

# Installer et configurer Nginx

- **Ressources :**

- [Total Nginx monitoring, with application performance and a bit more, using Telegraf/InfluxDB/Grafana.](#)

## Installation

- Installer *Nginx* et *Certbot* : `sudo aptitude install supervisor nginx python3-certbot-nginx python3-certbot-dns-ovh`
- Activer le redémarrage automatique du service : `systemctl enable nginx.service`
- Activer le démarrage automatique du service Systemd lançant deux fois par jour certbot renew : `systemctl enable certbot.timer`
  - Vérifier le status et démarrer le service si nécessaire : `systemctl status certbot.timer`

## Status de Nginx

Installer le point d'entrée permettant d'accéder au status de Nginx :

- Vérifier que Nginx est compilé avec le support du module Status : `nginx -V 2>&1 | grep -o with-http_stub_status_module`
- Créer une nouvelle conf `vi /etc/nginx/conf.d/status.conf` avec le contenu :

```
server {
    listen 9090;

    location /nginx_status {
        stub_status on;

        access_log off;
        allow 127.0.0.1;
        # Autoriser le réseau Docker :
        allow 172.18.5.0/24;
        deny all;
    }
}
```

- Prendre en compte la modification : `nginx -t && nginx -s reload`
- Vérifier que cela fonctionne : `curl 127.0.0.1:9090/nginx_status`

## Maintenir les logs web sur 1 an

Configure les logs avec maintient sur 1 an (obligation légale) :

- Éditer le fichier de config logrotate de Nginx : `vi /etc/logrotate.d/nginx`
  - y remplacer :
    - `rotate 14` par `rotate 400`
  - y ajouter les 3 lignes suivantes :

```
dateext
dateyesterday
dateformat .%Y-%m-%d
```

- Éditer le fichier `crontab` pour lancer les scripts présents dans `crond.daily` à minuit : `vi /etc/crontab`
  - la ligne pour `cron.daily` doit débuter par `0 0` (par défaut, c'est `25 6`)

## Modification du format des logs

Modifier les logs d'accès (ajout d'infos) pour Telegraf et GoAccess :

- Éditer la conf de Nginx `vi /etc/nginx/nginx.conf` et ajouter dans la section Log :

```
# Enabling request time
log_format enhanced-fmt '$remote_addr - $remote_user [$time_local] '
    '$request' $status $body_bytes_sent '
    '$http_referer' '$http_user_agent' '
    'rt=$request_time uct="$upstream_connect_time"
    uht="$upstream_header_time" urt="$upstream_response_time" '
    'gZR="$gzip_ratio" ';

access_log /var/log/nginx/access.log enhanced-fmt;
```

## Configuration de la compression

Activer la compression Gzip du contenu renvoyé par Nginx pour tous les types Mime (JS, CSS...) :

- Ressources :
  - [server-configs-nginx : compression.conf](#)
  - [Nginx Admin Guide : Compression and Decompression](#)
  - [Nginx : Module ngx\\_http\\_gzip\\_module](#)
- Éditer la conf de Nginx `vi /etc/nginx/nginx.conf` et remplacer la section Gzip (qui ne contient que `gzip on;`) par :

```
#+-----+
# Gzip Settings

gzip on;
gzip_comp_level 5;
gzip_min_length 256;
gzip_proxied any;
```

```
gzip_vary on;
gzip_types
    application/atom+xml
    application/geo+json
    application/javascript
    application/x-javascript
    application/json
    application/ld+json
    application/manifest+json
    application/rdf+xml
    application/rss+xml
    application/vnd.ms-fontobject
    application/wasm
    application/x-web-app-manifest+json
    application/xhtml+xml
    application/xml
    font/eot
    font/otf
    font/ttf
    image/bmp
    image/svg+xml
    text/cache-manifest
    text/calendar
    text/css
    text/javascript
    text/markdown
    text/plain
    text/xml
    text/vcard
    text/vnd.rim.location.xloc
    text/vtt
    text/x-component
    text/x-cross-domain-policy;
```

## Ajouter le support de Geolp

- Vérifier le support de Geolp par Nginx : `nginx -V 2>&1 | grep -o with-http_geoip_module`
- GeolP n'est plus maintenu ⇒ geoip2 dont le module et le support dans Nginx nécessite [la compilation du module](#)
  - À voir plus tard...

## Ajouter le support des fichiers d'authentification HTTP

- Installer le paquet suivant : `aptitude install apache2-utils`
- Pour créer un fichier `.htpasswd` : `htpasswd -c /etc/nginx/.htpasswd <user-name-1>`
- Ensuite, pour ajouter des utilisateurs (sans l'option `-c`) : `htpasswd /etc/nginx/.htpasswd`

<user-name-2>

- Pour limiter l'accès, utilise ensuite les directions suivantes dans une section server ou location :

```
auth_basic "Zone à accès restreint";  
auth_basic_user_file /etc/nginx/.htpasswd;
```

## Installer les scripts d'activation/désactivation des sites Nginx

- Nous utiliserons les scripts mis à disposition par ce dépôt : [perusio/nginx\\_ensite](https://github.com/perusio/nginx_ensite)
- Suivre l'installation automatique indiquée :
  - Se connecter en tant qu'admin : `ssh admin@web-paca-sinp`
  - Se place dans le dossier de téléchargement : `cd ~/dwl`
  - Cloner le dépôt : `git clone https://github.com/perusio/nginx_ensite.git`
  - Aller dans le dossier cloné : `cd nginx_ensite`
  - Lancer l'installation automatique : `sudo make install`
- Vérifier le fonctionnement des 2 nouvelles commandes : `nginx_dissite` et `nginx_ensite`
  - Penser à recharger Nginx : `sudo service nginx reload`

## Activer les prisons Fail2ban pour Nginx

- Penser à décommenter les prisons liées à Nginx présentes dans le fichier : `vi /etc/fail2ban/jail.d/defaults-debian.conf`

## Éviter l'erreur "bind() to 172.18.5.1:9090 failed (99: Cannot assign requested address)"

- Au redémarrage de la machine, il arrive que Docker ne soit pas complètement lancé. Cela provoque l'erreur : `bind() to 172.18.5.1:9090 failed (99: Cannot assign requested address)` et cela empêche Nginx de démarrer. Il faut donc le lancer manuellement : `systemctl start nginx`. L'erreur était due au fichier `/etc/nginx/conf.d/status.conf` qui contenait une ligne `listen 172.18.5.1:9090;`. Cette ligne n'est finalement pas utile car il suffit d'écouter sur le port 9090 avec la commande `listen 9090;` ; les paramètres `allow <...>` ; suffisent à limiter l'accès. Le port est bien accessible sur 127.0.0.1 comme sur 172.18.5.1 (pour un accès dans un container Docker).
- Pour éviter ce problème, nous avons modifier le fichier `/etc/nginx/conf.d/status.conf` comme indiqué précédemment. Deplus, nous avons modifié le script Systemd de Nginx : `/lib/systemd/system/nginx.service`
- Afin d'éviter que les modifications effectuées dans le fichier `/lib/systemd/system/nginx.service` soient écrasées à chaque mise à jour de Nginx, vous devez ajouter un fichier qui surchargera les valeurs par défaut.
  - **Source** : [Using systemd to control the Docker daemon](#)
- Pour créer automatiquement l'arborescence de dossier et le fichier nécessaire, utiliser la

commande suivante : `systemctl edit nginx`

- Les modifications devraient être présente dans le fichier suivant : `vi /etc/systemd/system/nginx.service.d/override.conf`
- Ajouter dans le nouveau fichier vide ceci :

```
[Unit]
Description=The nginx HTTP and reverse proxy server (overridden)
After=network.target remote-fs.target nss-lookup.target network-online.target docker.service
Wants=network-online.target
```

▪ **Notes :**

- l'indication `network-online.target` permet à Nginx d'attendre que le réseau soit démarré.
- l'indication `docker.service` dans `After=...` indique à Nginx que le service Docker doit être démarré.
- Sortez de l'édition du fichier en sauvegardant
- Lancer la prise en compte des modifications qui vérifiera une éventuelle erreur : `systemctl daemon-reload`
- Relancer le service Docker : `systemctl restart nginx`
- Vérifier la présence du texte (*overridden*) dans la description du service : `systemctl status nginx`
- Redémarrer la machine, attendre son redémarrage, s'y reconnecter et s'assurer que Nginx est bien démarré : `systemctl status nginx`

From:

<http://wiki-sinp.cbn-alpin.fr/> - **CBNA SINP**

Permanent link:

<http://wiki-sinp.cbn-alpin.fr/serveurs/installation/web-srv/nginx?rev=1616583962>

Last update: **2021/03/24 11:06**

