

AWS CodePipeline Permission Access - Setup Guide

Overview

AWS CodePipeline does not have a built-in permission management feature. Instead, access control is managed through AWS IAM (Identity and Access Management).

This guide covers how to create IAM users with limited access for developers, allowing them to:

✅ Allowed:

- View pipeline status
- View execution logs
- Re-trigger pipeline (Release change)
- Retry failed stage
- Stop execution

❌ Not Allowed:

- Create pipeline
- Edit pipeline
- Delete pipeline

[Overview](#)

[Step 1: Create IAM Policy](#)

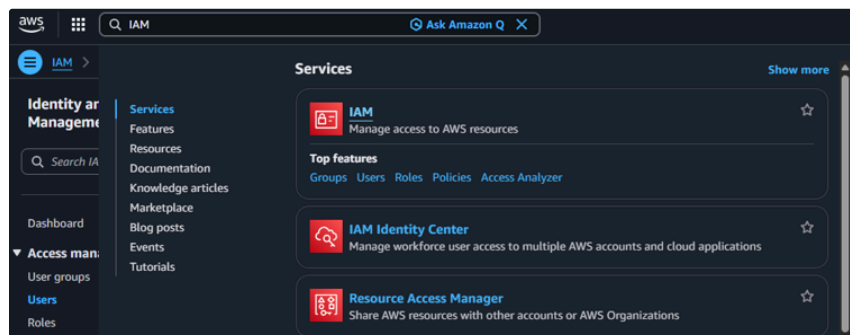
[Step 2: Create User Group](#)

[Step 3: Create First User](#)

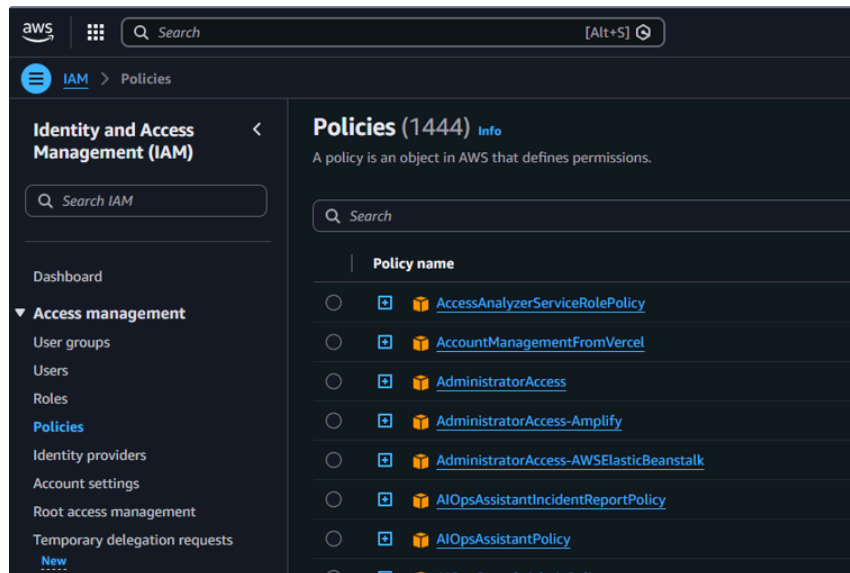
[Step 4: Share Login Info with Developer](#)

Step 1: Create IAM Policy

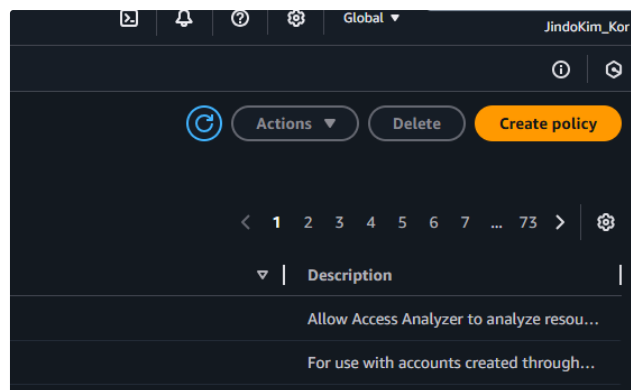
i A policy defines what actions are allowed. In this step, we create a custom policy that grants limited CodePipeline access.



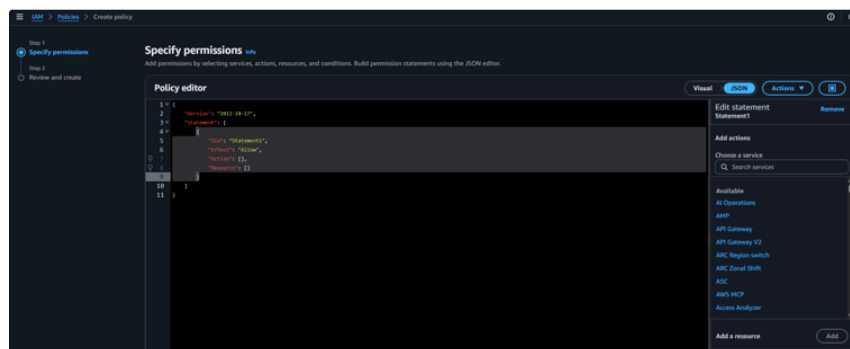
Search/Access 'IAM' service in AWS



Open 'Policies' on the left side of the menu



Click 'Create policy' button to create a collection of limited access for developers



Click 'JSON' Button to open JSON Editor

```
Policy editor
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "codepipeline:GetPipeline",
8         "codepipeline:GetPipelineState",
9         "codepipeline:GetPipelineExecution",
10        "codepipeline:ListPipelineExecutions",
11        "codepipeline:ListActionExecutions",
12        "codepipeline:ListActionTypes",
13        "codepipeline:ListPipelines",
14        "codepipeline:ListTagsForResource",
15        "codepipeline:ListRuleExecutions",
16        "codepipeline:StartPipelineExecution",
17        "codepipeline:StopPipelineExecution",
18        "codepipeline:RetryStageExecution"
19      ],
20       "Resource": "*"
21     }
22   ]
23 }
```

How it should look after applying the JSON file provided below

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "codepipeline:GetPipeline",
8         "codepipeline:GetPipelineState",
9         "codepipeline:GetPipelineExecution",
10        "codepipeline:ListPipelineExecutions",
11        "codepipeline:ListActionExecutions",
12        "codepipeline:ListActionTypes",
13        "codepipeline:ListPipelines",
14        "codepipeline:ListTagsForResource",
15        "codepipeline:ListRuleExecutions",
16        "codepipeline:StartPipelineExecution",
17        "codepipeline:StopPipelineExecution",
18        "codepipeline:RetryStageExecution"
19      ],
20       "Resource": "*"
21     }
22   ]
23 }
```

Note: Below is a brief explanation of each permission included in this policy.

Permission	Description
GetPipeline	View pipeline configuration
GetPipelineState	View current pipeline status
GetPipelineExecution	View execution details

ListPipelineExecutions	View execution history
ListActionExecutions	View action-level execution details
ListActionTypes	View available action types
ListPipelines	View list of all pipelines
ListTagsForResource	View pipeline tags
ListRuleExecutions	View rule execution details
StartPipelineExecution	Re-trigger pipeline
StopPipelineExecution	Stop running pipeline
RetryStageExecution	Retry failed stage

5688 of 6144 characters remaining

[Cancel](#)[Next](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Click 'Next' Button

IAM > Policies > Create policy

Step 1
Specify permissions

Step 2
Review and create

Review and create

Review the permissions, specify details, and tags.

Policy details

Policy name

Enter a meaningful name to identify this policy.

helprr-backend-developer-policy

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

Description - optional

Add a short explanation for this policy.

Limited access for backend developers to view, re-trigger, and stop pipelines

Maximum 1,000 characters. Use alphanumeric and '+', '@', '-' characters.

Permissions defined in this policy

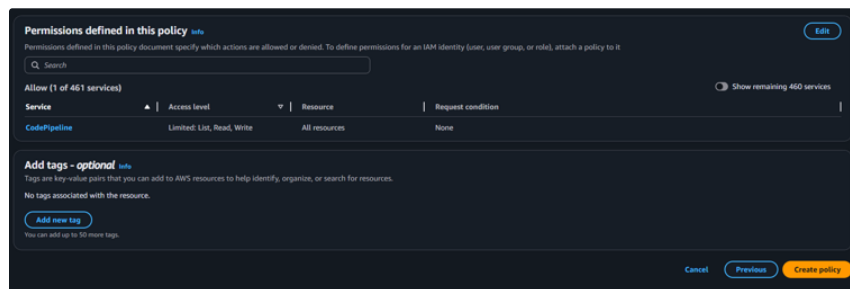
Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity user, user group, or role, use the format: `arn:aws:iam::aws:policy/AmazonEC2RoleforAWSLambda`.

Q Search

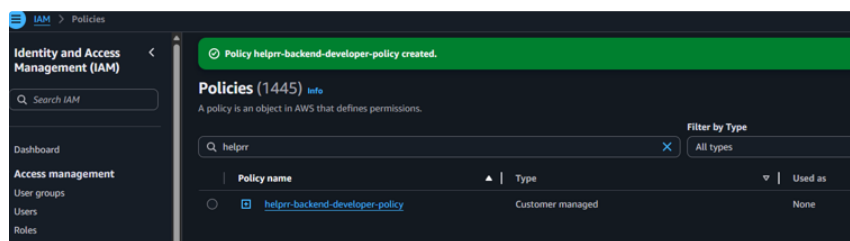
Fills up the Policy name and description

- Example:

- Policy name: `helprr-backend-developer-policy`
- description: `Limited access for backend developers to view, re-trigger, and stop pipelines`



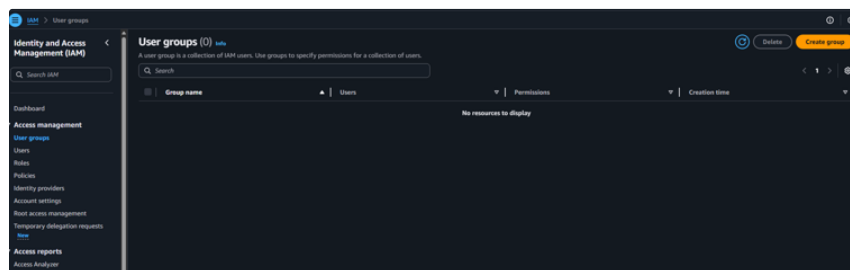
Confirm if 'CodePipeline' Service in the list and Click 'Create policy' button



The created success toast message showed and search the policy name to see whether it is under policy list

Step 2: Create User Group

- A user group allows you to **manage permissions for multiple users at once**. Instead of assigning the policy to each user individually, we attach it to a group. Any user added to this group will automatically have the same permissions.



Open 'User groups' on the left side of the menu and Click 'Create group' Button

Create user group

Name the group

User group name
Enter a meaningful name to identify this group.
helprr-backend-developer-group
Maximum 128 characters, use alphanumeric and '*'-'_.' characters.

Add users to the group - Optional (0)

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 | Group | Last activity | Creation time

No resources to display

Attach permissions policies - Optional (1/1114)

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

Filter by Type

helprr-backend-developer

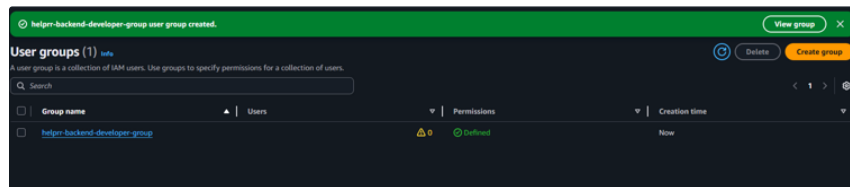
Policy name | Type | Used as | Description

helprr-backend-developer-policy | Customer managed | None | Limited access for backend developers...

Cancel Create user group

Fills up the 'User group name' and Select the custom policy created in previous step, and then Click the 'Create user group' button

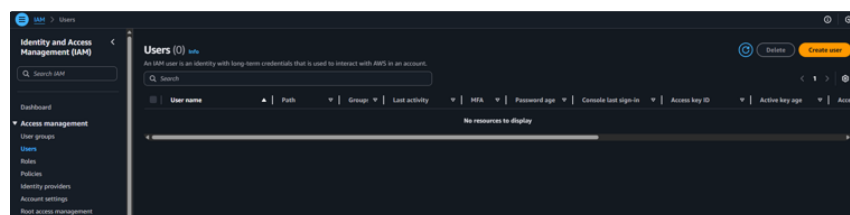
- Example:
 - group name: `helprr-backend-developer-group`
 - Attach permission policies: `helprr-backend-developer-policy`



The success message will appear and the group will be shown in the list

Step 3: Create First User

- Now we create an IAM user for the developer. This user will be added to the group we just created, which automatically grants them the limited pipeline access.



Open 'User' on the left side of the menu and Click 'Create user' Button

Specify user details

User details

User name
jin

☒ **Provide user access to the AWS Management Console - optional**
In addition to console access, users with `iam:consolelogin` 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.

☐ **Show password**

☒ **Users must create a new password at next sign-in - Recommended**
Users automatically get the `iam:ResetChangePassword` policy to allow them to change their own password.

Next

Fills up the 'User name' form and check 'Provide user access to the AWS Management Console - optional', And then Click 'Next' Button

Important: You must check "Provide user access to the AWS Management Console". Without this, the developer cannot log in to AWS Console to view pipelines.

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 that appropriate group.

User groups (1/1)

Group name	Users	Attached policies	Created
helpdesk-developer-group	0	helpdesk-developer-policy	2025-12-29 0 minutes ago

Next

Check the user group created in the previous step, and then Click the 'Next' Button

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name jin	Console password type Autogenerated	Require password reset Yes
------------------	--	-------------------------------

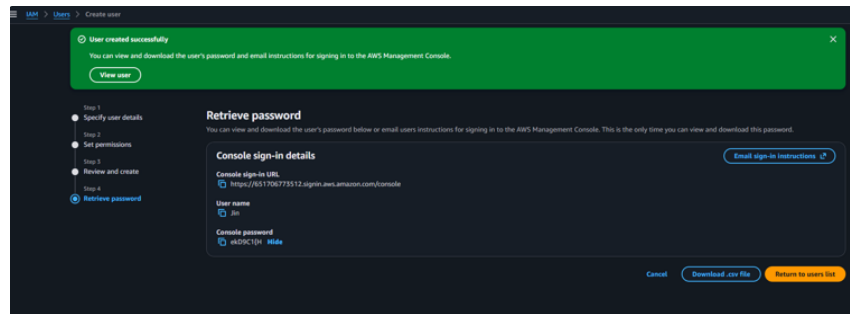
Permissions summary

Name	Type	Used as
helpdesk-developer-group	Group	Permissions group
iam:ResetChangePassword	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.

Create user

Review the settings and Click the 'Create user' Button



The success message will appear in green

⚠ Important: This is **the only time** you can view the password. Make sure to copy it before closing this page.

Step 4: Share Login Info with Developer

After creating the user, you will see the login credentials. Share the following information with the developer:

- **Console sign-in URL:** The URL to access AWS Console
- **User name:** The username created
- **Console password:** The temporary password