

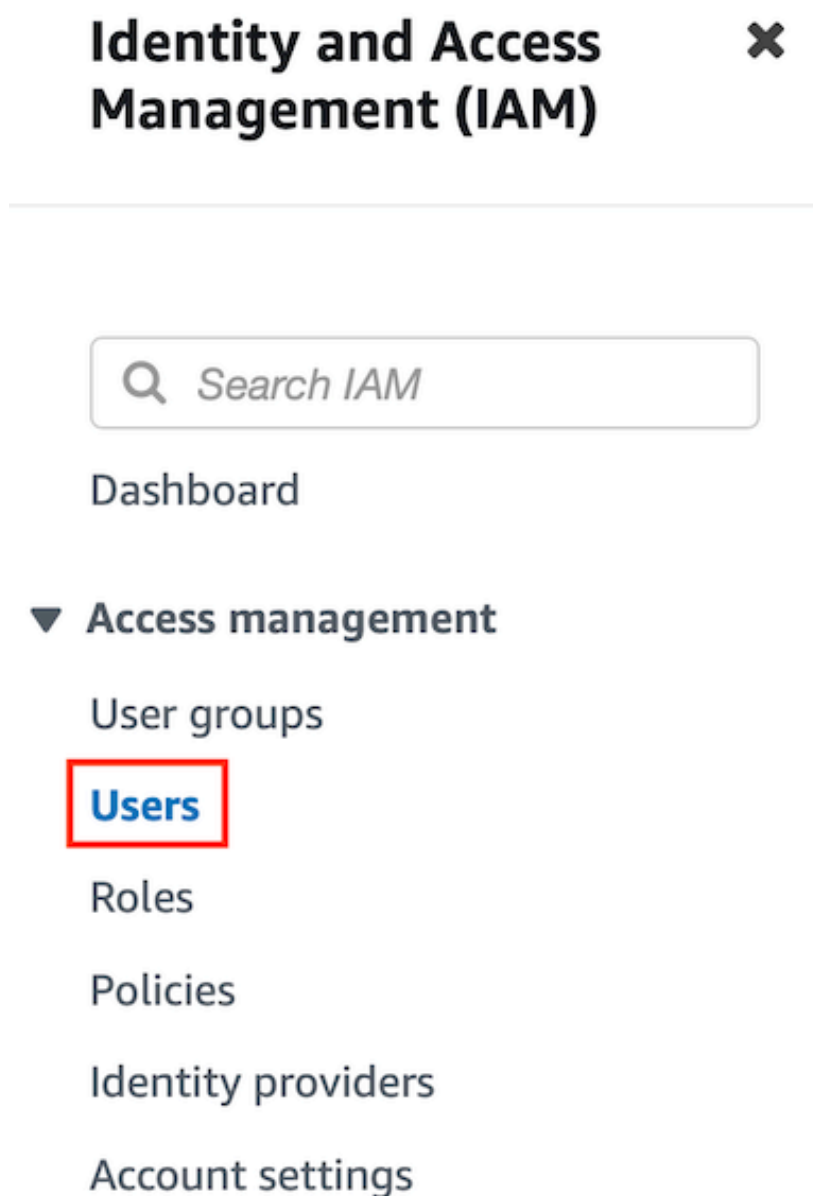
Creating the IAM on AWS

To create resources in AWS through the AWS CDK on the development machine, it is necessary to create a user in the AWS account, with administrator permissions, as detailed in the following steps.

1) Creating the IAM user:

To use the AWS CLI and deploy the infrastructure created with the AWS CDK project, you must first create a user in AWS IAM with specific permissions. This user's credentials will be used in the following topic. To get started, open the AWS console and go to the IAM service.

Within this console, click on the left side menu, on the option `Access Management -> Users`, as in the following figure:



On this screen, click on the `Add user` button, in the top right corner of the page.

On the first user creation screen, enter a name you want and **LEAVE UNCHECKED** the `Enable console access` option, as in the following figure:

Specify user details

User details

User name

aws_cdk

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](#)

Then click `Next`. On this screen, select the `Attach policies directly` option and choose the policy named `Administrator Access`:

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 (1/1085)

Create policy

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

Filter distributions by text, property or value

< 1 2 3 4 5 6 7 ... 55 >

	Policy name	Type	Attached entities
<input type="checkbox"/>	<div><div></div>AccessAnalyzerServiceRolePolicy</div>	AWS managed	0
<input checked="" type="checkbox"/>	<div><div></div>AdministratorAccess</div>	AWS managed - job function	3

Continue clicking the `Next` button, until the last screen for effective user creation. On this screen, click on the `Create user` button. After the user is created, you will see the list of all the IAM users you have, as in the following example:

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

<input type="checkbox"/>	User name	Groups	Last activity	MFA
<input type="checkbox"/>	amplify	None	✓ 9 days ago	None
<input type="checkbox"/>	aws_cdk	None	Never	None


In the list where your newly created user appears, click on their name to be redirected to the page with their access credentials:

[IAM](#) > [Users](#) > [aws_cdk](#)

aws_cdk

Summary

ARN

 `arn:aws:iam::946835467386:user/aws_cdk`

Created

February 13, 2023, 10:10 (UTC-03:00)

Console access

Disabled

Last console sign-in

-

Permissions

Groups

Tags

Security credentials

Access Advisor

Console sign-in

Console sign-in link

 `https://946835467386.signin.aws.amazon.com/console`

Console password

Not enabled

On this screen, click on the `security credentials` tab and go to the `Access keys` section:

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.

Create access key

No access keys

As a best practice, avoid using long-term credentials like access keys. Instead, use tools v

Create access key

In this session, click on the `Create access key` button. On the next screen, select the `Command Line Interface (CLI)` option, as shown in the following figure:

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.

On this screen, click on the `Next` button and then on the `Create access key` button.

The next screen will display the access key that was created for this user. Copy the values for `Access key` and `Secret access key`. This credential will be used in the following topic.

2) Configuring the AWS CLI with the credentials of the user created in IAM:

To configure the AWS CLI on your development machine, you must provide the credentials of the user created in IAM in the previous topic. To do this, open a terminal and type the following command:

```
aws configure
```

In the first requested parameter, enter the `Access Key ID` of the user created in IAM. Then provide the `Secret Access Key`. The third parameter is the desired region, which should be `us-east-1`. The last parameter is the command output format, which can be configured as `json`.

All resources that are created with the AWS CDK during this course will be created on behalf of that user.



siecola.com.br