

O que é container?

Container é um ambiente isolado contido em um servidor que, diferentemente das máquinas virtuais, divide um único host de controle. Para contextualizar melhor, vamos imaginar um navio cargueiro com vários containers dentro. Se um dos recipientes se danificar, não afetará os outros ou o navio, pois cada um está isolado e protegido. Afinal, são isolados, protegidos e estão carregando seus próprios produtos.

Trazendo para o mundo do desenvolvimento, cada container possui uma função e sua responsabilidade. Caso um deles sofra um dano, o funcionamento do sistema não para e a função afetada é redirecionada para um novo container.

É acessado a partir do kernel da sua máquina *host* (o kernel promove a ligação entre o *hardware* e o *software* do computador, proporcionando a execução dos aplicativos a partir dos recursos disponíveis na máquina).

Cada *container* tem seu próprio sistema, bem como registro, esquema de funcionamento, banco de dados e bibliotecas. Ou seja, não usa um sistema operacional existente. Um exemplo prático: em um ambiente comum, você precisaria instalar um sistema operacional para viabilizar o desenvolvimento ou o uso de uma aplicação. No *container* isso não é preciso. Trata-se de uma solução independente e, por isso, mais econômica.

O que é Docker?

O docker nada mais é do que uma plataforma open source que foi desenvolvida na linguagem de programação Google Go. Quem garante o bom funcionamento do docker é o Linux container, LXC, que é um sistema do kernel do Linux. Um docker permite a utilização de recursos isolados para desenvolvimento e teste de um software, sem a necessidade da criação de uma máquina virtual. Basicamente, o docker consegue trabalhar dentro de um servidor, isolando os recursos necessários, sem a necessidade de um sistema operacional exclusivo. Ou seja, ele utiliza as bibliotecas do sistema operacional do servidor em que está inserido.

Isso permite que o software seja executado em diferentes ambientes de desenvolvimento, pois os seus recursos são instalados no servidor e armazenados nos containers. O docker separa os recursos e compartilha as bibliotecas de kernel em comum com outros containers. Isso garante uma facilidade de adaptação do docker, assegurando a portabilidade, graças ao seu isolamento em qualquer servidor, sem a necessidade de um sistema operacional por módulo.