

AWS CLI Buckets

Transcrição

Dando continuidade ao processo de automação relacionado ao processo de envio das imagens para o nosso *bucket* da Alura, vamos dar uma olhada nos comandos que de fato poderemos usar com o **AWS CLI**.

Com o terminal aberto, antes de entrarmos nos comandos, é importante sabermos onde é que o **AWS** guarda as nossas credenciais, que é no diretório oculto `.aws`, na pasta do nosso usuário, dentro do qual existe um arquivo de configuração (`config`) e outro de credenciais (`credentials`). Quando abrimos esse arquivo, vemos o *access key* e o *secret* que o usuário *default* do AWS está utilizando para o acesso. Isso significa que a configuração feita anteriormente já está implementada, e que os comandos que fizermos já estarão acessando a nossa conta conforme esperado.

O *Command Line* serve para todos os serviços da **AWS**, e queremos executar os comandos relacionados ao S3; para tal, começaremos os comandos com `aws s3`. Vamos executar o comando de ajuda para sabermos todos aqueles que poderão ser executados através do S3:

```
aws s3 help
```

Ao executarmos esse comando, seu retorno é uma descrição de todos os que conseguimos analisar e executar através do S3. Um dos pontos mais relevantes é que independentemente de qual comando, sempre trabalharemos com pelo menos um argumento relacionado à localização, seja ela interna (da nossa máquina), ou do S3 (chamada de **S3Uri**).

Para ficarmos cientes, uma **S3Uri** representa um caminho de um objeto, ou de um *bucket*. E isso é representado por um protocolo S3, na seguinte forma: `s3://nomeDoBucket/objeto`.

Criando um *bucket* com o AWS CLI

Para o processo de automação funcionar, precisaremos criar um *bucket* para ele, partindo do princípio de que não há nenhum criado.

No S3, existe o comando `mb`, de *make_bucket*, seguido do nome do *bucket* que queremos criar, seguindo as regras em relação à nomeação deles. Usaremos agora o nome `alura-images`:

```
aws s3 mb alura-images
```

Lembrando que o nome do *bucket* é único, então crie um com um nome diferente dos existentes.

Ao executarmos o comando, receberemos um erro dizendo que o mesmo é inválido. Isso ocorre pois precisaremos trabalhar com a **S3Uri**, que vimos há pouco. Portanto, se queremos criar o balde `alura-images`, deixaremos `s3://alura-images`, e se esse *bucket* estiver disponível, ele será criado:

```
aws s3 mb s3://alura-images
```

Como esse *bucket* ainda não havia sido criado, conseguimos criá-lo com sucesso. Para entrarmos de fato nesse *bucket*, vamos visualizar quais baldes existem em nossa conta:

```
aws s3 ls
```

Com isso, teremos uma relação dos *buckets* associados à nossa conta, incluindo o `alura-images`, que acabamos de criar. E se olharmos no console web do navegador, também visualizaremos o *bucket* lá.

Removendo um *bucket* com o AWS CLI

E se quisermos remover algum *bucket* - lembrando que o mesmo não poderá ser renomeado - iremos executar o comando `rb`, de `remove_bucket`, para deletar o *bucket* que acabamos de criar:

```
aws s3 rb s3://alura-images
```

Removendo um *bucket* não vazio

Por meio do console web, o *bucket* **teste1-alura**, que contém um arquivo de imagem, de que forma exibiremos o seu conteúdo com linha de comando? Poderemos executar o seguinte comando:

```
aws s3 ls s3://teste1-alura
```

Esse comando listará os objetos que estão dentro do *bucket* **teste1-alura**. Agora vamos tentar removê-lo:

```
aws s3 rb s3://teste1-alura
```

Receberemos um erro! Esse é um ponto extremamente importante de ser conhecido: só poderemos remover *buckets* de nossas contas do S3, tanto pela interface de linha de comando quanto pelo console web, **se esse *bucket* estiver vazio**. Ou seja, só conseguiremos removê-lo se antes apagarmos os objetos dentro dele.

Podemos dar uma olhada nas opções que o comando `rb` nos traz:

```
aws s3 rb help
```

Quando executamos esse comando, ele nos mostra seus detalhes, como nome, descrição, uma sinopse de como o comando pode ser executado e as opções que podemos adicionar nele, mais precisamente a opção `--force`.

Essa opção serve justamente para conseguirmos remover todos os objetos do *bucket*, incluindo o próprio. Logo, se precisarmos remover um balde que não esteja vazio, poderemos adicionar a opção `--force`, que irá forçar a remoção dos objetos:

```
aws s3 rb s3://teste1-alura --force
```

A partir desse comando, sua saída nos exibe o `delete`, a nível de objeto, isto é, os objetos que foram removidos, e o comando `remove_bucket`, exibindo a remoção do *bucket* **teste1-alura**.

Então vimos um pouco sobre como manipular nosso *bucket* através do *Command Line Interface* do AWS. Veremos adiante os comandos a nível de objetos, pois acabamos de visualizar os comandos a nível de *buckets*.