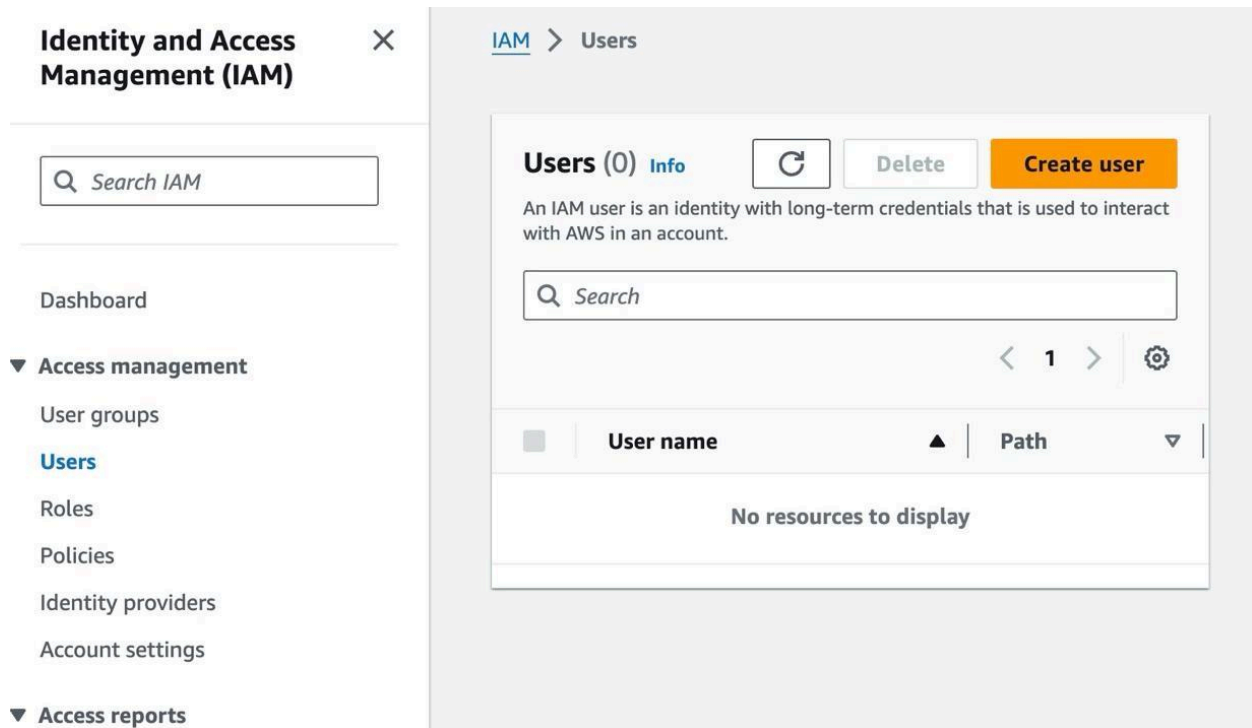
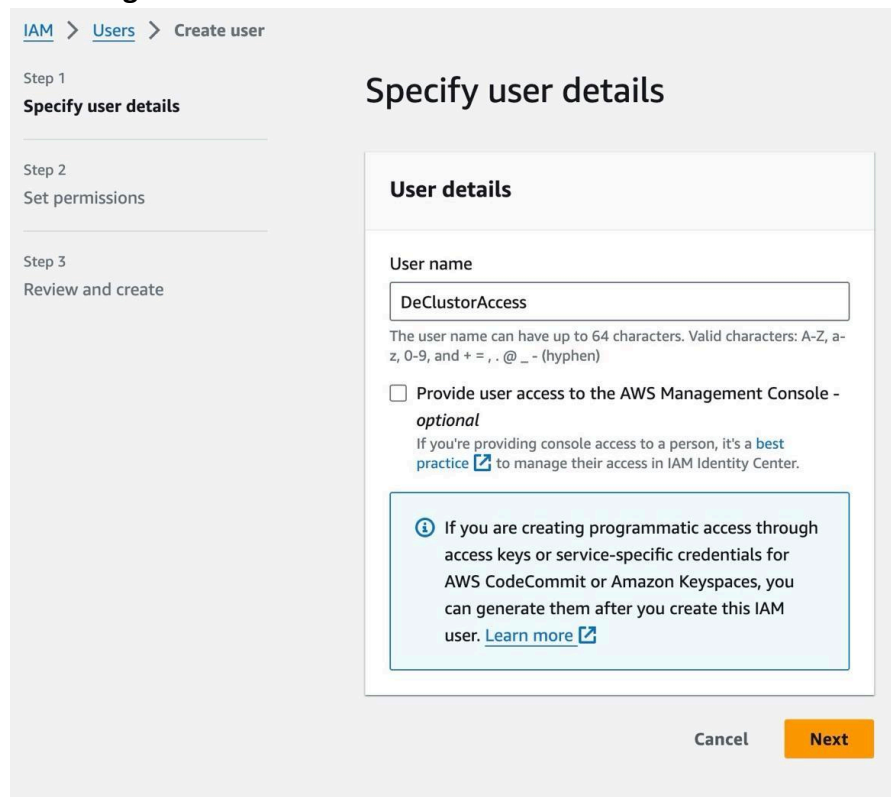


1. Log in to AWS Management Console, navigate to Identify and Access Management(IAM), and create a user



2. Configure User Details



3. Set Permissions

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.

4. Add AmazonECS_FullAccess policy and CloudwatchFullAccess policy

Permissions policies (2/1218)

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

Filter by Type

ECS

All types

10 matches

	Policy name	Type	Attached entities
☒	AmazonECS_FullAccess	AWS managed	0
☐	AmazonECSInfrastructureRolePolicyForServiceConnectTrans...	AWS managed	0
☐	AmazonECSInfrastructureRolePolicyForVolumes	AWS managed	0

Permissions policies (2/1218)

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

Filter by Type

CloudwatchFullAccess

All types

2 matches

	Policy name	Type	Attached entities
☒	CloudWatchFullAccess	AWS managed	0
☐	CloudWatchFullAccessV2	AWS managed	0

Permissions policies (1/1218)

Filter by Type

organization 11 matches

Policy name	Type	Attached entities
☐ AmazonDetectiveOrganizationsA...	AWS managed	0
☐ AmazonDevOpsGuruOrganizatio...	AWS managed	0
☐ AWSBackupOrganizationAdminA...	AWS managed	0
☐ AWSConfigRoleForOrganizations	AWS managed	0
☐ AWSOrganizationsFullAccess	AWS managed	0
☒ AWSOrganizationsReadOnlyAccess	AWS managed	0
☐ AWSOrganizationsServiceTrustPo...	AWS managed	1
☐ AWSResourceExplorerOrganizati...	AWS managed	0
☐ AWSSecurityHubOrganizationsAc...	AWS managed	0
☐ AWSWellArchitectedOrganizatio...	AWS managed	0
☐ Health_OrganizationsServiceRole...	AWS managed	0

We need these three policies to manage your AWS accounts and get metric data from Cloudwatch

Permissions policies (3)

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

Filter by Type

Search All types

Policy name	Type	Attached via
☐ AmazonECS_FullAccess	AWS managed	Directly
☐ AWSOrganizationsReadOnlyAccess	AWS managed	Directly
☐ CloudWatchFullAccess	AWS managed	Directly

5. Click on the user you just create

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

IAM > Users

Users (1) Info

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

Search

User name	Path	Group	Last activity	MFA	Password age	Console last
☐ DeClusterAccess	/	0	8 minutes ago	-	-	-

6. Create Access key for DeClusterAccess user

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

IAM > Users > DeClusterAccess

DeClusterAccess

Delete

Summary

ARN
arn:aws:iam::.../DeClusterAccess

Console access
Disabled

Created
July 03, 2024, 14:19 (UTC-07:00)

Last console sign-in
-

Access key 1
Used today. 3 days old.

Access key 2
Create access key

Permissions

Groups

Tags (1)

Security credentials

Access Advisor

Console sign-in

Enable console access

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

Console password
Not enabled

Multi-factor authentication (MFA) (0)

Remove

Resync

Assign MFA device

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](#)

IAM > Users > DeClusterAccess > 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.

Use case

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 authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.

Q Search IAM

Dashboard

▼ Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

▼ Access reports

Access Analyzer

Access keys (1)

Create access key

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](#)

AKIATCKASPWJL6VJPL7H

Actions ▼

Description

Status

Allow Declustor to get access to my ECS cluster

✔ Active

Last used

Created

4 minutes ago

3 days ago

Last used region

Last used service

us-east-2

cloudwatch