

Access key
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
 AKIA5DJR3Y6G3IS3VCSS	 ***** Show

Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [Best practices for managing AWS access keys](#).