

# Installation/Configuration de notre serveur DNS

## 1. Création d'un réseau Docker :

Dans notre réseau les serveurs sont séparés des clients (Question sécurité) , c'est pour cela que j'ai créé un réseau pour mettre les serveurs.

Pour ce faire :

la commande **docker network** est utilisée pour la création de réseaux. Voici la commande que j'ai utilisée :

```
docker network create --driver bridge --subnet 172.16.0.0/24 reseau-interne-serveur
```

create : permet la création d'un réseau.

--driver bridge : est un type qui permet à plusieurs conteneurs sur le même réseau de pouvoir communiquer.

--subnet : l'adresse IP que vous voulez donner au réseau ainsi que son masque.

**Reseau-interne-serveur : le nom du réseau**

## 2. Création d'une image via un dockerfile :

Au lieu d'utiliser une image toute faite j'ai préféré créer une image qui permettra la création de mon serveur (Construction).

docker build -t nom de l'image " .

⑩ -t : Nom et éventuellement une balise au format 'nom: balise'

⑩ . : le point représente le dockerfile qui contient les instructions de constructions.

```
#Notre image sera issue d'une image ubuntu
FROM ubuntu:bionic

# vérification de MAJ, installation de MAJ, installations bind9 sur notre image
RUN apt update && apt upgrade -y && apt install bind9 bind9utils -y

#Copie les fichiers de configuration pré-configurer

COPY named.conf /etc/bind/
COPY named.conf.options /etc/bind/
COPY named.conf.default_zones /etc/bind/
COPY named.conf.local /etc/bind/

COPY db.tl1-3.ephec-tl.be /etc/bind/zones/

#Ouvrir le port 53 en UDP/TCP
EXPOSE 53/udp 53/tcp
|
#commande qui permet au serveur la lecture de la configuration BIND9
CMD named -g
```

## 3. création du serveur

BEN-TAHRI MERWANE

Ici je vais utiliser un container comme serveur DNS.

La commande `docker run` commence par créer une couche de conteneur

`docker run -tid --network reseau-interne-serveur --ip 172.16.0.4 --name serveur-dns serveur-dns`

- ⑩ `-tid` : `-d` → lancer le conteneur en arrière plan, `-i` → Gardez STDIN ouvert même s'il n'est pas attaché, `-t` → Attribuer un pseudo-TTY.
- ⑩ `--network` : ajouter le conteneur dans le réseau défini
- ⑩ `--ip` : permet de lui attribuer une adresse IP fixe
- ⑩ `--name` : le nom que vous voulez donner à votre image
- ⑩ `serveur-dns` : le dernier `serveur-dns` est le nom que j'ai donné à mon image.

Enfin une fois la construction de votre réseau nous allons tester la configuration du serveur.

- `Docker exec ti nom-conteneur bash` : cela va me permettre d'accéder à l'invite de commande dans mon conteneur.
- Vérifier que les fichiers ont bien été copiés dans votre `/etc/bind` comme décrits dans la *dockerfile*
- installer le paquet `dnsutils` pour pouvoir utiliser la commande **dig** qui permet d'effectuer des tests .