

Introdução

Introdução ao conceito de ter um servidor.

- O que é um servidor?
- Onde consigo um servidor?
- O que é a cloud?

O que é um servidor?

Antes de entender como manter um servidor, é preciso se perguntar: O que é um servidor?

De acordo com a **wikipedia**, um servidor "*é um software ou computador, com sistema de computação centralizada que fornece serviços a uma rede de computadores, chamada de cliente.*"
. Ou, de uma forma mais extensa:

A internet funciona na base de **protocolos**: padronizações de como enviar e receber dados entre computadores. Alguns protocolos famosos são:

- IP (Internet Protocol)
- TCP (Transmission Control Protocol)
- HTTP (HyperText Transfer Protocol)
- SMTP (Simple Mail Transfer Protocol)
- DNS (Domain Name System)

Entre muitos outros.

Todos os protocolos têm computadores como origem e destino. Geralmente, são computadores diferentes.

Logicamente, os dois computadores tem que estar acessíveis (conectados a rede) e disponíveis (ligados, prontos para receber mensagens) quando a comunicação for feita.

Ou seja: Se você quiser mandar uma mensagem de texto diretamente para um computador de um amigo, **o computador dele precisa estar ligado e conectado na internet**. Não parece a melhor ideia, parece?

(PS: protocolos que falam de cliente a cliente diretamente são chamados de protocolos peer-to-peer, ou p2p).

Logo, introduziu-se o conceito de servidores. Um servidor é um computador feito para estar conectado e disponível (quase) 100% do tempo. Assim, ao invés de computadores se comunicarem diretamente (*cliente a cliente*), os computadores se comunicam com um servidor.

Quem fica com a responsabilidade de manter a mensagem e ser consultado sobre mensagens novas é o servidor.

Nesse livro, você irá aprender a parte de *software* de um servidor. Desde quais sistemas operacionais escolher até a parte de automação de servidores em cloud.

Está pronto?

Onde consigo um servidor?

Para conseguir fazer um *setup* em um servidor, é preciso primeiro *ter* um servidor...

Felizmente, temos algumas opções na mesa:

- Servidores são computadores! Você pode fazer todos os tutoriais daqui em uma máquina virtual, ou mesmo na sua própria máquina (Não recomendado).
- Existem clouds que alugam servidores por um preço muito barato. Se você é estudante, todas as grandes clouds (como Amazon AWS, Google GCP, Microsoft Azure, DigitalOcean, entre outros) oferecem crédito grátis em suas nuvens. Se não é mais estudante, sem problema! Os preços são tão baratos quanto 5 dólares por mês.
- Caso aplicável, saiba também que o IMEsec tem um servidor físico no bloco A do IME.

Uma pergunta seguinte seria: *O que é a cloud?*

O que é a cloud?



Clouds são apenas um arranjo de computadores prontos para serem alugados de várias formas. O jeito mais simples de se vender um serviço cloud é simplesmente oferecer o aluguel do servidor crú, apenas com um sistema operacional instalado, e deixar quem está alugando fazer todo o trabalho de manter seus serviços. Essa forma de alugar servidores é chamada de **laaS, ou Infrastructure as a Service**. É exatamente o que queremos!

Mas, para completude de conversa, clouds podem ser vendidas de outras maneiras: em vez de oferecer o servidor crú, muitas clouds também oferecem um serviço de 'rodar a sua aplicação, independente de quantos servidores forem necessários'. Ou seja, você fornece o código, eles

rodam o código. A parte de como rodar o código fica a cargo da própria cloud. Essa forma de alugar servidores é comumente chamada de **PaaS, ou Platform As A Service**. Outro nome comum para isso é **Serverless**, dado que para você não existem servidores, só existe código rodando.

Outra maneira que irei citar de como vender uma cloud é o jeito da *iCloud*: Ela não oferece servidores, muito menos roda o seu código nas máquinas da Apple. Em vez disso tudo, o que a *iCloud* faz é vender serviços que rodam nos servidores deles, como por exemplo guardar imagens, vídeos e dados miscelâneos dos seus dispositivos Apple. Essa forma de vender servidores é chamada de **SaaS, ou Software as a Service**, e é a que impõe a maior quantidade de abstrações em cima do servidor. Para todo efeito, não é necessário nem saber o que é um servidor ou código!

