



## Mãos à obra: Enviando dados para o bucket

Vamos montar agora nosso **script** que é capaz de enviar os arquivos de **backup** para a Amazon, agora nós temos um novo desafio passado pelos diretores da Mutillidae que seria de salvar cada tabela em um arquivo separado e consolidar toda essa informação em um diretório com a data na qual o **backup** foi realizado.

Para consolidar toda essa informação, vamos criar na **home** um diretório chamado de **backup\_amazon**. Vá até a sua home com o comando:

```
cd ~
```

E depois digite:

```
mkdir backup_amazon
```

Feito isso, entre no diretório **Scripts** e crie um novo script chamado de **backup-amazon.sh**, coloque na primeira linha o interpretador **#!/bin/bash**. Na próxima linha, coloque o caminho completo do diretório **backup\_amazon** que acabamos de criar e vamos mudar para esse diretório:

```
CAMINHO_BACKUP=/home/(usuário)/backup_amazon
cd $CAMINHO_BACKUP
```

Agora a próxima etapa será criar um diretório com a data atual para armazenar os dados da tabela **mutillidae**, caso não exista o diretório, iremos criá-lo:

```
data=$(date +%F)
if [ ! -d $data ]
then
    mkdir $data
fi
```

Na sequência, vamos através de um laço **for** pegar cada tabela do banco mutillidae e iremos salvar no diretório com a data que deverá estar dentro do diretório com o caminho do backup:

```
tabelas=$(mysql -u root mutillidae -e "show tables;" | grep -v Tables)
for tabela in $tabelas
do
    mysqldump -u root mutillidae $tabela > $CAMINHO_BACKUP/$data/$tabela.sql
done
```

Posteriormente que esses nossos dados sejam enviados para nosso bucket que criamos na Amazon:

```
aws s3 sync $CAMINHO_BACKUP s3://[nome do seu bucket]
```

Como nós vimos, vamos programar para esse script ser executado uma vez por dia através do **crontab**, salve as alterações do script e posteriormente, dê permissão de execução com o comando

```
chmod +x backup-amazon.sh
```

Na sequência abra o **crontab** com o comando:

```
sudo crontab -e
```

E coloque na última linha um horário para fazer o backup, dê preferência para um horário próximo a realização desse exercício, por exemplo se forem 10:20, escolha 10:22 para que o crontab execute nosso Script:

```
22 10 * * * [caminho completo para o script backup-amazon.sh]
```

Salve as alterações feitas no crontab e depois quando for aproximadamente o horário colocado no **crontab** vá para seu **bucket** na Amazon. Qual é o resultado?

**IMPORTANTE:** Uma vez que o exercício for finalizado, remova a linha que colocamos no crontab, pois do contrário esses dados serão enviados todos os dias para o bucket da Amazon, consumindo assim os recursos.