

# NetAlertX : surveillance du réseau

Dans ce tutoriel, nous allons apprendre à installer l'application NetAlertX (Pi.Alert) sur un NAS Synology. L'installation sera effectuée dans un conteneur Docker, via Container Manager.

NetAlertX est une application gratuite et open source dont l'objectif est de surveiller votre réseau local dans le but de référencer tous les appareils connectés. Que ce soit un **appareil connu ou inconnu** (intrusion ?), il sera détecté par NetAlertX. Ainsi, vous pouvez **recevoir une notification** lorsqu'**un nouvel appareil se connecte pour la première fois** ou **si un appareil tombe en panne** (connecté puis déconnecté).

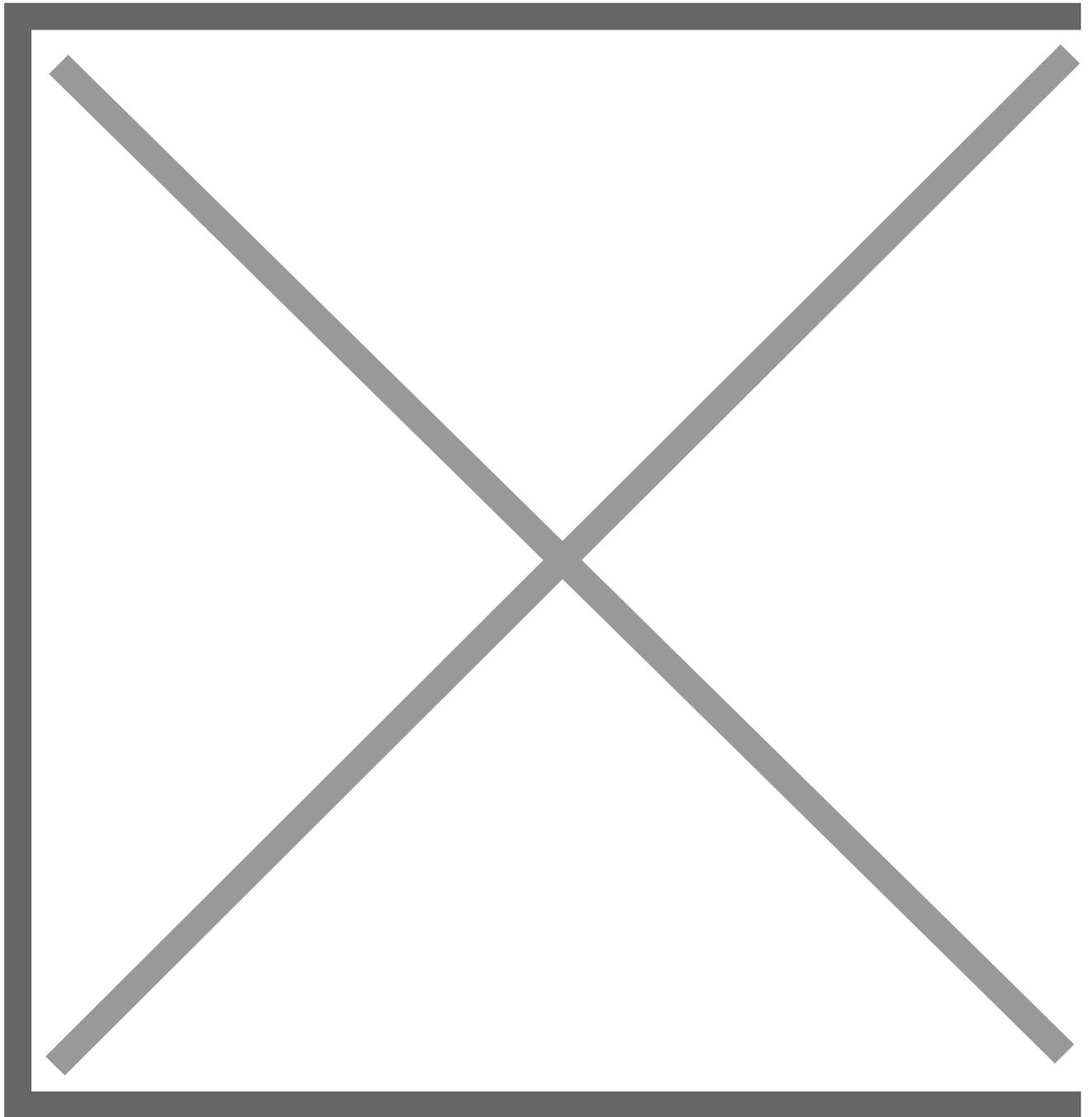
## Déployer NetAlertX avec Docker

Avant toute chose, vous devez installer l'application "**Container Manager**" sur votre NAS, si ce n'est pas déjà fait. Puis, vous devez créer un dossier pour ce conteneur. Pour ma part, il s'agit du répertoire "**netaalertx**" créé sous "**docker**".

Dans ce répertoire nouvellement créé, vous devez créer les trois sous-répertoires suivants :

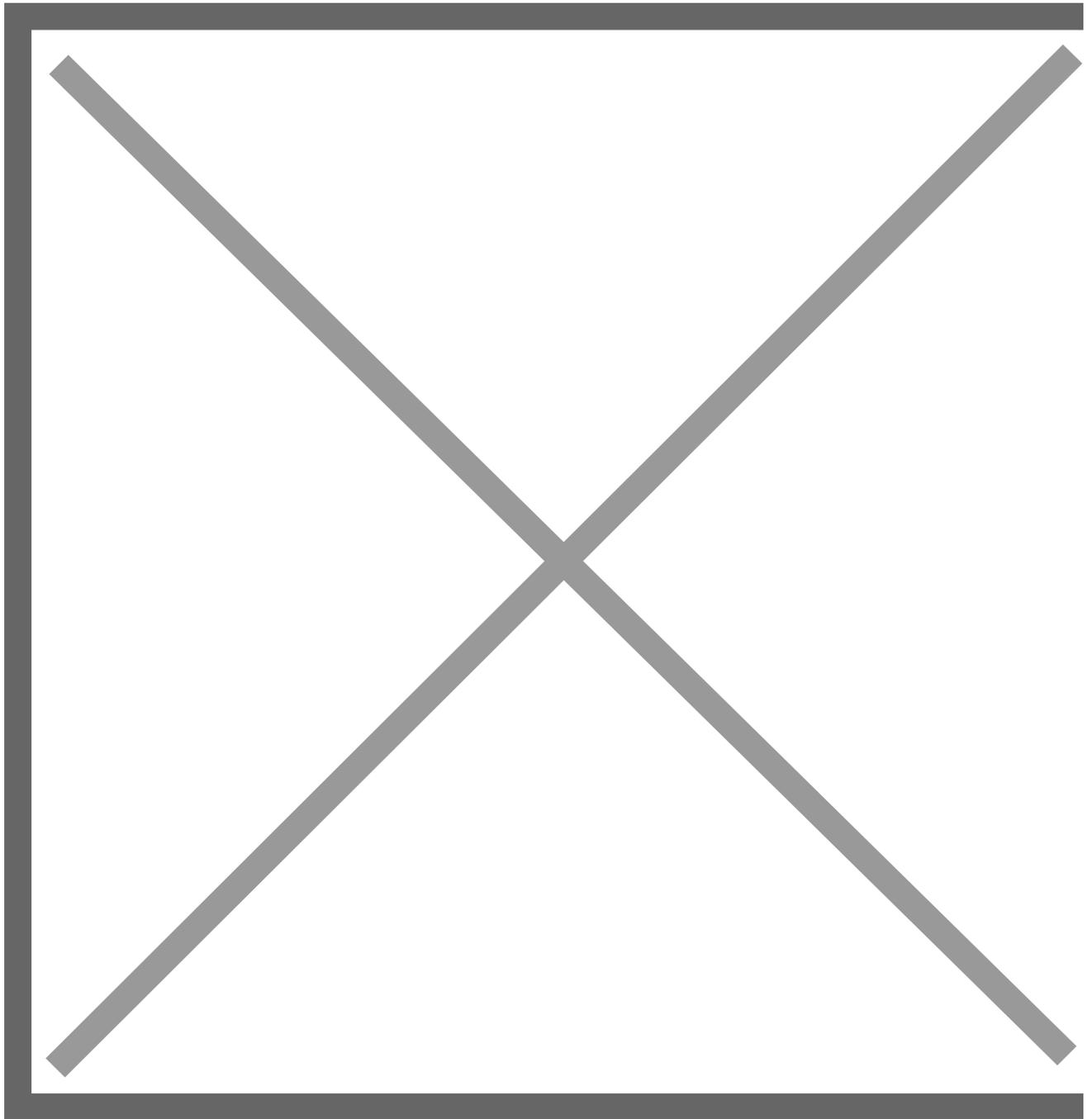
- **log**
- **db**
- **config**

Ils seront utilisés par l'application pour stocker des données (journaux, configuration, base de données).



Ensuite, ouvrez "**Container Manager**" puis cliquez sur "**Projet**" afin de créer un nouveau projet en cliquant sur le bouton nommé "**Créer**".

Vous devez donner un nom à ce projet, sélectionner le répertoire créé précédemment et indiquer le code de configuration Docker Compose permettant de déployer l'image Docker de l'application. Ce qui donne :



Voici le code Docker Compose (*aidez-vous des commentaires dans le fichier pour la compréhension de la config*) :

```
services:
  netaalertx:
    image: jokobsk/netaalertx:latest # Nom de l'image Docker à utiliser
    container_name: netaalertx      # Nom du conteneur Docker
    healthcheck:                    # Vérifier état de santé du conteneur (attention au port)
      test: curl -f http://127.0.0.1:15000/ || exit 1
    mem_limit: 1g                   # Limite RAM de 1 Go pour le conteneur
    security_opt:
```

```
- no-new-privileges:true      # Option de sécurité contre l'élévation de privilèges
volumes:
- /volume1/docker/netalertx/config:/app/config:rw      # Volume pour "config" (configuration)
- /volume1/docker/netalertx/db:/app/db:rw             # Volume pour "db" (base de données)
- /volume1/docker/netalertx/log:/app/front/log:rw     # Volume pour "log" (journaux)
environment:
TZ: Europe/Paris          # Fuseau horaire, ici Paris
PORT: 15000              # Port à utiliser sur la machine locale (adapter le healthcheck en fonction)
HOST_USER_ID: 1032       # UserID à utiliser (n'utilisez pas "root")
HOST_USER_GID: 100       # UserID à utiliser (n'utilisez pas "root")
# ALWAYS_FRESH_INSTALL: true # Réinitialise l'application (effacer toutes les données - désactivé ici)
network_mode: host       # Mode "host" obligatoire pour cette application
restart: on-failure:5    # Restart en cas de problème avec le conteneur ; 5 tentatives pour relancer
```

## Docker Compose NetAlertX pour Synology.jpg

Dans le cas présent, le port sur lequel sera joignable l'application, est le **port 15000**. Veillez à vérifier également les chemins vers les dossiers, sous l'instruction "**volumes**".

Le **conteneur sera exécuté à partir de l'utilisateur "docker"**, créé par mes soins sur le NAS. Il s'agit d'un utilisateur non privilégié qui ne dispose d'aucune permission spéciale sur le NAS, si ce n'est le droit de lecture et écriture dans le répertoire "**docker**".

Quand la configuration est prête, poursuivez jusqu'à la fin et lancez la création du projet... Patientez pendant le téléchargement de l'image Docker et la création du conteneur associé.

Dès à présent, vous pouvez accéder à l'application NetAlertX de cette façon :

- **http://<adresse IP de votre NAS>:15000**

**Vous avez désormais accès à l'application NetAlertX grâce à votre NAS !**

## NetAlertX sur NAS Synology avec Docker

Révision #1

Créé 12 janvier 2025 11:49:46 par Angelo

Mis à jour 1 avril 2025 18:42:33 par Angelo