

A CONSTRUÇÃO DE UMA ARQUITETURA COM DOCKER: O CASO DO ROCKET.CHAT PARA O PROJETO AULA ABERTA

Thiago Tremenpol 1
Ibsem Agrello Dias 2

INTRODUÇÃO

As aplicações instaladas na nuvem, necessitam de requisitos de configuração que permitam sua escalabilidade e crescimento de forma rápida. A demanda por estas características ocorre por conta do crescimento exponencial de negócios virtuais. A startup é pensada em torno da inovação, com foco no cliente e para atingir esse cliente em escala mundial, logo velocidade e escalabilidade são requisitos não negociáveis em aplicações que atendam os modelos de negócio de startups. As tecnologias adotadas neste trabalho foram desenvolvidas principalmente para atender esses dois requisitos principais, escalabilidade e velocidade de instalação. Outros aspectos positivos podem ser enumerados, porém o destaque de como funciona uma arquitetura com a utilização de *docker* será em mostrar os passos e recursos de implantar uma aplicação com *docker*. O caso apresentado surge da demanda do projeto aula aberta que necessitava instalar um ambiente de *chatbot* em um espaço curto de tempo, acessível dentro e fora do ambiente do Cesusc, com disponibilidade 24 x 7 e que dependendo da demanda fosse viável escalar rapidamente a infraestrutura.

OBJETIVO

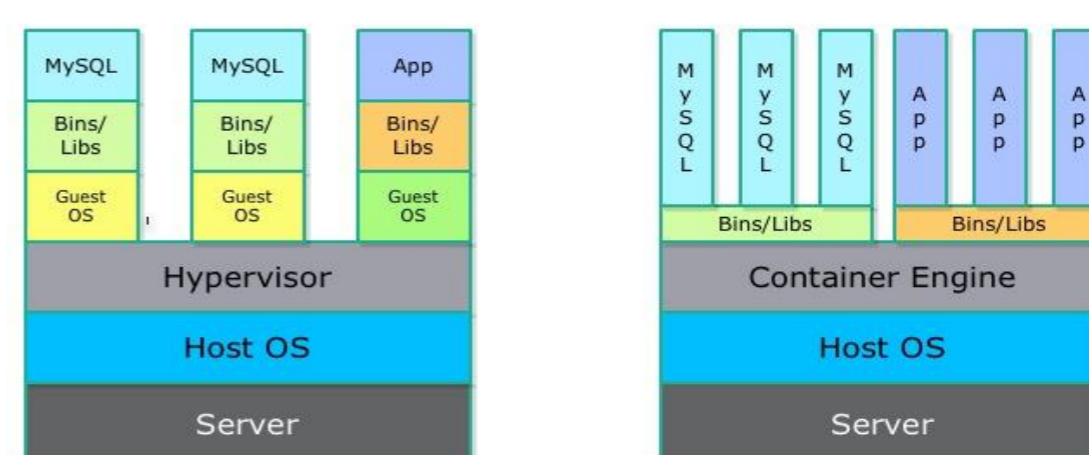
Este trabalho apresentará as principais características e passos de um processo de implementação de uma infra-estrutura de TI alocada na nuvem em uma instância AWS utilizando a tecnologia *Docker*. O objetivo principal é demonstrar os benefícios de adotar esta tipologia para instalar aplicações que tenham por requisito alta escalabilidade.

DESENVOLVIMENTO

O software *rocket.chat*^[1] é uma aplicação desenvolvida em javascript que tem por características a criação de equipes de trabalho com chats, recursos de compartilhamento de documentos, envio de emails, videoconferência, com destaque para o uso de bots, entre tantos outros. A instalação da aplicação necessita um conjunto mínimo de requisitos de infra-estrutura e a medida que seu número de usuários cresce e o aprendizado do uso de seus recursos acontece a necessidade por máquina e memória pode crescer de maneira exponencial. Estas características em conjunto com o tempo curto para instalação de 1 semana, levou a adoção do *docker*. Ao utilizar *docker*^[2] como plataforma facilitou-se o empacotamento, versionamento e imutabilidade do ambiente da aplicação, uma vez que algumas boas práticas são seguidas e as aplicações que rodam no *docker* são facilmente escaláveis sem nenhum tipo de *lock-in* com o *cloud provider*, ou seja é possível migrar para outro serviço de *cloud* ou data-center em questão de minutos.

A diferença entre as máquinas virtuais tradicionais e os *containers* com *docker* é que o ambiente de instalação no segundo empacota a aplicação permitindo a redundância sem existir problemas de acoplamento com as outras aplicações que fazem parte de solução completa. A Figura 1 demonstra a escalabilidade, ou seja a inserção de novos *containers* com *MySQL* e da *APP* na imagem de forma padronizada, já no lado esquerdo da figura percebe-se que existe acoplamentos entre a *APP* e o *MySQL* e as bibliotecas de sistemas e o próprio sistema operacional.

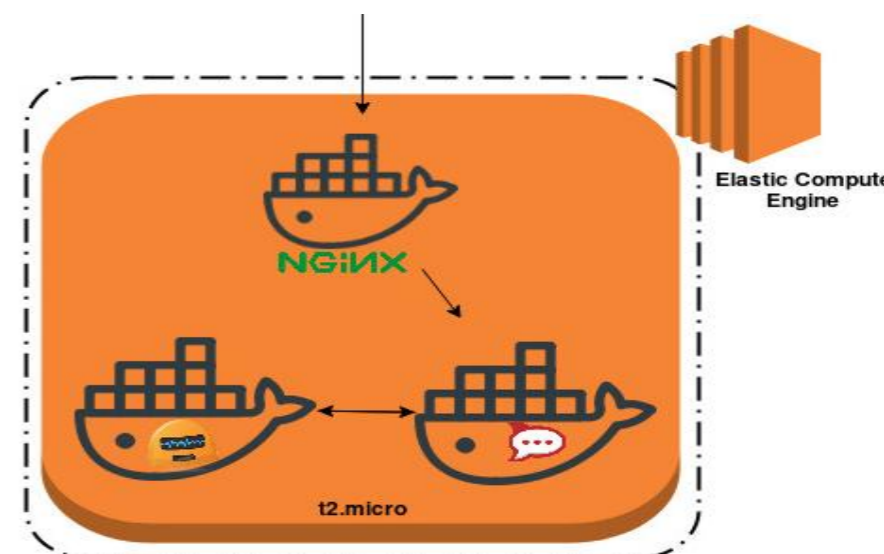
Figura 1 – Diferença entre Máquina Virtual e *Docker*



Fonte: elaborado pelos autores.

Partindo deste tipo de infraestrutura a configuração realizada para a aplicação Rocketchat e de bot foi disponibilizada. A Figura 2 mostra a disposição destas aplicações acrescidas de uma terceira aplicação denominada *NGiNX* que foi utilizado para a configuração de proxy.

Figura 2 – Configuração da arquitetura do *Rocket.chat* com *Docker* na AWS



Fonte: elaborado pelos autores.

CONSIDERAÇÕES FINAIS

Este modelo de arquitetura permite inserir novas aplicações no mesmo ambiente sem afetar as já existentes. Os recursos como memória e processamento são compartilhados de acordo com a demandas das aplicações, e caso exista necessidade de escalar os recursos ou trocar de ambiente não há necessidade de configurar novamente cada aplicação pois estas estão empacotadas em um *docker* (representado pela baleia) e migram facilmente para outro ambiente. Logo o desenvolvimento de aplicações que atendem requisitos de soluções para startups, bem como grandes corporações que necessitam de escalabilidade e fácil acoplamento tem a por prudência implantar essas aplicações em um ambiente com *docker*.

1 Graduando em Análise e Desenvolvimento de Sistemas. Faculdade Cesusc. E-mail: thiagotrennepohl@gmail.com
2 Mestre em Administração. Faculdade Cesusc. E-mail: ibsem.dias@gmail.com

REFERÊNCIAS BIBLIOGRÁFICAS

[1] Rocket.chat. Extraído da página: <https://rocket.chat/docs/>

[2] Merkel, D. (2014). Docker: lightweight linux containers for consistent development and deployment. *Linux Journal*, 2014(239), 2.