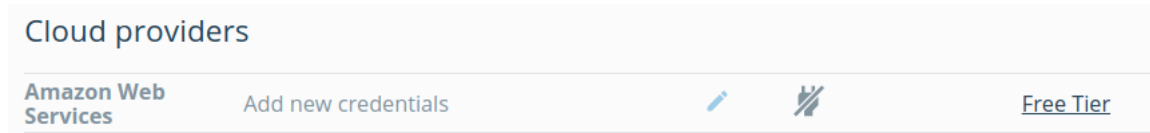


## Ligando com a sua conta da Amazon

### Transcrição

Com as contas do AWS e do Docker Cloud criadas, podemos *linkar* as duas. Para isso, no menu lateral da esquerda, clicamos em **Cloud Settings**, vamos até a sessão **Cloud providers**, e em **Amazon Web Services**, clicamos no botão **Connect provider**, com um símbolo de tomada:



Será aberta uma página para adicionarmos as credenciais da Amazon, que possui um passo-a-passo, que pode ser acessado [aqui \(https://docs.docker.com/docker-cloud/infrastructure/link-aws/\)](https://docs.docker.com/docker-cloud/infrastructure/link-aws/).

### Criando uma política para o Docker Cloud

Primeiramente, devemos criar um **dockercloud-policy**, na [página do IAM \(https://console.aws.amazon.com/iam/home?#/policies\)](https://console.aws.amazon.com/iam/home?#/policies), que é o gerenciador de identidade do AWS, onde damos permissão a alguém para acessar algum recurso da nossa conta do AWS. Na página, clicamos no botão **Create policy**, e na página que abrir, em **Create Your Own Policy**, clicamos em **Select**.

#### Create Your Own Policy

Use the policy editor to type or paste in your own policy.

Select

Abrirá um formulário, nós damos um nome e uma descrição para a política, e no campo **Policy Document**, colocamos o seguinte conteúdo:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:*",
        "iam:ListInstanceProfiles"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

## Review Policy

Customize permissions by editing the following policy document. For more information about the access policy language, see [Overview of Policies](#) in the *Using IAM* guide. To test the effects of this policy before applying your changes, use the [IAM Policy Simulator](#).

### Policy Name

docker-cloud

### Description

A política para liberar os recursos para o [Docker Cloud](#)

### Policy Document

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Action": [  
6         "ec2:*",  
7         "iam:ListInstanceProfiles"  
8       ],  
9       "Effect": "Allow",  
10      "Resource": "*" ,  
11    }  
12  ]  
13 }
```

Com o formulário preenchido, clicamos em **Validate Policy** para verificar se está tudo correto e depois em **Create Policy**.

## Criando um role

Após isso, precisamos criar um **Role**, [nesta página](https://console.aws.amazon.com/iam/home#/roles) (<https://console.aws.amazon.com/iam/home#/roles>). Clicamos em **Create new role**. Na página que abrir, selecionamos a opção **Role for cross-account access**

☒ **Role for cross-account access**

› **Provide access between AWS accounts you own**

Allows IAM users from one of your other AWS accounts to access this account.

Select

› **Provide access between your AWS account and a 3rd party AWS account**

Allows IAM users from a 3rd party AWS account to access this account and enforces use of [External ID](#).

Select

Na segunda opção, em **Provide access between your AWS account and a 3rd party AWS account**, clicamos em **Select**. Aparecerá um formulário com dois campos:

- No campo **Account ID**, temos que colocar o ID da aplicação que irá utilizar a nossa conta do AWS, que é **689684103426**.
- No campo **External ID**, temos que colocar o nome do nosso usuário do **Docker Cloud**.

Em seguida, clicamos em **Next Step**. Na página que abrir, **Attach Policy**, selecionamos a política que criamos anteriormente e clicamos em **Next Step**.

Por fim, damos um nome e uma descrição para o **role** e clicamos em **Create role**. Com o **role** criado, clicamos nele e no campo **Role ARN** é onde temos o ID que dará acesso à nossa conta do AWS, então é bom termos muito cuidado com ele. É ele que colocamos no Docker Cloud.

Ao salvar, as contas estão *linkadas*, assim o Docker já tem a capacidade de administrar a nossa conta do AWS. E para qualquer coisa, a [documentação do Docker](https://docs.docker.com/) (<https://docs.docker.com/>) pode nos auxiliar.

Com isso, no próximo vídeo, podemos subir os nossos *containers* no AWS.