## Container Nginx-Proxy et Letsencrypt-Compagnon

Container servant de proxy web à tous les autres containers nécessitant une interface web accessible.

## Forcer la demande d'un certificat

- Lorsque le site web d'un container persiste a ne pas s'afficher en HTTPS
- Se connecter au container nginx-proxy-letsencryp : docker exec -it nginx-proxy-letsencrypt /bin/bash
- Dans le dossier /app/ lancer la commande suivante en adaptant votre domaine :
  - ∘ Ex.:./letsencrypt\_service example.silene.eu,www.example.silene.eu
  - Note : plusieurs domaines ou sous-domaines peuvent être indiqué en les séparant par une virgule. Le premier sera le principal.
  - ATTENTION : les domaines avec wildcard "\*" ne sont pas encore supporté.

## Partager les logs Nginx avec l'hôte



Une amélioration possible serait de créer un utilisateur nginx ou www-data sur l'hôte puis d'utiliser les identifiant de cet utilisateur dans le container nginx-proxy. Cela permettrait d'avoir des droits cohérents entre le container et l'hôte pour les fichiers de logs.

- Pourquoi?
  - Afin de pouvoir faire fonctionner correctement le service Fail2ban de l'hôte qui se base sur les logs Nginx
- Problèmes :
  - transférer les IP des clients dans le container Nginx et éviter de voir dans les logs Nginx
     l'IP de la Gateway du réseau Docker
  - activer correctement Logrotate pour que Nginx présent dans le container gère correctement la création de nouveaux fichiers de log : propriété correcte et écriture des logs dans le nouveau fichier.
- Sur l'hôte :
- Créer un dossier /var/log/nginx/ avec :

mkdir /var/log/nginx/

Donner les bons droits :

chown root:adm /var/log/nginx/ ; chmod 640 /var/log/nginx/

Créer les fichiers par défaut avec :

## touch /var/log/nginx/access.log /var/log/nginx/error.log

- Donner les bons droits aux fichiers de base. C'est important car nous utilisons copytruncate dans la conf Logrotate:
  - Se connecter au service nginx-proxy :

```
docker exec -it nginx-proxy bash
```

- Regarder l'id de l'utilisateur Nginx avec la commande id.
- Puis modifier la propriété des fichiers de log :

```
chown 101:adm *.log
```

• Ajouter le paquet logrotate avec :

```
apt install logrotate
```

• Créer un nouveau fichier de conf logrotate :

```
touch /etc/logrotate.d/nginx
```

• Y placer le contenu suivant (vérifier les chemins utilisés!) :

```
/var/log/nginx/*.log {
    daily
    missingok
    rotate 400
    compress
    delaycompress
    notifempty
    su root adm
    # User "nginx" in container has an id of 101
    #create 0640 101 adm
    # To avoid permission issues with directory and files shared
between host and container: see copytruncate
    copytruncate
    dateext
    dateyesterday
    dateformat .%Y-%m-%d
    sharedscripts
    postrotate
        cd /home/admin/docker/proxy/ \
        && /usr/bin/docker compose kill -s USR1 nginx-proxy
    endscript
```

- Modifier le fichier docker-compose.yml de la stack proxy dans /home/admin/docker/ en tant qu'utilisateur admin :
  - le service nginx-proxy doit contenir :

```
network_mode: "host"
```

- Le mode "hôte" permet d'obtenir les IP réelles des clients dans les logs Nginx avec la variable \$remote addr
- commenter la section ports: sinon l'erreur! nginx-proxy Published ports are discarded when using host network mode apparaîtra.
- Ajouter dans la section volumes : l'entrée suivante :

```
volumes:
```

- /var/log/nginx/:/var/log/nginx/
- Modifier le fichier nginx.conf avec :

• Relancer la stack proxy avec :

```
docker compose down && docker compose up -d
```

- Vérifier que les logs sont écris dans le fichier /var/log/nginx/access.log
- Tester la config Logrotate avec :

```
logrotate -f /etc/logrotate.d/nginx
```

From:

https://wiki-sinp.cbn-alpin.fr/ - CBNA SINP

Permanent link:

https://wiki-sinp.cbn-alpin.fr/serveurs/installation/bkp-srv/docker-nginx-proxy

Last update: 2025/10/15 08:28

