



Utilizando o Rocket.Chat em container



Uma história sobre como um serviço Rocket Chat foi implementado sobre containers, os cuidados que não foram tomados, os riscos, os benefícios e as boas práticas da documentação do serviço.



Fabrício Machado

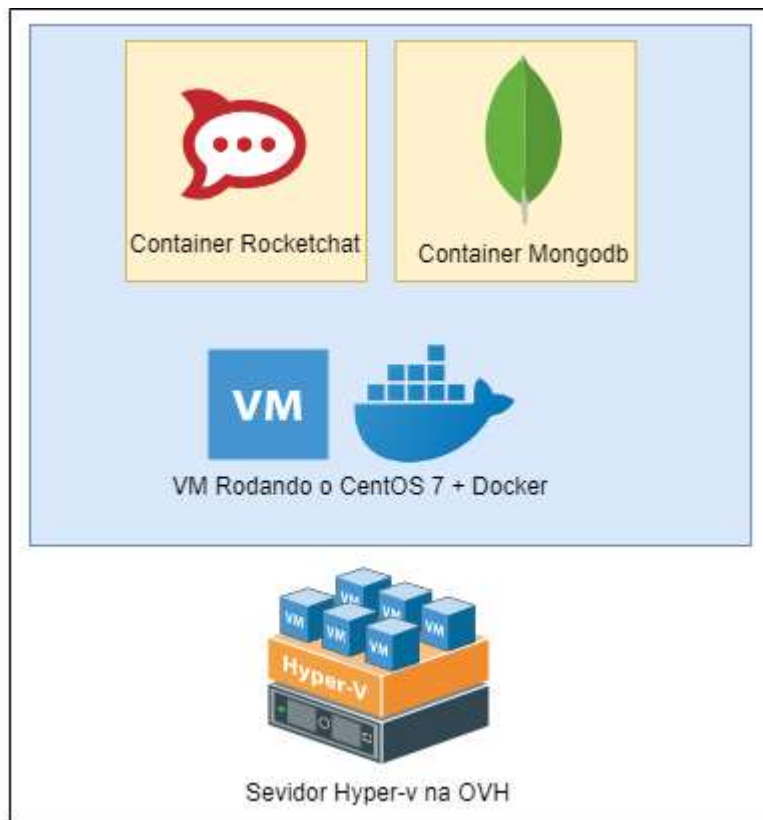
- Formando em Redes de Computadores.
- Atuando na área de TI por 19 anos.
- Atuou em empresas que prestaram serviços para Terra e Dell, entre outras empresas.
- Atualmente é Sysadmin na InterOp.
- Entusiasta de DevOps e participante dos meetups de docker.
- Proprietário de um Raspberry PI :D



Solicitação: Implantar o Rocket.chat para os colaboradores da empresa.



Infraestrutura:



Como Fazer?

Usando docker e criando container era o jeito mais rápido de implementar o serviço e ficar liberado para atender os chamados de projeto dos nossos clientes.





Como implementar isso da maneira errada:

- https://hub.docker.com/_/rocket-chat

No docker hub existe apenas a orientação para colocar os containers no ar.

```
docker run --name db -d mongo:3.0 --smallfiles
```

```
docker run --name rocketchat -p 80:3000 --env ROOT_URL=http://localhost --link db -d rocket.chat
```

E assim foi feito.



E os meses se passaram ...





**Então recebemos uma solicitação para
atualizar a versão do Rocket.chat**





Nesse momento de atualizar o container nos deparamos com uma situação complicada.





Afinal qual era o problema e o risco?



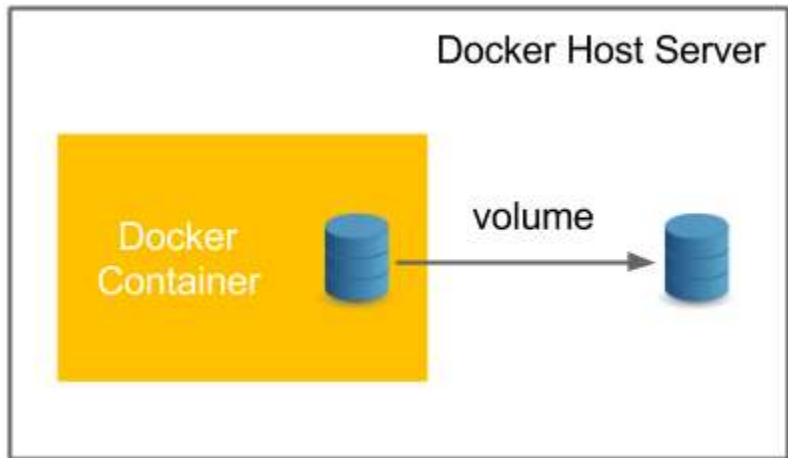


**Existiam ali nos containers 6 meses de
conversas, arquivos e imagens dos
colaboradores!**



Alguém esqueceu os volumes!

Caso alguém encerre o container, os dados das conversas, imagens, banco de dados, configurações do serviços, **TODAS** deixariam de existir.





Bom sendo assim, como fazer o certo?

Segundo a documentação da própria Rocket.chat (<https://rocket.chat/docs/installation/docker-containers/>), no docker-compose estão previstos 2 volumes: Um para o banco e um para o container do serviço.

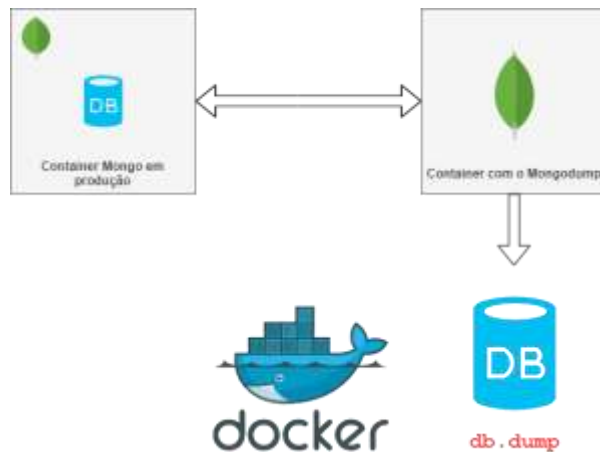
```
rocketchat:
  image: rocket.chat:latest
  command: bash -c 'for i in `seq 1 30`; do node main.js && s=$$? && break'
  restart: unless-stopped
  volumes:
    - ./uploads:/app/uploads
```

```
mongo:
  image: mongo:4.0
  restart: unless-stopped
  volumes:
    - ./data/db:/data/db
    - ./data/dump:/dump
```

Felizmente, nesse caso todos se salvaram :D

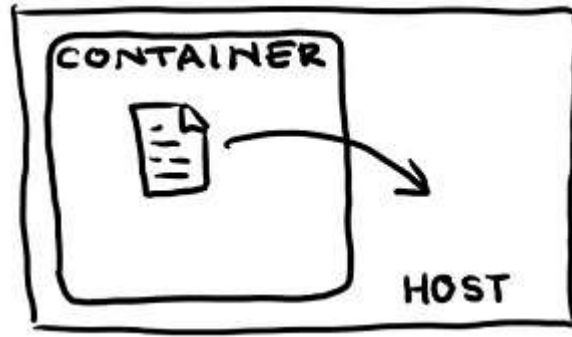
Efetuamos um dump do banco do mongo usando um outro container com um volume mapeado.

(https://jeromejaglaire.com/doc/programming/mongodb_docker_mongodump_mongorestore)



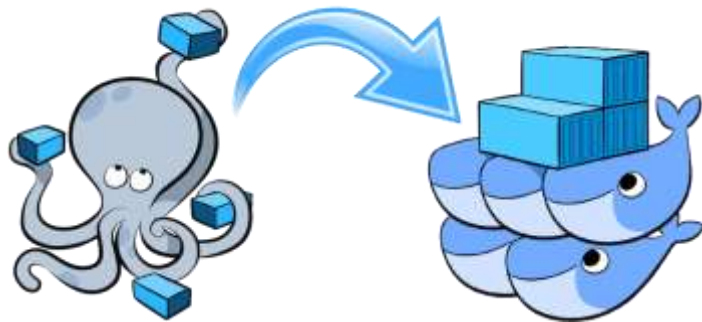
Felizmente, nesse caso todos se salvaram :D

Depois copiamos os dados do container do Rocket.chat usando o `docker cp`



Felizmente, nesse caso todos se salvaram :D

Criamos um docker-compose.yaml file para, quando for necessário, atualizar a versão do Rocket.chat no container não seja uma novela novamente!





Demo



Perguntas?



Obrigado!

Contatos:

