Instituto Federal de Educação Ciência e Tecnologia do Ceará
Departamento de Telemática - Sistemas Operacionais de Rede

Emanoel Silva de Sousa
Edesson da Silva Mendes

Avaliação 09 - Docker Compose

O Compose é uma ferramenta para definir e executar aplicativos Docker de vários contêineres. Através do compose, você usa o arquivo YAML para configurar os serviços de seu aplicativo, com isso com um único comando, você cria e inicia todos os serviços de sua configuração.

O Docker Compose é utilizado justamente para facilitar o provisionamento e gerenciamento de multi-contêineres principalmente em ambientes de desenvolvimento, testes automatizados ou cenários de execução em um único host.

Ele também ainda realiza o isolamento do ambiente de um conjunto de contêineres separando-os por nome de projeto, que geralmente é dado o nome do local onde ele está sendo executado.

Recursos

Os principais recursos que torna o Docker Compose eficaz:
Vários ambientes isolados em um único host.
Preserva os dados do volume quando os contêineres são criados.
Recrie apenas contêineres que foram alterados.
Suportam variáveis e movem uma composição entre ambientes.

Serviços

O Docker Compose trata todos os contêineres que desejamos executar como serviços e é dessa forma que devemos referenciá-los no arquivo de configuração. Tratando os contêineres dessa forma, o Docker Compose consegue atribuir o mesmo conjunto de configurações para todos os contêineres que fizerem parte deste serviço.

Estrutura do arquivo

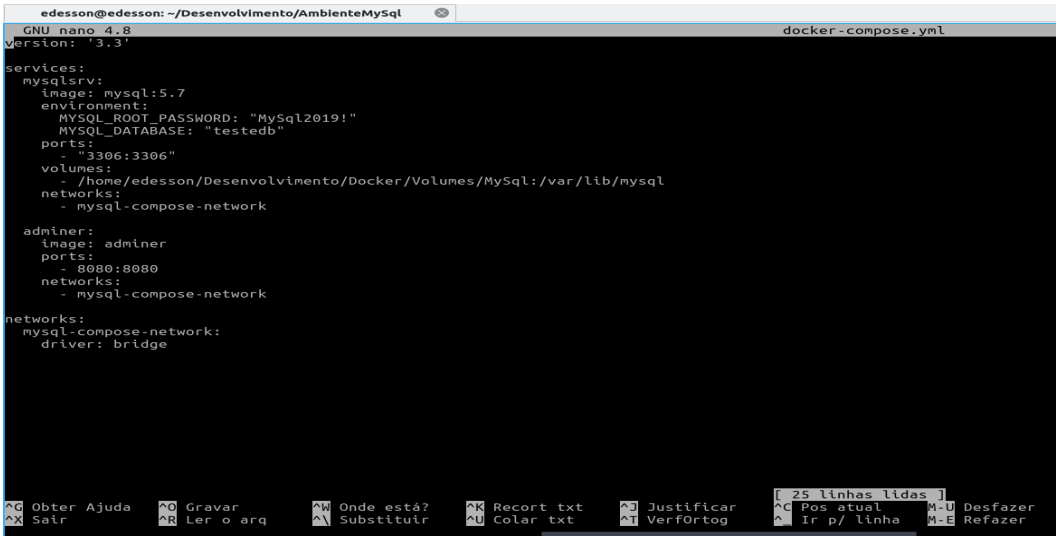
Para executar nossos contêineres com o Compose é necessário possuir um arquivo YAML que contenha todas as informações e parâmetros que desejamos passar para a execução deles. Por padrão, os comandos docker-compose procuram um arquivo no diretório corrente nomeado como docker-compose.yml, porém é possível indicar um outro nome de arquivo e também em um outro local passando o parâmetro -f.

version: '3.3': versão utilizada do Docker Compose,muda de acordo com a versão.
services: inicia um serviço e tudo abaixo dessa linha faz parte deste serviço.
image: imagem do container.
volumes: volume que é utilizado no container.
environment: variáveis de ambiente utilizadas para acessar o container.

Docker Compose - MySql e Adminer

Para a simulação foi utilizado o sistema Lubuntu. O arquivo **docker-compose.yml** será criado e pode ser salvo em uma pasta qualquer. O comando **docker-compose up -d** cuidará da criação da network e dos containers esperados.

A seguir está o conteúdo do arquivo **docker-compose.yml**:



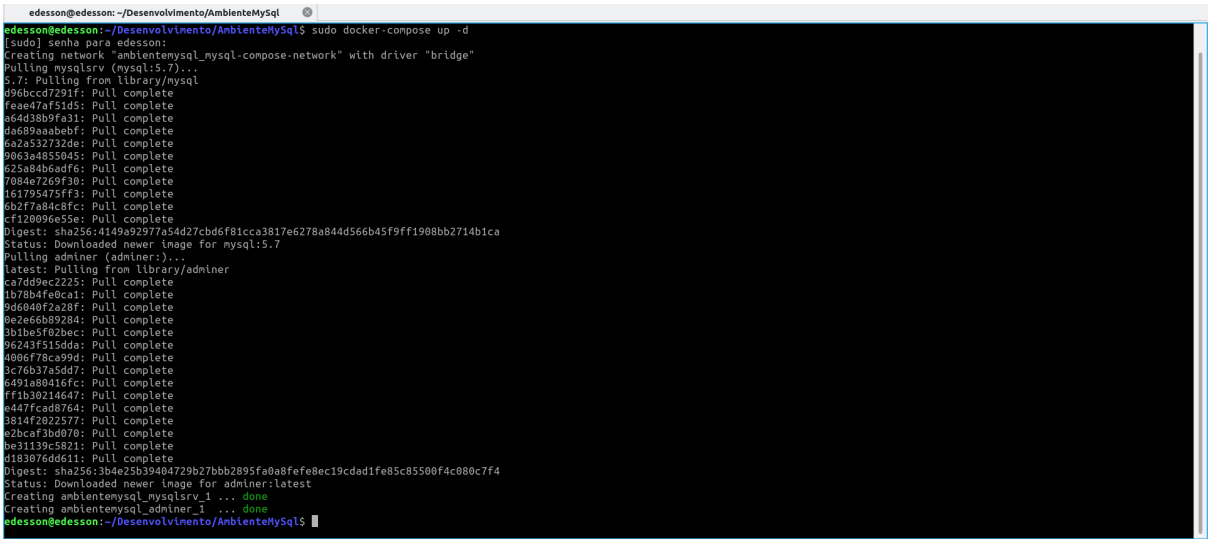
O serviço mysqlsrv é a instância do MySQL 5.7 que vai ser criada para acesso na porta 3306. O serviço adminer é o container que vai permitir a execução do ADMINER na porta 8080.

As imagens que foram referenciadas no arquivo, serão baixadas caso não existam.

Foi especificado um volume para o mysqlrsv, onde serão salvos os arquivos de dados no Lubuntu: (/home/edesson/Desenvolvimento/Docker/Volumes/MySql).

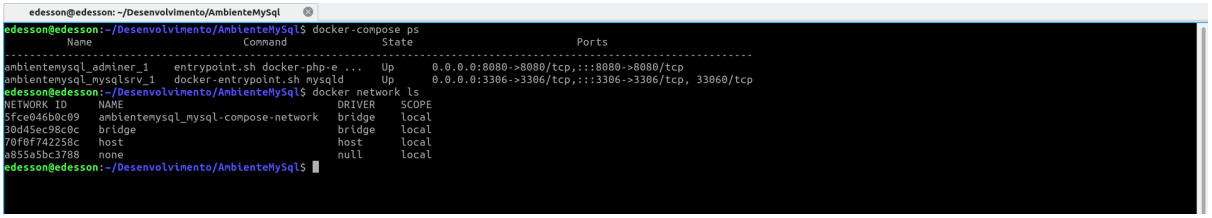
Por meio da network mysql-compose-network acontecerá a comunicação entre os containers mysqlsrv e adminer.

A seguir o resultado do comando **sudo docker-compose up -d**:

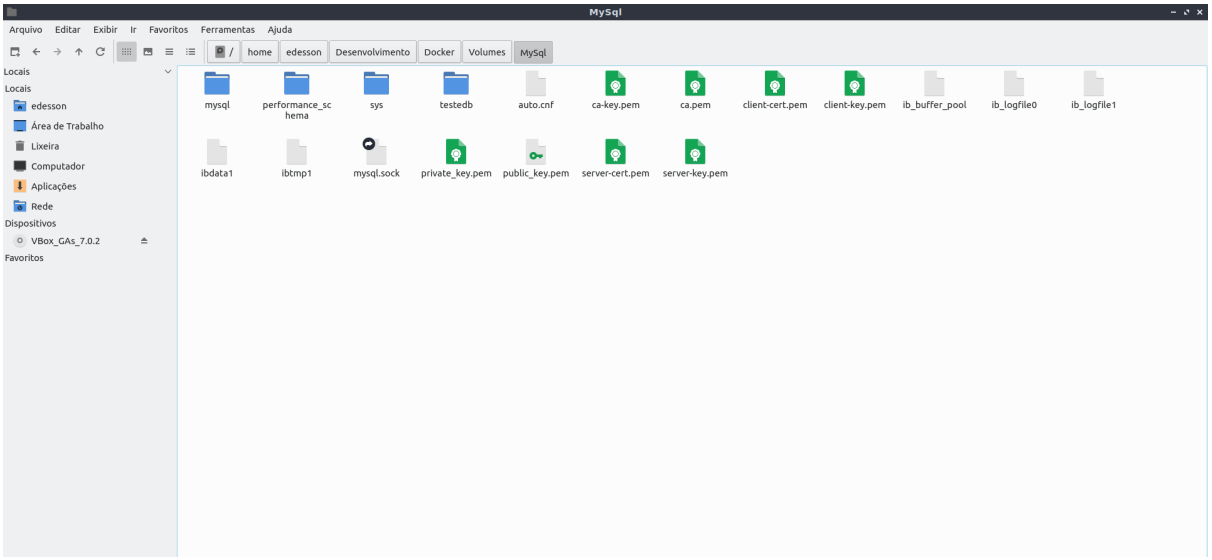


O comando **docker-compose ps** ps mostrará que os containers do MySQL (porta 3306) e do Adminer (porta 8080) foram gerados corretamente e se encontram em

execução. O comando **docker network ls** mostra que a rede mysql-compose-network foi criada com sucesso (como ambientemysql_mysql-compose-network, resultado da concatenação com o nome do diretório em que se encontra o arquivo docker-compose.yml).

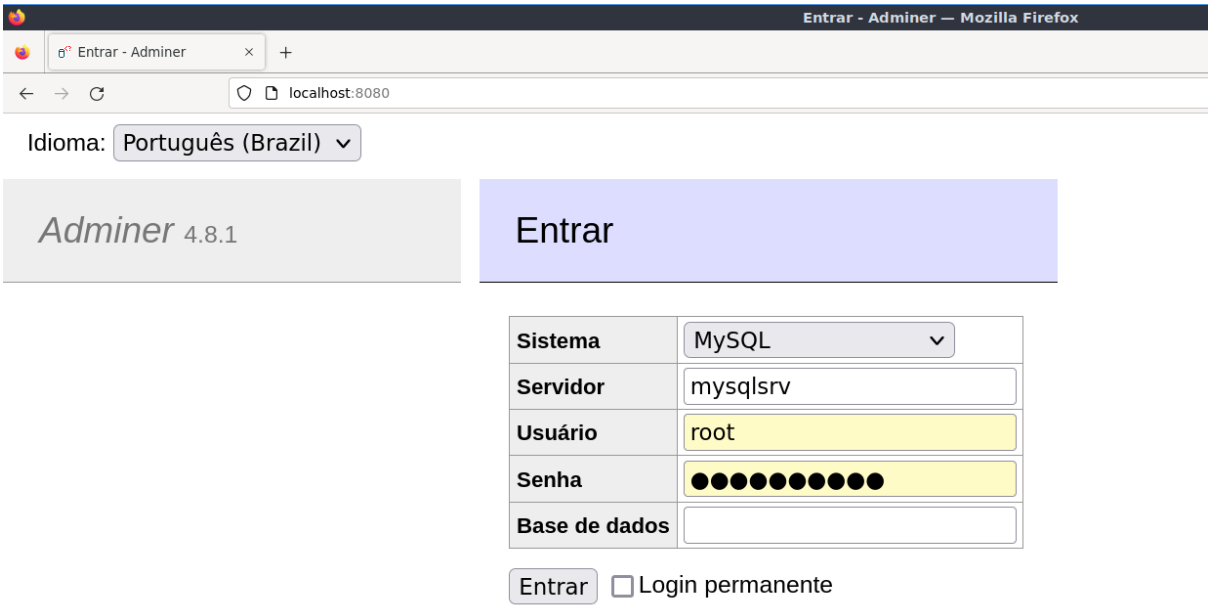


A seguir arquivos e diretórios que foram criados para o volume definidos no arquivo **docker-compose.yml:**



Testando o ambiente:

Testando o Adminer via locahost. Informar o nome do container do MySql como servidor e a senha indicada no MYSQL_ROOT_PASSWORD do arquivo docker-compose.yml.



Concluído o processo de login, aparecerá o banco de dados testedb (também especificado em docker-compose.yml, através da variável de ambiente MYSQL_DATABASE) como uma das opções disponíveis: