# **Capítulo 4 - Construindo minha primeira imagem Docker**

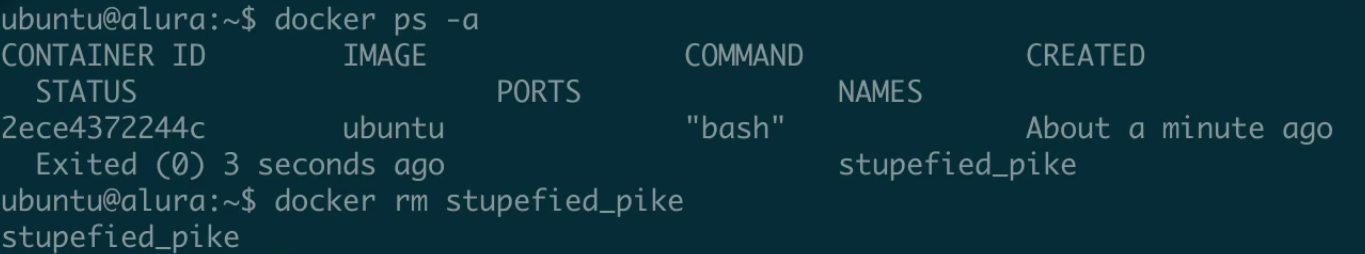
Nesta aula construiremos nossa própria imagem com a instalação. Criemos primeiramente um container para trabalharmos em cima dele:

docker run -it ubuntu bash

Dentro dele fazemos a instalação do Web Server Apache:

apt-get update && apt-get install -y apache2

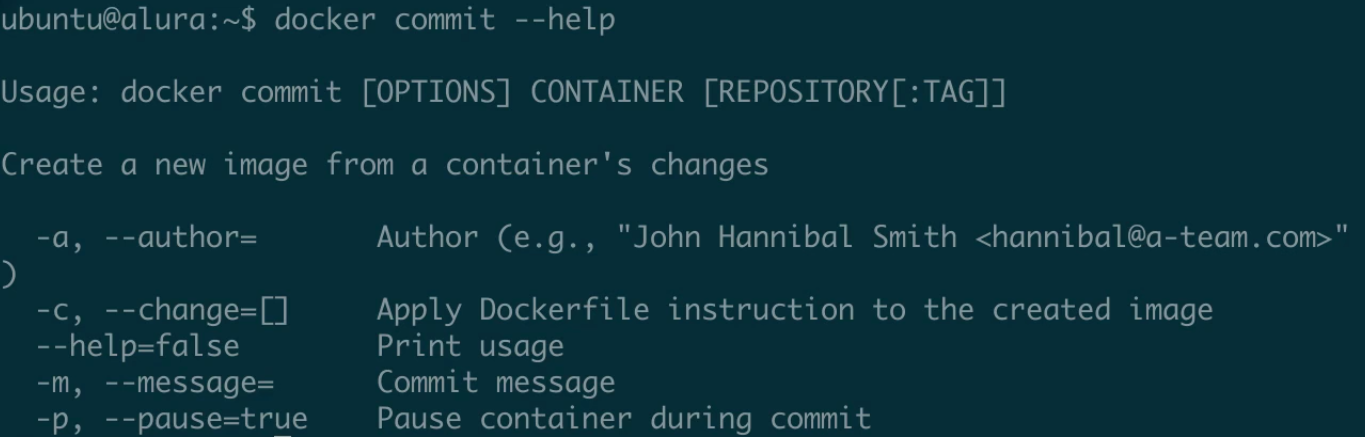
Podemos verificar seu status, e removê-lo:



Se buscarmos pelo Apache que instalamos anteriormente dentro daquele container, verificamos que ele não existe mais. Mesmo recriando novamente o container.

Se quisermos persistir qualquer dado ou pacote instalado, isto terá que ser feito na imagem. Então reinstalamos o Apache dentro do container recém criado.

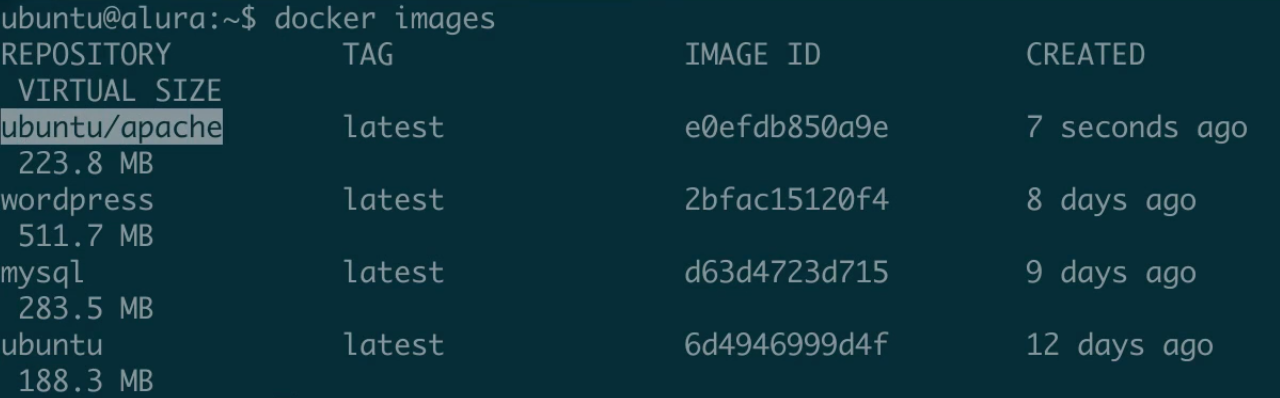
O Docker possui uma feature chamada **commit** com as seguintes opções:



Se fizermos

docker commit -m "instalação do apache" [nome ou id do container] [imagem]/apache

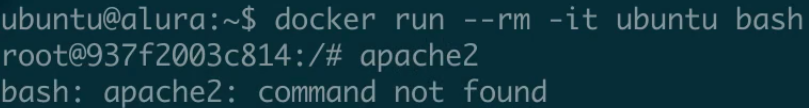
E verificarmos as imagens



Vemos que, além da nossa imagem do ubuntu, foi criada a imagem relacionada ao Apache. A **ubuntu/apache** é um *fork* da nossa imagem base **ubuntu**, a qual usamos para instalar o Apache. O container possui *188.3 MB* + *223.8 MB* de tamanho virtual. Esta é uma feature do Docker que trabalha com camadas.

Agora sim é possível destruir o container sem nos preocuparmos com o Apache ser destruído junto. Além disso, qualquer container que for criado com essa imagem - **ubuntu/apache** - virá com o Apache instalado, pois ela virou persistente.

Vamos criar um container descartável demonstrativo para verificarmos que na imagem do ubuntu não vai persistir aquela instalação inicial do Apache:



De fato, o Apache não é instalado nesse container.

## **Mapeamento de portas**

Vamos destruir nosso container da imagem ubuntu/apache e criamos um container com a opção **-p**:

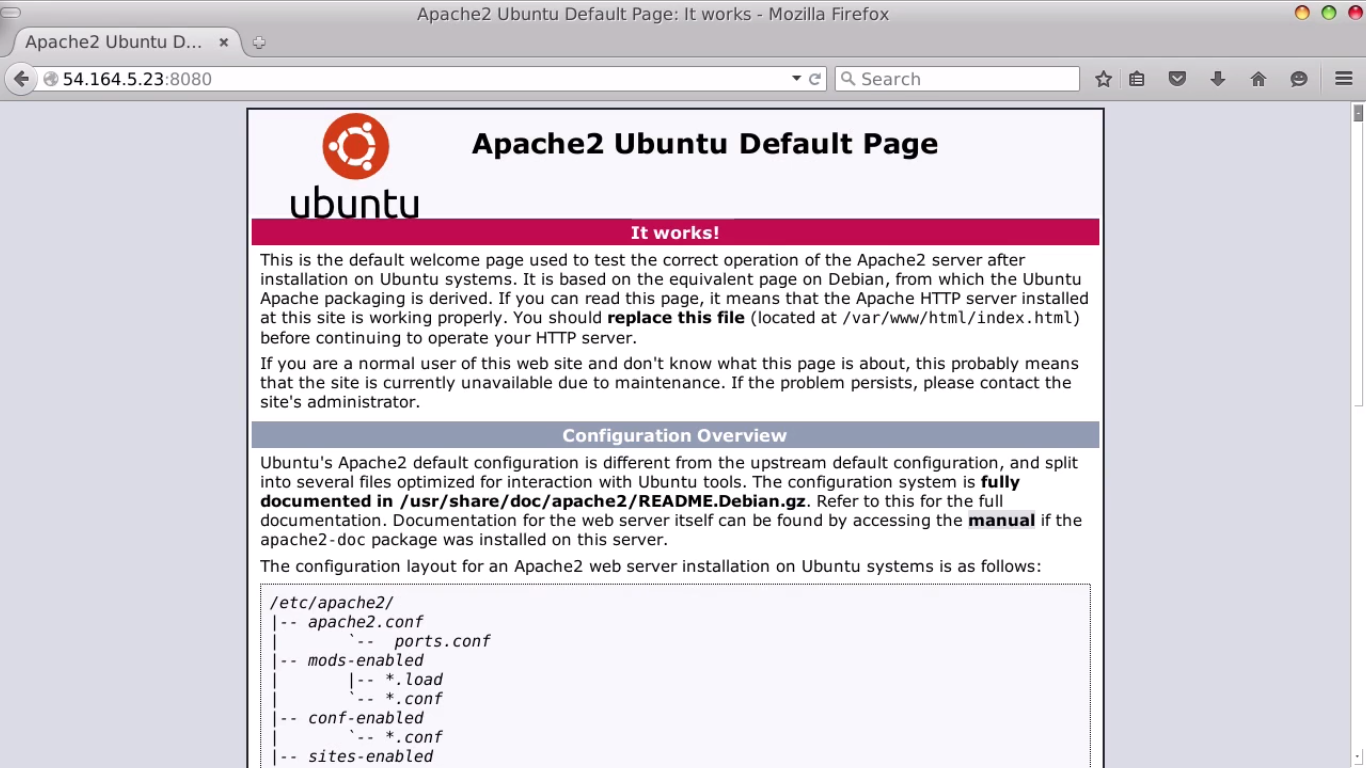
docker run -it -p 8080:80 ubuntu/apache bash

Mas, afinal, o que significa esse "8080:80"? Vimos muito rapidamente em aulas anteriores que são as portas. "8080" é a porta local a qual o Docker irá utilizar. Os dois pontos são a referência, ou mapeamento para a porta 80 de dentro do container. Esta porta é utilizada, normalmente, na configuração do Apache.

Entramos dentro do container fazendo o mapeamento da porta 8080. Rodamos o Apache fazendo:

apachectl start

E vamos no browser testar informando a porta local. Isso nos traz a página inicial do Apache funcionando dentro do container:

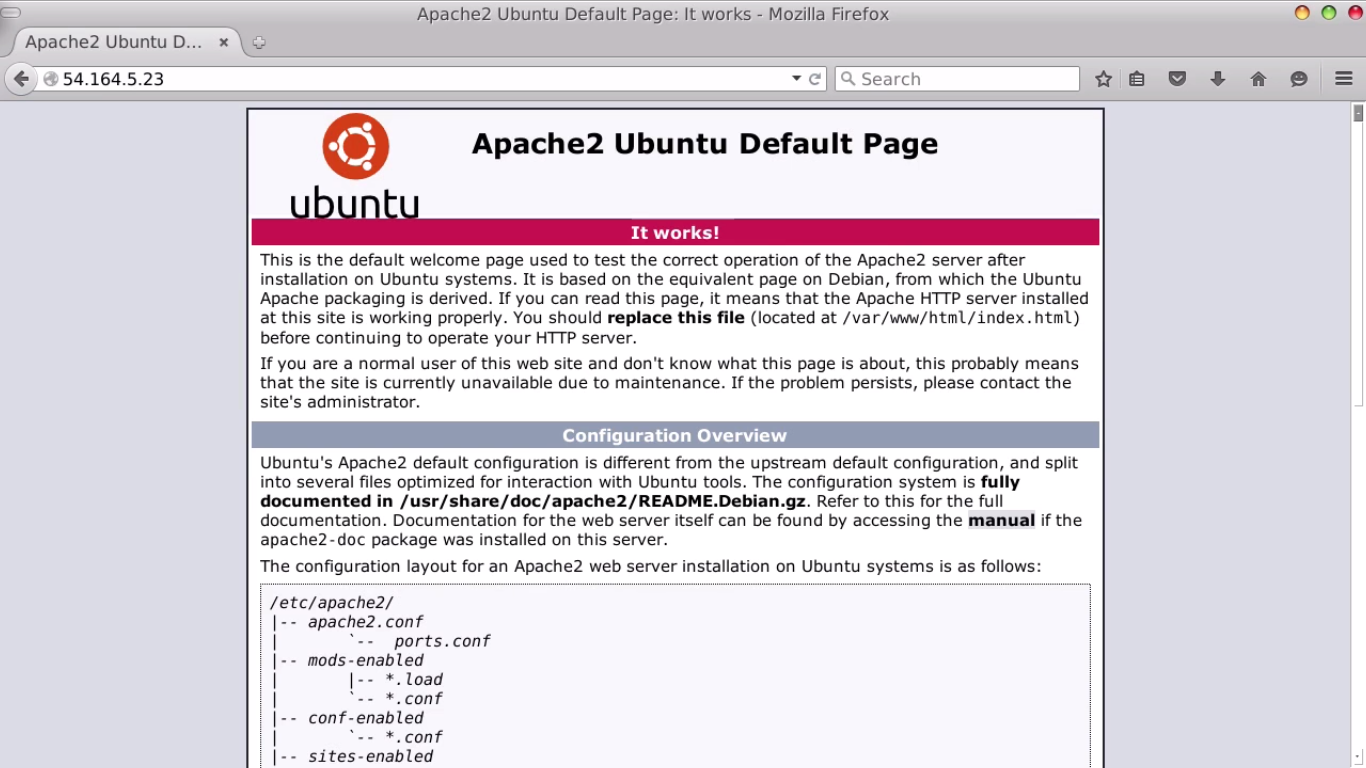


Estamos mapeando a saída do Web Server, o qual está dentro do container, para nossa máquina local.

Vamos repetir tal processo só que agora usando **80:80**:

docker run -it -p 80:80 ubuntu/apache bash

Usando o mesmo ip e porta no navegador, não mais funciona. Agora deixamos claro que o "80" é padrão tanto para a máquina local quanto para o container. Então basta digitar o ip:



Na próxima aula continuaremos com a construção de containers e imagens usando outras instruções.