

Docker Networking

DigitalHouse >
Coding School



**Certified Tech
Developer**
The Ultimate Degree

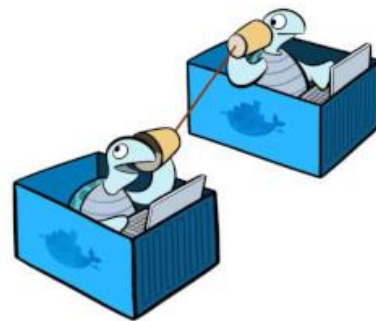
Índice

1. [Introdução às redes Docker](#)
2. [Bridge Driver em ação](#)
3. [Resolução de nomes](#)

1 | Introdução às redes Docker

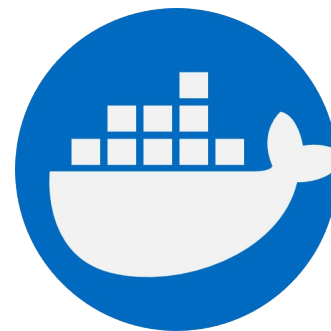
Introdução às redes Docker

Vamos nos concentrar no driver chamado bridge, pois com a nova abordagem do Docker voltada para habilitar os desenvolvedores, é aquele que a maioria usará em seus computadores. Todos os outros drivers são voltados para ambientes produtivos, mas essa área hoje é dominada pelo Kubernetes, que possui um modelo de Networking completamente diferente.



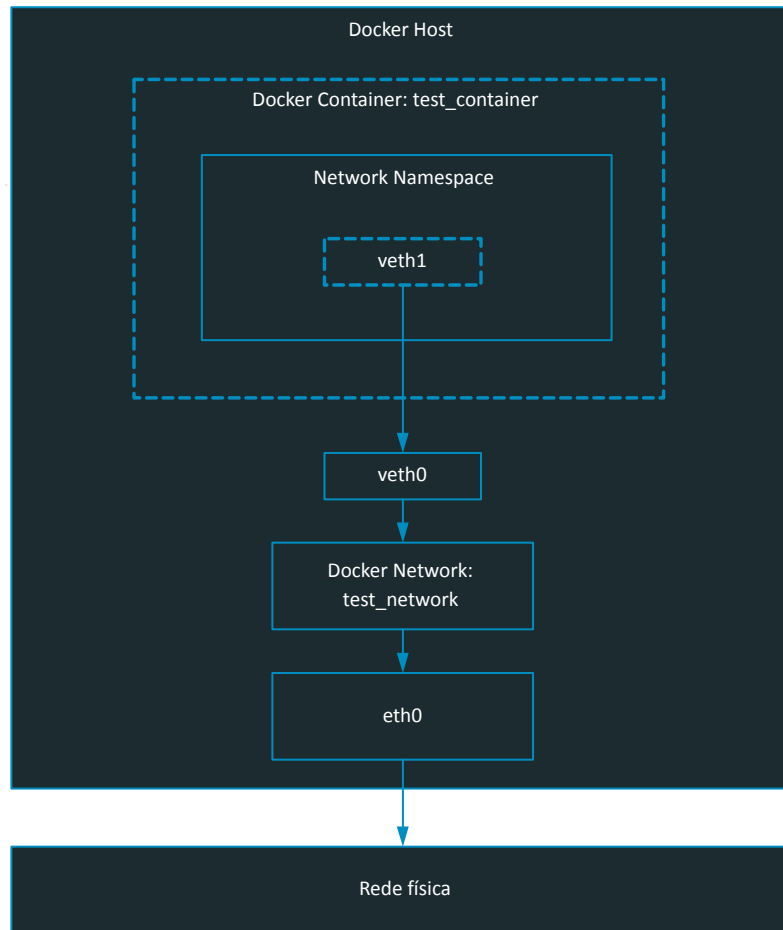
Introdução às redes Docker

- As redes do tipo bridge são locais e exclusivas do host onde foram criadas.
- Eles são o tipo de rede padrão no Docker.
- Simula a criação de switches ou hubs, nível 2 (no modelo OSI). Podemos usar ferramentas como 'brctl' no Linux para ver o funcionamento interno.
- Após a instalação do Docker, uma rede do tipo bridge chamada "bridge" é criada por padrão. As boas práticas indicam que isso não deve ser usado e em vez disso, novas redes devem ser criadas para usos específicos.



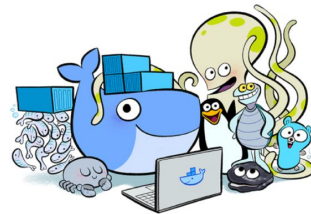
2 | Bridge Driver em ação

Bridge Driver em ação



3 | Resolução de nomes

Resolução de nomes



Os IPs são efêmeros e ainda mais no mundo dos containers – eles também deveriam ser. Portanto, precisamos de uma maneira que permita que um container converse com outro sem necessariamente saber o IP que foi atribuído a ele.

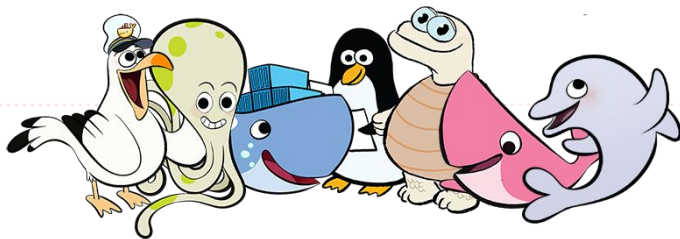
O que o Docker faz por nós?

Para todas as redes que criamos (as redes criadas por padrão não nos fornecem essa vantagem) e para todos os containers deliberadamente nomeados usando `--name` no `docker create` — conectados a eles, o Docker nos diz que fornece um serviço de resolução de nomes dentro do escopo da própria rede, usando o nome do container para identificá-lo por meio do serviço DNS.

Resolução de nomes

docker network

- **create:** nos permite criar uma rede.
 - **--internal:** Ampliando esta opção a rede criada será do tipo 'privada'.
- **connect:** nos permite conectar um container existente a uma rede.



docker run

- **--network <networkName>:** nome da rede à qual vamos conectar o container.
- **--name:** nome do container, necessário para que a resolução de nomes funcione.
- **-p <hostPort>:<containerPort>:** Ele nos permite publicar certas portas de um container no host para poder acessar o serviço dentro do container.
- **-P:** nos permite publicar todas as portas definidas na especificação do container (Dockerfile) em portas aleatórias do host.

Onde encontrar mais informação?



- Documentação do Docker:
 - criação de rede docker
 - Visão geral da rede
 - Rede com contêineres autônomos



Blogs:

[How Docker Container Networking Works - Mimic It Using Linux Network Namespaces](#)



Livro:

[Poulton, N. \(2017\). *Docker Deep Dive*.](#)

DigitalHouse>
Coding School