

## Laboratório 1 - DNS

Aluno: Gabriel Davi Silva Pereira

Matrícula: 170010341

Pré-requisitos:

- Docker

Configuração:

- Para a criação de uma rede compartilhada pelos containers, foi executado o seguinte comando:
  - **docker network create --subnet=192.168.10.0/16 frc\_net**
    - criação de uma network com range de 16 valores de ip chamado frc\_net
- Criação do container 1:
  - **docker run -it --name container1 --network frc\_net --ip 192.168.10.1 alpine ash**
    - Cria-se um container com a imagem linux alpine, conectado com a rede frc com um ip estático de 192.168.10.1
- Criação do container 2:
  - **docker run -it --name container2 --network frc\_net --ip 192.168.10.2 alpine ash**
    - Cria-se um container com a imagem linux alpine, conectado com a rede frc com um ip estático de 192.168.10.2

Hosts:

- Container 1

```
/ # cat /etc/hostnames
cat: can't open '/etc/hostnames': No such file or directory
/ # cat /etc/hosts
127.0.0.1      localhost
::1          localhost ip6-localhost ip6-loopback
fe00::0       ip6-localnet
ff00::0       ip6-mcastprefix
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
192.168.10.1   c2132b81db7e
/ #
```

Observa-se que para o container 1, o docker mapeia o ip para um id que lhe é atribuído no momento de sua criação, no caso c2132b81db7e

- Container 2

```
/ # cat /etc/hosts
127.0.0.1      localhost
::1           localhost ip6-localhost ip6-loopback
fe00::0       ip6-localnet
ff00::0       ip6-mcastprefix
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
192.168.10.2   2e1722c24a28
```

Observa-se que para o container 2, o docker mapeia o ip para um id que lhe é atribuído no momento de sua criação, no caso 2e1722c24a28

Pingando container 2 dentro do container 1:

```
/ # ping 2e1722c24a28
PING 2e1722c24a28 (192.168.10.2): 56 data bytes
64 bytes from 192.168.10.2: seq=0 ttl=64 time=0.745 ms
64 bytes from 192.168.10.2: seq=1 ttl=64 time=0.314 ms
64 bytes from 192.168.10.2: seq=2 ttl=64 time=0.358 ms
64 bytes from 192.168.10.2: seq=3 ttl=64 time=0.469 ms
64 bytes from 192.168.10.2: seq=4 ttl=64 time=0.402 ms
64 bytes from 192.168.10.2: seq=5 ttl=64 time=0.403 ms
64 bytes from 192.168.10.2: seq=6 ttl=64 time=0.361 ms
64 bytes from 192.168.10.2: seq=7 ttl=64 time=0.393 ms
```

Observa-se que ao utilizar-se o host configurado, o comando ping retorna positivo.

Pingando container 1 dentro do container 2:

```
PING c2132b81db7e (192.168.10.1): 56 data bytes
64 bytes from 192.168.10.1: seq=0 ttl=64 time=1.085 ms
64 bytes from 192.168.10.1: seq=1 ttl=64 time=0.321 ms
64 bytes from 192.168.10.1: seq=2 ttl=64 time=0.363 ms
64 bytes from 192.168.10.1: seq=3 ttl=64 time=0.511 ms
64 bytes from 192.168.10.1: seq=4 ttl=64 time=0.335 ms
64 bytes from 192.168.10.1: seq=5 ttl=64 time=0.356 ms
64 bytes from 192.168.10.1: seq=6 ttl=64 time=0.402 ms
64 bytes from 192.168.10.1: seq=7 ttl=64 time=0.395 ms
```

Servidor DNS:

Para o uso do docker para configuração, foi aliado os conhecimentos passados por esse tutorial <https://fabiotaavarespr.dev/posts/configurar-dns-bind9-com-docker/> com os passos dados pelo exercício.

O docker já possui um serviço de DNS que mapeia os ips do containers para seu id. Então para poder configurar o dns de um container, foi usada a imagem **labbsr0x/docker-dns-bind9**.

Seguindo o passo a passo do tutorial, com o que foi passado no exercício, foi gerado o seguinte resultado:

<https://github.com/GabrielDVpereira/FRC/tree/main/lab01/bind9>

Questões de pesquisa:

1) Qual é o retorno do comando `nslookup`? O que significa?

R: Mostra o nome do domínio de seu sistema

2) O que é o nome `localhost`? E o endereço `127.0.0.1` dado a ele no arquivo `/etc/hosts`? Por que deve sempre existir este endereço e nome em sistemas UNIX/Linux?

R: Localhost é o computador que está executando um programa. Seu endereço é atribuído por padrão `127.0.0.1`, onde `127` é o valor atribuídos por endereços loopback. O localhost tem propósitos importantes para fim de testes de programa e velocidade.

3) O que é o FQDN?

R: Nome de domínio que especifica sua localização exata na árvore hierárquica do DNS.

4) Podemos ter 2 servidores DNS na mesma rede? Qual é a configuração mais adequada para esta situação?

R: Sim, haverá um servidor primário e um secundário. Com isso, caso a execução do servidor primário falhar, será feita a busca no servidor secundário.

5) O que é DNS reverso? Como isto foi implementado no lab?

R: No DNS reverso, consulta-se um dado host, passando-se seu ip. Em nosso lab, criou-se o arquivo db.warsstar que mapeou o range de valores associados a um host para seu dado nome para seu host

6) O que é a entrada MX inserida no domínio? Podem haver mais de uma?

R: Indica que o servidor está esperando dados de email.

7) O que é resposta autoritativa dada por um servidor DNS? Explique.

R: São informações que um servidor DNS fornece sobre seu domínio.

8) O que é um servidor caching-only?

R: Um servidor caching-only, em resposta à solicitação de um cliente, encaminha uma consulta a servidores DNS na Internet e, uma vez obtida a resposta, armazena estas informações localmente para consultas futuras.