

Serveur

Tout ce qui est du serveur, c'est par ici

- [Basic](#)
- [Cockpit](#)
- [Sauvegarde](#)
- [Docker](#)
 - [Installation de mon serveur](#)
 - [Portainer](#)
 - [Volumes](#)
 - [Réseau interne Docker](#)
 - [Commandes docker](#)
 - [Nginx Proxy Manager](#)
 - [Nextcloud](#)
 - [Bookstack](#)
 - [WordPress](#)
 - [Jellyfin](#)
 - [Collabora](#)
 - [Home Assistant](#)
 - [Mealie](#)
 - [Wireguard avec WebUI](#)
 - [Cloud-commander](#)
 - [Pi-Hole](#)
 - [Restreamer](#)
 - [Pyload](#)
 - [Heimdall](#)
 - [Adguard](#)
 - [OpenSpeedTest](#)
 - [Glances](#)
 - [Homarr](#)

- [AudioBookShelf](#)
- [FreshRSS](#)
- [MeTube](#)
- [Vaultwarden](#)
- [Yourls Raccourcisseur de lien](#)
- [Nginx-RTMP](#)
- [Srt-live-server](#)
- [Filegator](#)
- [Stirling-PDF](#)
- [Joal](#)

Basic

Gestion des utilisateurs

Crée un nouvel utilisateur

```
sudo adduser username
```

Changer le mot de passe d'un utilisateur

```
sudo passwd utilisateur
```

Ajouter un groupe

```
sudo addgroup nom_groupe
```

Ajouter un utilisateur dans un groupe

```
sudo adduser utilisateur groupe
```

Changer le répertoire personnel

```
sudo usermod -m -d /newhome/username username
```

Supprimer un mot de passe

Il suffit de démarrer en live et d'accéder au fichier `/etc/shadow`, ensuite repérer la ligne portant le nom de l'utilisateur puis supprimer les caractères entre les deux premières séries de « `:` », exemple pour Julien :

```
julien: $6$68oL9D6yVVCMTVswSb.51oFHZ/:17999:0:99999:7:::
```

Changer l'UID et le GID d'un utilisateur

Pour changer le GID (id du groupe) en **900** de l'utilisateur **toto** par exemple :

```
sudo groupmod -g 900 toto
```

Pour changer l'UID (**-u**) 800 et le GUI (**-g**) 900 de l'utilisateur **toto** :

```
sudo usermod -u 800 -g 900 toto
```

Transfère de fichiers

L'outil scp permet de transférer des fichiers entre 2 ordinateurs via SSH

```
scp -r -p source destination
```

Exemple d'un transfert de test.txt vers le répertoire Téléchargement d'un serveur :

```
scp -r -p /home/julien/Téléchargement/test.txt serveur@192.168.1.2/home/serveur/Téléchargement
```

Socks (équivalent vpn)

Cette astuce permet de passer par la connexion du serveur par lequel on se connecte via un tunnel SSH

En premier lieu, il faut créer une redirection du serveur sur un port local, pour se faire voici la commande à faire :

```
ssh user@serveur -p port -D 8080
```

Exemple :

```
ssh julien@158.136.345.678 -p 1234 -D 8080
```

Note : l'option **-p** sert juste à spécifier le port utilisé pour le port SSH s'il est différent du port par défaut (22).

Ensuite, il suffit de rediriger la connexion de l'ordinateur vers l'adresse locale avec le port 8080 avec comme protocole socks.

Sous Mate, il suffit d'aller dans serveur mandataire et de renseigner dans la ligne socks l'adresse

127.0.0.1 et le port **8080** Texte alternatif

Syntaxe pour les scripts

<https://bash.cyberciti.biz/guide/Shebang>

Explications sur les commandes à taper => <https://explainshell.com/#>

Hdparm

Ce paquet permet la gestion (entre autres) de l'énergie des disques durs.

`hdparm -S 1 /dev/sdX` permet de définir le temps avant mise en veille du disque dur lorsqu'il est inactif (1 après le -S signifie 5 secondes).

`hdparm -y /dev/sdX` permet de tester immédiatement la mise en veille du disque.

`hdparm -I /dev/sdX | grep level` permet de renseigner sur la possibilité d'effectuer la mise en veille (si le disque le supporte, vous obtiendrez un nombre (254 par défaut), et vous pouvez continuer.)

SSH

Pour lancer des programmes graphique sur le serveur distant en ssh, il suffit d'entrer cette commande une fois connecter `export DISPLAY=:0`

Problème d'empreinte suite à une réinstallation de l'ordinateur distant.

Voici la syntaxe pour supprimer l'empreinte de l'ordinateur précédent (remplacer IP par l'ip de la machine distante) :

```
ssh-keygen -f "/home/$USER/.ssh/known_hosts" -R "IP"
```

CRON

Cron permet d'exécuter des actions répétées sur le système, exemple redémarré l'ordinateur tous les jours à 2 heures du matin du lundi au samedi.

`crontab -e` permet l'édition

`crontab -l` permet de lister les actions.

Pour l'exemple voici la ligne que doit contenir crontab

```
0 2 * * 1-6 /sbin/reboot
```

Pour décortiquer : `0` (minutes) `2` (heures) `*` (jour du mois (`12`, `13`, `14`, etc.)) `*` (mois (1 pour janvier etc.)) `1-6` (jour de la semaine (lundi à samedi)) `/sbin/reboot` (la commande à exécuter).

Liens utiles :

<http://corntab.com/>

<https://crontab.guru/>

<http://www.cronmaker.com>

<https://crontab-generator.org>

INCRON

Incron permet d'exécuter des actions par rapport à un évènement sur le système, exemple déplacé un fichier lorsqu'un nouveau est créé.

Doc Ubuntu => <https://doc.ubuntu-fr.org/incron>

Nettoyage du système

APT

L'option **clean**, option radicale, supprime la totalité des paquets présents dans `/var/cache/apt/archives`. (Notez que cela n'a aucun impact sur les paquets installés.)

```
sudo apt-get clean
```

L'option **autoclean** permet de supprimer les paquets présents dans `/var/cache/apt/archives`, mais en conservant ceux qui ont un équivalent dans les dépôts ; par exemple, si `/var/cache/apt/archives` contient plusieurs versions du même logiciel, les plus anciennes, plus présentes dans les dépôts, seront supprimées, mais la version à jour (ayant un équivalent dans le dépôt) sera conservée. Ceci vous permet de récupérer beaucoup d'espace disque, mais moins toutefois qu'avec "**clean**".

```
sudo apt-get autoclean
```

DPKG

Voyons tous les paquets marqués comme **rc** par dpkg. Cet état signifie que les fichiers de configuration ne sont pas encore supprimés.

Vous pouvez voir la liste des paquets.

```
dpkg --get-configure | grep "^rc"
```

Extrayons les paquets marqués **rc**.

```
dpkg --list |grep "^rc" | cut -d " " -f 3
```

Maintenant, enlevons tous les paquets marqués comme **rc**.

```
dpkg --list |grep "^rc" | cut -d " " -f 3 | xargs sudo dpkg --purge
```

Variante :

```
apt purge $(dpkg --list |grep '^rc' |awk '{print $2}')
```

Bonus

Ajouter l'adresse IP en page d'accueil

Pour afficher l'adresse IP en TTY ou terminal lors de l'allumage du pc ou lors de la connexion, il y a deux solutions :

Avant la session

Pour ajouter l'adresse IP avant l'entrée dans la session, il suffit de créer un fichier avec l'extension `.issue` dans `/etc/issue.d/`.

L'option pour l'IP V4 est `\4` et `\6` pour la V6

Après la session

Pour ajouter l'adresse IP à l'entrée dans la session, il suffit de créer un fichier dans `/etc/motd.d/`.

L'option pour l'IP V4 est `\4` et `\6` pour la V6

Cockpit

Cockpit est un logiciel qui s'utilise via une interface web qui permet de gérer son serveur. On peut vérifier les mises à jours, la charge système, l'utilisation du stockage, arrêter ou redémarrer, il dispose également d'un terminal.

On peut s'y connecter en utilisant les identifiants des utilisateurs du système.

Installation

Il est présent dans les dépôts de la plupart des distributions, pour Debian:

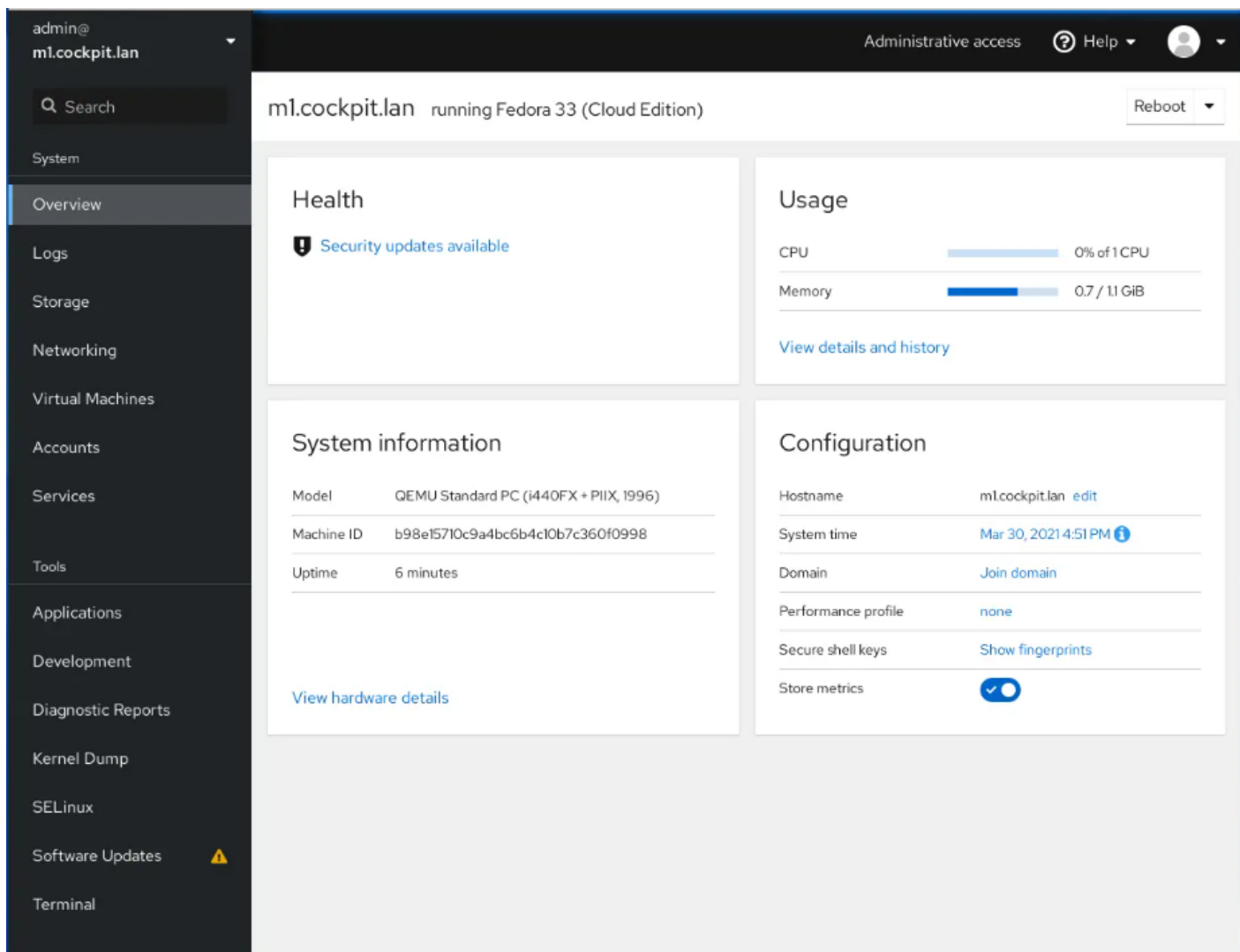
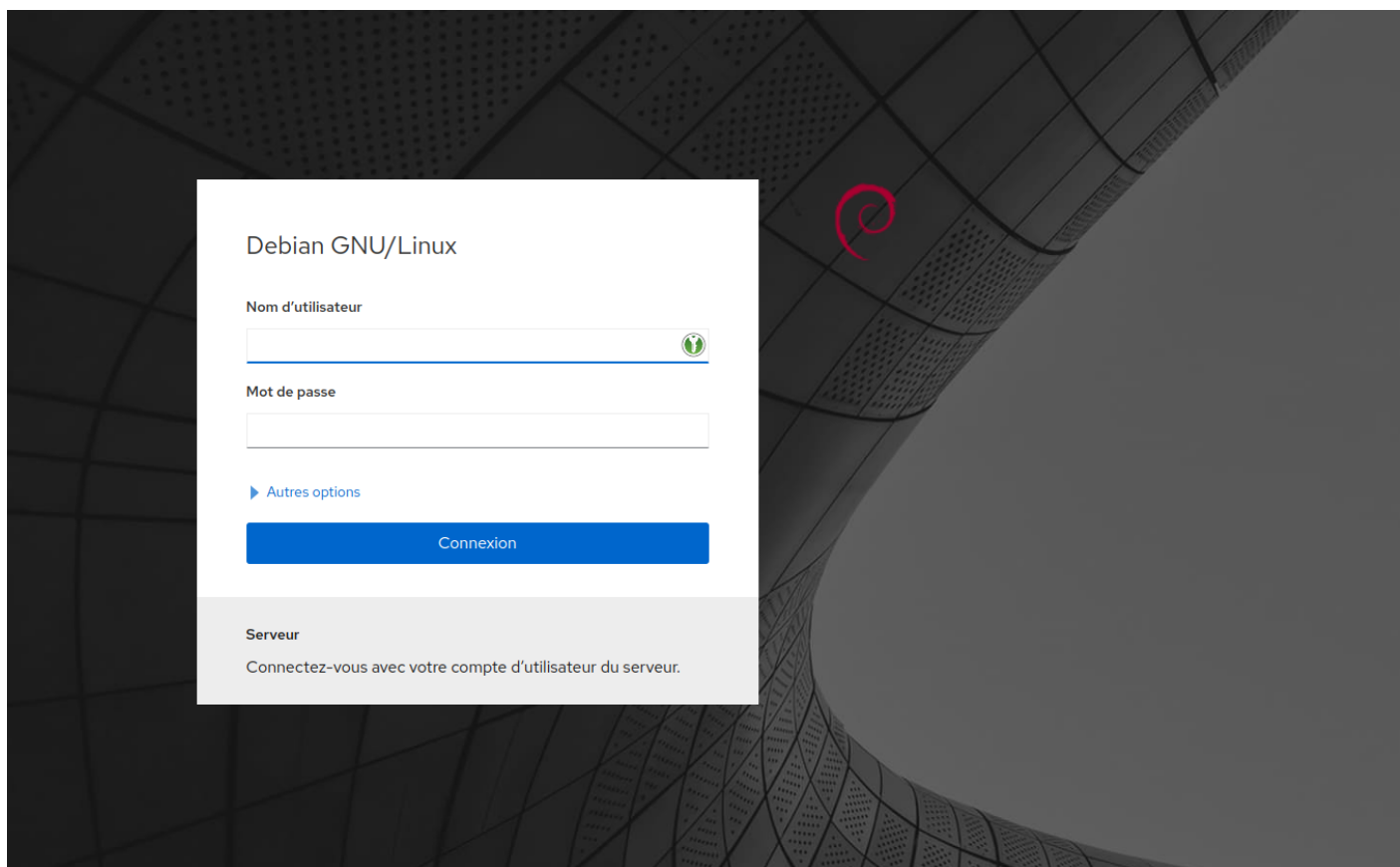
```
sudo apt-get install cockpit
```

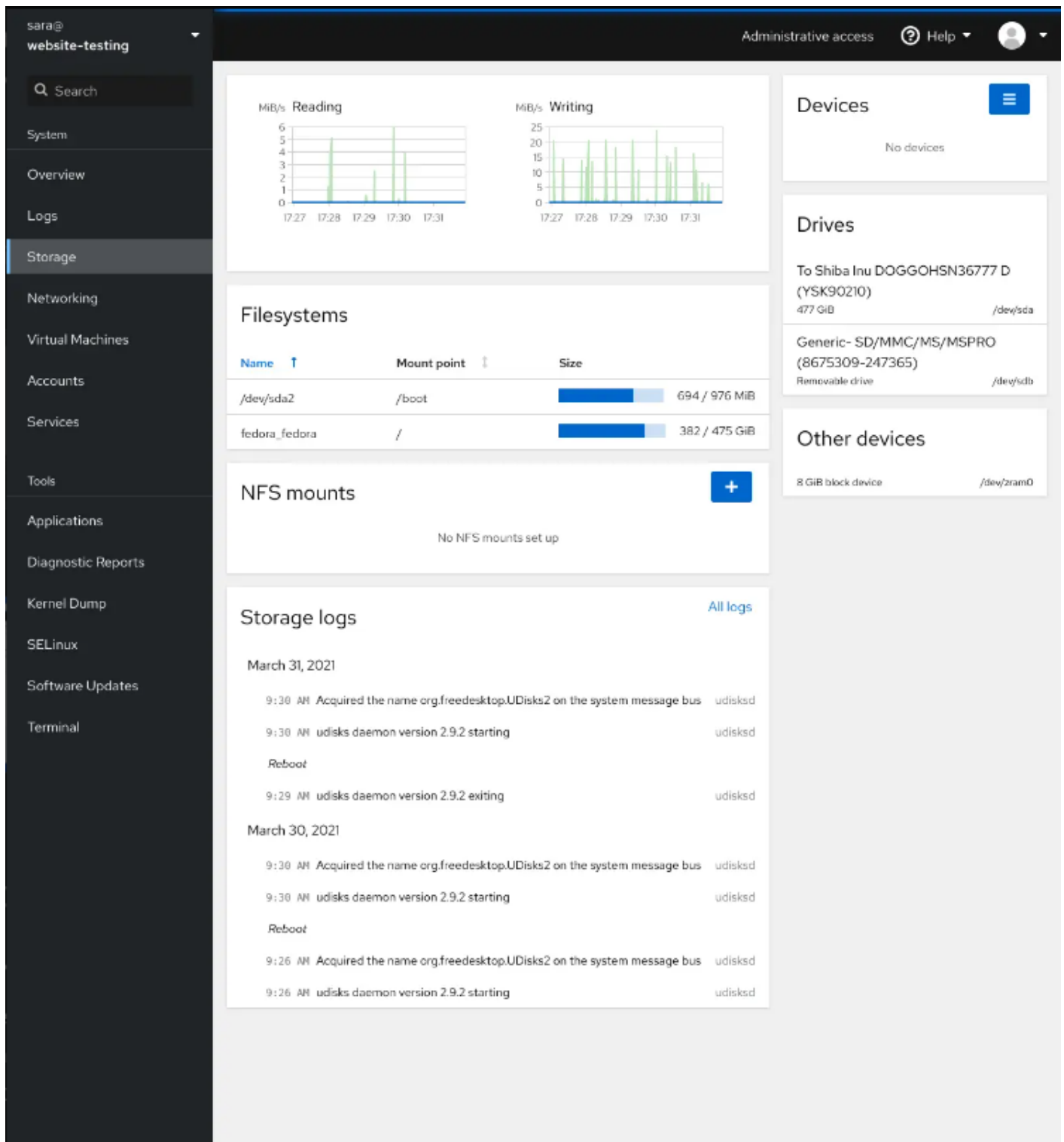
Puis on va le démarrer de suite

```
systemctl enable --now cockpit.socket
```

Utilisation

il suffit via un navigateur de s'y rendre avec son IP le port 9090, exemple <https://192.168.1.2:9090>





Sauvegarde

Borgbackup

Pour effectuer des sauvegardes incrémentielles, j'utilise BorgBackup qui a l'avantage de pouvoir garder seulement X sauvegardes, de restaurer uniquement un fichier ou dossier en particulier et de monter la sauvegarde dans le système de fichier.

En prérequis, il faut bien sûr avoir installé le paquet `borgbackup`.

Voici mon script :

```
#!/bin/bash

# Définir le répertoire de sauvegarde
backup_dir="/mnt/disque1/borgbackup"

# Définir le nom de l'archive
archive_name=borg_backup

# Définir le nombre de sauvegardes à conserver
num_backups_to_keep=5

# Obtenir la date et l'heure actuelles
current_date_time=$(date +%d-%m-%Y_%H-%M-%S)

# Supprimer les sauvegardes excédentaires
borg prune --keep-last $num_backups_to_keep "${backup_dir}"

# Créer la sauvegarde incrémentielle en excluant les dossiers indiqués
borg create -v --stats --compression none \
  --exclude /swapfile \
  --exclude '/dev/*' \
  --exclude '/proc/*' \
  --exclude '/sys/*' \
  --exclude '/tmp/*' \
```

```
--exclude '/run/*' \  
--exclude '/mnt/*' \  
--exclude '/media/*' \  
--exclude '/lost+found' \  
"${backup_dir}::${archive_name}_${current_date_time}" /  
  
# https://doc.ubuntu-fr.org/borgbackup  
# restauration  
# borg extract --progress /chemin/vers/repertoire/de/sauvegarde::borg_backup_01-01-2021_12-00-00  
  
# Montage de la sauvegarde  
# borg mount /backup/borg::20230119 /tmp/backup  
# Démontage  
# borg umount /tmp/backup
```

Ne pas oublier de personnaliser selon les besoins et la configuration du système de fichier.
Notez qu'à la fin du script, je me suis laissé des notes.
Ne pas oublier d'initialiser le dossier de sauvegarde avec la commande suivante :

```
borg init -e none /chemin/vers/dossier
```

DD et Dump

`dd` va me permettre de faire une image de la partition EFI et `dump` pour faire une sauvegarde de la partition EXT4.

L'avantage d'utiliser `dump` est qu'il fait partie des outils pour l'EXT1-4 pour faire des sauvegardes et ne sauvegarde pas l'espace vide.

Le script comporte une copie sur un disque local supplémentaire, puis l'envoi sur un serveur FTP.

Voici le script simplifié :

```
#!/bin/bash  
  
# serveur ftp  
FTP_USER='user'  
FTP_PASSWORD='password'  
FTP_SERVER='serveur'  
FTP_MOUNTPOINT='/point/de/montage/ftp'  
FTP_BACKUP='/dossier/seur/serveur/ftp'
```

```
# Définir la source EFI
src_efi=/dev/sda1
dst_dir=${FTP_MOUNTPOINT}${FTP_BACKUP}
dst_image=$TRANSIT_D/efi_image_$(date +%d-%m-%Y).img
error_log=$TRANSIT_D/ERROR_image_$(date +%d-%m-%Y)

# Zone Transit avant l'envoi en ftp
TRANSIT_D=/mnt/usb1/backup

# création du dossier de montage

fusermount -u $FTP_MOUNTPOINT ;
rm -r $FTP_MOUNTPOINT ;
mkdir $FTP_MOUNTPOINT

# Montage FTP

curlftpfs $FTP_SERVER $FTP_MOUNTPOINT -o user=$FTP_USER:$FTP_PASSWORD

# Copie avec dd pour l'EFI et dump pour l'ext4
echo "Démarrage de la création de l'image disque..."
dd if=$src_efi of=$dst_image bs=4096 conv=noerror,sync
dump -0uf $TRANSIT_D/dump-$(date +%d-%m-%Y) /

# Vérifier si la copie s'est terminée avec succès
if [ $? -eq 0 ]; then
    echo "Image disque créée avec succès dans $dst_image."
else
    echo "Erreur lors de la création de l'image disque."
    touch $error_log
fi

# Specifique Freebox

chmod 600 $TRANSIT_D/*

# transfère en FTP

sync
```

```
sleep 2
echo "Transfert FTP de efi_image_$(date +%d-%m-%Y).img"
mv $dst_image $dst_dir
echo "Transfert FTP de dump-$(date +%d-%m-%Y)"
mv $TRANSIT_D/dump-$(date +%d-%m-%Y) $dst_dir
mv $error_log $dst_dir ;

# suppression des fichier de plus de 10 jours

echo "Suppression des ancien fichiers"
find $dst_dir -type f -mtime +10 -delete

# Démontage du dossier ftp

echo "Démontage du dossier ftp"
sync
fusermount -u $FTP_MOUNTPOINT

# Suppression du point de montage

sleep 2
rm -r $FTP_MOUNTPOINT
```

Bien sûr, il ne faut pas oublier de l'adapter et de créer une tâche CRON.

Docker

Tout ce qui concerne docker ainsi que les paramétrages d'application, c'est par ici.

Installation de mon serveur

Étape d'installation de mon serveur en docker

Installation de Debian en NetInstall.

La première étape est d'installer une distribution sans interface graphique pour notre serveur, mon choix c'est orienté sur Debian qui à l'avantage pour moi d'utiliser **APT** pour la gestion des paquet et est très robuste.

Il est téléchargeable [ici](#).

Construction d'un RAID5.

Tuto bien expliqué avec [mdadm](#) justegeek.fr.

Docker

Portainer

Installation

Prérequis :

Il faut que ces 2 paquets soit installé sur le serveur, Debian pour ma part :

```
sudo apt-get install docker.io docker-compose
```

Docker compose

Il suffit de créer le fichier suivant se nommant `docker-compose.yml`

```
version: "2"
services:
  portainer:
    image: portainer/portainer-ce:latest
    container_name: portainer
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock
      - portainer_data:/data
    ports:
      - 8000:8000
      - 9000:9000
      - 9443:9443
    restart: always
volumes:
  portainer_data:
```

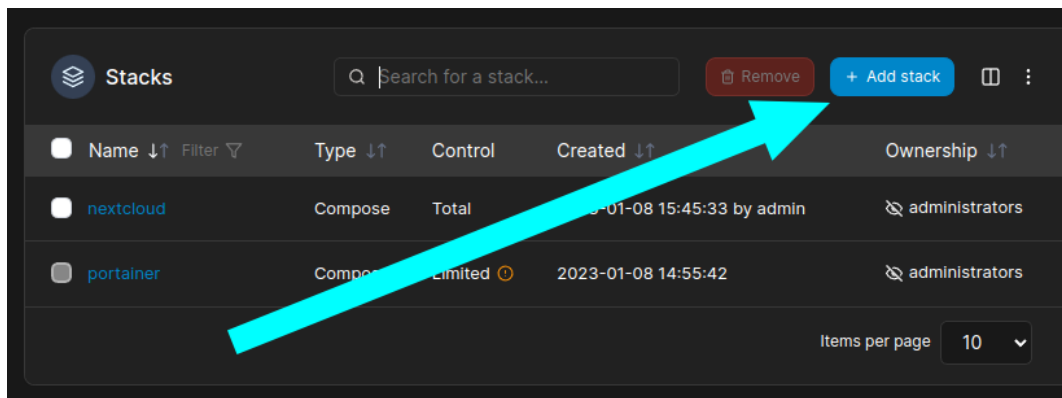
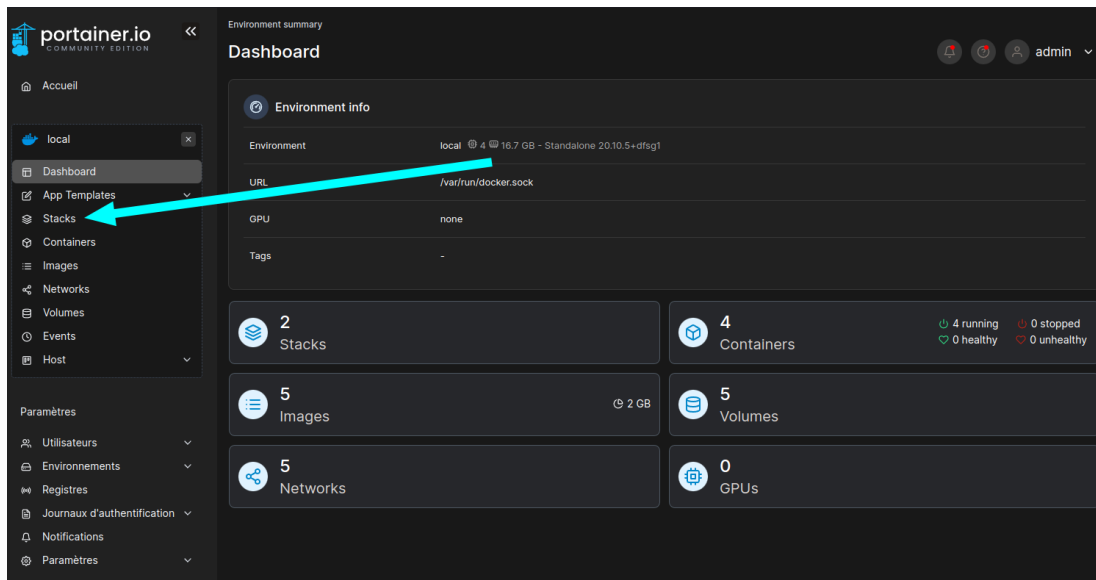
Une fois fait, il suffit de lancer la commande dans le dossier comportant le fichier précédent

```
sudo docker-compose up -d
```

Utilisation

Il suffit de se connecter sur l'interface web via <https://ip:9443>

Ajouter une stack



Nom

Cette pile sera déployée à l'aide: `docker compose`.

Méthode de construction

Web editor

Utilisez notre éditeur Web

Upload

Téléchargez depuis votre ordinateur


Repository

Utilisez un référentiel git

Custom template

Utilisez un modèle personnalisé

Éditeur Web

Ctrl + F pour la recherche 

Vous pouvez obtenir plus d'informations sur le format de fichier de composition dans la [documentation officielle](#).

Ce nom de conteneur est déjà utilisé par un autre conteneur fonctionnant dans cet environnement: `nextcloud`.

```
8 - PGID = 1000
9 - TZ = Europe / Paris
dix volumes:
11 - appdata: / config
12 - données: / données
13 ports:
14 - 28443: 443
15 redémarrer: à moins d'être arrêté
16 db:
17 image: mariadb:dernier
18 volumes:
19 - db_data: / var / lib / mysql
20 environnement:
21 - MYSQL_ROOT_PASSWORD = mot-de-passe
22 - MYSQL_DATABASE = nextcloud
23 - MYSQL_USER = nextcloud-admin
24 - MYSQL_PASSWORD = mot-de-passe-admin
25 ports:
26 - 23306: 3306
27 redémarrer: à moins d'être arrêté
28
29 volumes:
30 appdata:
31 données:
32 db_data:
```


Webhooks

Créez un

webhook


Stack




 Fonction Business Edition


Variables d'environnement

Ces valeurs seront utilisées comme substitutions dans le fichier de pile

 Advanced mode

 Switch to advanced mode to copy & paste multiple variables

+ Add an environment variable

 Load variables from .env file

Contrôle d'accès

Activez le

contrôle d'accès



Administrators

Je veux limiter la gestion de cette ressource aux administrateurs uniquement

Restricted

Je souhaite restreindre la gestion de cette ressource à un ensemble d'utilisateurs et / ou d'équipes

Actions

Déployez la pile

Astuces

Mode sombre

Il suffit d'aller sur le nom du compte en haut à droite => **My account**, dans la section **User theme**, on peut choisir **Dark Theme**.

Published Ports

Pour avoir l'accès direct quand on clique sur le port sans devoir effacer 0.0.0.0 pour remplacer par l'ip du serveur, il suffit d'aller sous **Settings** de gauche => **Environments** => **local** et renseigner l'**ip publique**.

Volumes

Pour personnaliser le nom des volumes en créant une stack il faut lui indiquer son nom avant les deux points:

```
version: "2"
services:
  exemple:
    image:
    container_name:
    environment:
      - PUID=1000
      - PGID=1000
    volumes:
      - le-nom-personnaliser:/config
    ports:
      - 80:80
```

Puis à la fin il faut rajouter à la racine le nom de ce volume décalé de 2 espaces et terminer par deux points attacher :

```
version: "2"
services:
  exemple:
    image:
    container_name:
    environment:
      - PUID=1000
      - PGID=1000
    volumes:
      - le-nom-personnaliser:/config
    ports:
      - 80:80
  volumes:
    le-nom-personnaliser:
```

Pour une lecture seul ont peut rajouter `:ro` derrière le chemin du conteneur.

volumes:

- le-nom-personnaliser:/config:ro

Réseau interne Docker

Nom du réseau

Pour avoir un nom de réseau personnalisé qui peut servir à avoir deux conteneurs sur le même réseau.

Il faut rajouter ces lignes-là dans le conteneur :

```
networks:  
  - nom_perso
```

Et rajouter les lignes suivantes à la fin du compose :

```
networks:  
  nom_perso:  
    name: nom_perso  
    driver: bridge
```

<https://stackoverflow.com/questions/38088279/communication-between-multiple-docker-compose-projects>

IP fixe du conteneur

Pour mettre une IP fixe pour un conteneur, il suffit de rajouter les lignes comme dans l'exemple :

```
version: '3'  
services:  
  my_service:  
    image: my_image  
    networks:  
      nom_perso:  
        ipv4_address: 192.168.0.10
```

networks:

nom_perso:

ipam:

config:

- subnet: 192.168.0.0/24

Commandes docker

Ouvrir le shell d'un conteneur.

Une fois en ssh sur la machine de l'hôte, il suffit de lancer la commande suivante en root :

```
docker exec -it nom_du_conteneur /bin/bash
```

Lancer une commande dans un conteneur depuis l'hôte avec un utilisateur spécifique.

```
docker exec -it -u utilisateur nom_du_conteneur commande à lancer
```

Docker

Nginx Proxy Manager

J'utilise l'image de JC21

Voici le compose :

```
services:
  app:
    image: 'jc21/nginx-proxy-manager:latest'
    restart: unless-stopped
    ports:
      - '80:80'
      - '81:81'
      - '443:443'
    volumes:
      - data:/data
      - letsencrypt:/etc/letsencrypt
volumes:
  data:
  letsencrypt:
```

Pour accéder à Nginx proxy manager, il suffit d'aller à l'adresse http du serveur sur le port 81 (qui peut être personnalisé).

Ne pas oublier d'ouvrir les ports 80 et 443 du routeur.

Login par défaut

```
admin@example.com
changeme
```

[Source](#)

Ajouter des protections

Ajout de configuration avancé Nginx

Edit Proxy Host

⚡ Details

📁 Custom locations

🔒 SSL

⚙️ Advanced

Nginx variables available to you are:

- \$server # Host/IP
- \$port # Port Number
- \$forward_scheme # http or https

Custom Nginx Configuration

Cancel

Save

```
# Active la protection contre les attaques XSS (Cross-Site Scripting) dans le navigateur web, en bloquant les
scripts potentiellement malveillants.
more_set_headers "X-XSS-Protection: 1; mode=block";

# Empêche le navigateur d'effectuer une détection automatique du type de contenu, ce qui empêche certaines
attaques de type MIME Sniffing.
more_set_headers "X-Content-Type-Options: nosniff";

# Indique aux moteurs de recherche de ne pas indexer et suivre les liens de cette page.
# Mettre index et follow pour un site web devant être référencer.
more_set_headers "X-Robots-Tag: noindex, nofollow";

# Contrôle les informations de référent envoyées lors de la navigation entre sites web, évitant ainsi les fuites
d'informations sensibles.
more_set_headers "Referrer-Policy: no-referrer-when-downgrade";

# Force le navigateur à utiliser HTTPS pour les ressources chargées, au lieu d'HTTP, améliorant ainsi la sécurité.
more_set_headers "Content-Security-Policy: upgrade-insecure-requests";

# Désactive la fonctionnalité d'intérêt des cohorts, qui peut avoir des implications sur la confidentialité des
données.
more_set_headers "Permissions-Policy: interest-cohort=()";
```

Empêche le chargement de cette page dans un iframe, sauf si le site est de la même origine, évitant ainsi les attaques de type Clickjacking.

```
more_set_headers "X-Frame-Options: SAMEORIGIN";
```

Indique qu'aucune politique de domaine croisé n'est autorisée, ce qui empêche certaines attaques impliquant du contenu cross-domain.

```
more_set_headers "X-Permitted-Cross-Domain-Policies: none";
```

Le niveau de sécurité peut être vérifié sur ce site => <https://securityheaders.com/>

Nextcloud

Stack

Voici une stack pour Nextcloud avec ça base de donnée.

Ne pas oublier de personnaliser les ports et mot de passe souhaiter.

```
version: "2.1"

services:
  nextcloud:
    image: lscr.io/linuxserver/nextcloud:latest
    container_name: nextcloud
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Europe/Paris
    volumes:
      - appconfig:/config
      - data:/data
    ports:
      - 28443:443/tcp
    networks:
      nextcloud-net:
        ipv4_address: 172.140.6.4
    restart: unless-stopped
    depends_on:
      - db
      - valkey
  db:
    image: mariadb:latest
    volumes:
      - db_data:/var/lib/mysql
    environment:
```

- MYSQL_ROOT_PASSWORD=mot-de-passe
- MYSQL_DATABASE=nextcloud
- MYSQL_USER=nextcloud-admin
- MYSQL_PASSWORD=mot-de-passe-admin

ports:

- 23306:3306

networks:

nextcloud-net:

ipv4_address: 172.140.6.3

restart: unless-stopped

valkey:

image: valkey/valkey:latest

volumes:

- valkey_data:/data

ports:

- 6379:6379/tcp

networks:

nextcloud-net:

ipv4_address: 172.140.6.2

restart: unless-stopped

volumes:

appconfig:

data:

db_data:

redis_data:

networks:

nextcloud-net:

ipam:

config:

- subnet: 172.140.6.0/16

Configuration

Il faut se connecter à l'interface de nextcloud en tapant l'ip du serveur:port choisi précédemment.

Par rapport à l'exemple : <https://192.168.1.34:28443>

Une fois sur l'interface, créer le premier utilisateur et renseigner la base de donnée. Exemple par rapport à la stack précédente :



Créer un compte administrateur

Nom d'utilisateur

admin



Mot de passe

.....



Mot de passe très faible

Stockage & base de données

Répertoire des données

/data

Choisissez votre base de données

SQLite

MySQL/MariaDB

PostgreSQL

Utilisateur de la base de données

nextcloud-admin

Mot de passe de la base de données

.....



Nom de la base de données

nextcloud

Hôte de la base de données

192.168.1.34:23306

Veuillez spécifier le numéro du port avec le nom de l'hôte (ex: localhost:5432).

Installer

Besoin d'aide ? [Lire la documentation](#)

Avant toute manipulation, il est conseillé de faire une copie de sauvegarde au cas où.

/config/www/nextcloud/config/config.php

Nom de domaine

```
'trusted_domains' =>
array (
  0 => '192.168.1.34:28443',
  1 => 'sousdomaine.domaine.com',
  0 => '172.140.6.04:443',
  0 => '172.140.6.04:28443',
),
```

OnlyOffice

```
'onlyoffice' =>
array (
  'verify_peer_off' => true,
),
```

Valkey anciennement Redis

```
'redis' =>
array (
  'host' => '172.140.6.2',
  'port' => 6379,
),
```

Configuration supplémentaire pour le transfert le fichier Nextcloud.

Maintenant que tout est en place, il existe une limitation au niveau du transfert des données, on peut voir les valeurs dans l'interface du Nextcloud => Paramètres => Système => la partie **PHP**.

Pour afficher les valeurs, il suffit de se connecter à la console du serveur Nextcloud puis lire le fichier `php.ini` qui se trouve dans `/etc/php81/php.ini`.

Rechercher les valeurs `memory_limit=`, `post_max_size=` et `upload_max_filesize=`.

Pour modifier les valeurs, il faut rajouter les options avec leurs valeurs dans le fichier `php-local.ini` qui se trouve dans `/config/php/`. Les valeurs ici surpasseront ceux de `php.ini`.

Exemple :

```
; Edit this file to override php.ini directives and restart the container
```

```
memory_limit=4096M
upload_max_filesize=10240M
max_file_uploads = 20
post_max_size=10240M
```

Sous-dossiers/Fichiers manquants.

J'ai rajouté un stockage supplémentaire et le souci est que je ne voyais pas tout les sous-dossiers via l'interface web, la solution est d'adapter et lancer la commande suivante, directement sur la machine hôte (fonctionne uniquement pour Nextcloud de Linuxserve).

```
docker exec container_name sudo -u abc php /config/www/nextcloud/occ files:scan --all
```

Mise à jour Nextcloud 27

```
docker exec container_name sudo -u abc /app/www/public/occ files:scan --all
```

Mise à jour Nextcloud 28, plus besoin.

Fichier/Dossier toujours pas visible.

Suite à la commande suivante :

```
docker exec container_name sudo -u abc /app/www/public/occ files:scan --all
```

Il y a l'erreur suivante :

```
Entry "chemin/vers/le/fichier/dossier" will not be accessible due to incompatible encoding
```

La solution est d'utiliser le programme `convmv` pour conversion en utf-8

```
convmv -f utf-8 -t utf-8 -r --notest --nfc chemin/vers/le/fichier/dossier
```

Erreurs :

X-Robots-Tag

L'erreur "*L'en-tête HTTP "X-Robots-Tag" n'est pas configurée pour être égale à "noindex, nofollow"*" se corrige en éditant le fichier `default.conf` qui se trouve dans `/config/nginx/site-confs/`, il suffit de trouver la ligne `add_header X-Robots-Tag` et remplacer `none` par `noindex, nofollow`.

FPM initialization failed

De temps en temps, je reçois l'erreur suivante :

```
[12-Apr-2023 09:15:12] ERROR: unable to bind listening socket for address '127.0.0.1:9000': Address in use (98)
[12-Apr-2023 09:15:12] ERROR: FPM initialization failed
```

Ce qui rend inaccessible Nextcloud, pour résoudre le problème, il faut libérer le port 9000, hors, il est utilisé par défaut pour portainer en http, du coup, j'ai changé le port externe de portainer.

Bookstack

Installation

Voici la stack que j'ai utilisé pour installer Bookstack en docker.

Il faut bien-sûr remplacer les mots de passe, le port ainsi que le lien choisi pour y accéder.

```
---
version: "2"
services:
  bookstack:
    image: lscr.io/linuxserver/bookstack
    container_name: bookstack
    environment:
      - PUID=1000
      - PGID=1000
      - APP_URL=https://wiki.domaine.fr
      #- ALLOWED_IFRAME_HOSTS=https://wiki.domaine.fr
      - DB_HOST=bookstack_db
      - DB_PORT=3306
      - DB_USER=bookstack
      - DB_PASS=<mot de passe DB>
      - DB_DATABASE=bookstackapp
    volumes:
      - config:/config
    ports:
      - 64080:80
    restart: unless-stopped
    depends_on:
      - bookstack_db
  bookstack_db:
    image: lscr.io/linuxserver/mariadb
    container_name: bookstack_db
    environment:
```

```
- PUID=1000
- PGID=1000
- MYSQL_ROOT_PASSWORD=<mot de passe root DB>
- TZ=Europe/Paris
- MYSQL_DATABASE=bookstackapp
- MYSQL_USER=bookstack
- MYSQL_PASSWORD=<mot de passe DB>
```

volumes:

```
- config-db:/config
```

restart: unless-stopped

volumes:

config:

config-db:

Une fois le déploiement fait, il faut créer les redirections DNS et générer un certificat.

Une fois les opérations faite, il suffit d'accéder au sous-domaine et se connecter.

Utilisateur par défaut est `admin@admin.com` et le mot de passe `password` a changé tout de suite `[[`.

Avancer

Pour les paramètres avancés, il faut aller dans `/config/www/.env`.

Un paramètre intéressant est le dark-mode, on peut également configurer le serveur SMTP pour l'envoi de courriel.

Voici la liste complète des paramètres que l'on peut modifier ou ajouter :

```
# Full list of environment variables that can be used with BookStack.
# Selectively copy these to your '.env' file as required.
# Each option is shown with it's default value.
# Do not copy this whole file to use as your '.env' file.

# Application environment
# Can be 'production', 'development', 'testing' or 'demo'
APP_ENV=production

# Enable debug mode
# Shows advanced debug information and errors.
# CAN EXPOSE OTHER VARIABLES, LEAVE DISABLED
```

APP_DEBUG=false

Application key

Used for encryption where needed.

Run `php artisan key:generate` to generate a valid key.

APP_KEY=SomeRandomString

Application URL

This must be the root URL that you want to host BookStack on.

All URL's in BookStack will be generated using this value.

APP_URL=https://example.com

Application default language

The default language choice to show.

May be overridden by user-preference or visitor browser settings.

APP_LANG=en

Auto-detect language for public visitors.

Uses browser-sent headers to infer a language.

APP_LANG will be used if such a header is not provided.

APP_AUTO_LANG_PUBLIC=true

Application timezone

Used where dates are displayed such as on exported content.

Valid timezone values can be found here: <https://www.php.net/manual/en/timezones.php>

APP_TIMEZONE=UTC

Application theme

Used to specific a themes/<APP_THEME> folder where BookStack UI

overrides can be made. Defaults to disabled.

APP_THEME=false

Trusted proxies

Used to indicate trust of systems that proxy to the application so

certain header values (Such as "X-Forwarded-For") can be used from the

incoming proxy request to provide origin detail.

Set to an IP address, or multiple comma seperated IP addresses.

Can alternatively be set to "*" to trust all proxy addresses.

APP_PROXIES=null

Database details

Host can contain a port (localhost:3306) or a separate DB_PORT option can be used.

DB_HOST=localhost

DB_PORT=3306

DB_DATABASE=database_database

```
DB_USERNAME=database_username
DB_PASSWORD=database_user_password
```

```
# MySQL specific connection options
# Path to Certificate Authority (CA) certificate file for your MySQL instance.
# When this option is used host name identity verification will be performed
# which checks the hostname, used by the client, against names within the
# certificate itself (Common Name or Subject Alternative Name).
MYSQL_ATTR_SSL_CA="/path/to/ca.pem"
```

```
# Mail system to use
# Can be 'smtp' or 'sendmail'
MAIL_DRIVER=smtp
```

```
# Mail sending options
MAIL_FROM=mail@bookstackapp.com
MAIL_FROM_NAME=BookStack
```

```
# SMTP mail options
MAIL_HOST=localhost
MAIL_PORT=1025
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null
```

```
# Cache & Session driver to use
# Can be 'file', 'database', 'memcached' or 'redis'
CACHE_DRIVER=file
SESSION_DRIVER=file
```

```
# Session configuration
SESSION_LIFETIME=120
SESSION_COOKIE_NAME=bookstack_session
SESSION_SECURE_COOKIE=false
```

```
# Cache key prefix
# Can be used to prevent conflicts multiple BookStack instances use the same store.
CACHE_PREFIX=bookstack
```

```
# Memcached server configuration
# If using a UNIX socket path for the host, set the port to 0
# This follows the following format: HOST:PORT:WEIGHT
# For multiple servers separate with a comma
MEMCACHED_SERVERS=127.0.0.1:11211:100
```

```
# Redis server configuration
# This follows the following format: HOST:PORT:DATABASE
# or, if using a password: HOST:PORT:DATABASE:PASSWORD
# For multiple servers separate with a comma. These will be clustered.
REDIS_SERVERS=127.0.0.1:6379:0

# Queue driver to use
# Can be 'sync', 'database' or 'redis'
QUEUE_CONNECTION=sync

# Storage system to use
# Can be 'local', 'local_secure' or 's3'
STORAGE_TYPE=local

# Image storage system to use
# Defaults to the value of STORAGE_TYPE if unset.
# Accepts the same values as STORAGE_TYPE.
STORAGE_IMAGE_TYPE=local

# Attachment storage system to use
# Defaults to the value of STORAGE_TYPE if unset.
# Accepts the same values as STORAGE_TYPE although 'local' will be forced to 'local_secure'.
STORAGE_ATTACHMENT_TYPE=local_secure

# Amazon S3 storage configuration
STORAGE_S3_KEY=your-s3-key
STORAGE_S3_SECRET=your-s3-secret
STORAGE_S3_BUCKET=s3-bucket-name
STORAGE_S3_REGION=s3-bucket-region

# S3 endpoint to use for storage calls
# Only set this if using a non-Amazon s3-compatible service such as Minio
STORAGE_S3_ENDPOINT=https://my-custom-s3-compatible.service.com:8001

# Storage URL prefix
# Used as a base for any generated image urls.
# An s3-format URL will be generated if not set.
STORAGE_URL=false

# Authentication method to use
# Can be 'standard', 'ldap', 'saml2' or 'oidc'
AUTH_METHOD=standard

# Automatically initiate login via external auth system if it's the only auth method.
# Works with saml2 or oidc auth methods.
```


AUTH_AUTO_INITIATE=false

Social authentication configuration

All disabled by default.

Refer to <https://www.bookstackapp.com/docs/admin/third-party-auth/>

AZURE_APP_ID=false

AZURE_APP_SECRET=false

AZURE_TENANT=false

AZURE_AUTO_REGISTER=false

AZURE_AUTO_CONFIRM_EMAIL=false

DISCORD_APP_ID=false

DISCORD_APP_SECRET=false

DISCORD_AUTO_REGISTER=false

DISCORD_AUTO_CONFIRM_EMAIL=false

FACEBOOK_APP_ID=false

FACEBOOK_APP_SECRET=false

FACEBOOK_AUTO_REGISTER=false

FACEBOOK_AUTO_CONFIRM_EMAIL=false

GITHUB_APP_ID=false

GITHUB_APP_SECRET=false

GITHUB_AUTO_REGISTER=false

GITHUB_AUTO_CONFIRM_EMAIL=false

GITLAB_APP_ID=false

GITLAB_APP_SECRET=false

GITLAB_BASE_URI=false

GITLAB_AUTO_REGISTER=false

GITLAB_AUTO_CONFIRM_EMAIL=false

GOOGLE_APP_ID=false

GOOGLE_APP_SECRET=false

GOOGLE_SELECT_ACCOUNT=false

GOOGLE_AUTO_REGISTER=false

GOOGLE_AUTO_CONFIRM_EMAIL=false

OKTA_BASE_URL=false

OKTA_APP_ID=false

OKTA_APP_SECRET=false

OKTA_AUTO_REGISTER=false

OKTA_AUTO_CONFIRM_EMAIL=false

```
SLACK_APP_ID=false
SLACK_APP_SECRET=false
SLACK_AUTO_REGISTER=false
SLACK_AUTO_CONFIRM_EMAIL=false
```

```
TWITCH_APP_ID=false
TWITCH_APP_SECRET=false
TWITCH_AUTO_REGISTER=false
TWITCH_AUTO_CONFIRM_EMAIL=false
```

```
TWITTER_APP_ID=false
TWITTER_APP_SECRET=false
TWITTER_AUTO_REGISTER=false
TWITTER_AUTO_CONFIRM_EMAIL=false
```

```
# LDAP authentication configuration
# Refer to https://www.bookstackapp.com/docs/admin/ldap-auth/
LDAP_SERVER=false
LDAP_BASE_DN=false
LDAP_DN=false
LDAP_PASS=false
LDAP_USER_FILTER=false
LDAP_VERSION=false
LDAP_START_TLS=false
LDAP_TLS_INSECURE=false
LDAP_ID_ATTRIBUTE=uid
LDAP_EMAIL_ATTRIBUTE=mail
LDAP_DISPLAY_NAME_ATTRIBUTE=cn
LDAP_THUMBNAIL_ATTRIBUTE=null
LDAP_FOLLOW_REFERRALS=true
LDAP_DUMP_USER_DETAILS=false
```

```
# LDAP group sync configuration
# Refer to https://www.bookstackapp.com/docs/admin/ldap-auth/
LDAP_USER_TO_GROUPS=false
LDAP_GROUP_ATTRIBUTE="memberOf"
LDAP_REMOVE_FROM_GROUPS=false
LDAP_DUMP_USER_GROUPS=false
```

```
# SAML authentication configuration
# Refer to https://www.bookstackapp.com/docs/admin/saml2-auth/
SAML2_NAME=SSO
SAML2_EMAIL_ATTRIBUTE=email
SAML2_DISPLAY_NAME_ATTRIBUTES=username
SAML2_EXTERNAL_ID_ATTRIBUTE=null
```

```
SAML2_IDP_ENTITYID=null
SAML2_IDP_SSO=null
SAML2_IDP_SLO=null
SAML2_IDP_x509=null
SAML2_ONELOGIN_OVERRIDES=null
SAML2_DUMP_USER_DETAILS=false
SAML2_AUTOLOAD_METADATA=false
SAML2_IDP_AUTHNCONTEXT=true
SAML2_SP_x509=null
SAML2_SP_x509_KEY=null

# SAML group sync configuration
# Refer to https://www.bookstackapp.com/docs/admin/saml2-auth/
SAML2_USER_TO_GROUPS=false
SAML2_GROUP_ATTRIBUTE=group
SAML2_REMOVE_FROM_GROUPS=false

# OpenID Connect authentication configuration
# Refer to https://www.bookstackapp.com/docs/admin/oidc-auth/
OIDC_NAME=SSO
OIDC_DISPLAY_NAME_CLAIMS=name
OIDC_CLIENT_ID=null
OIDC_CLIENT_SECRET=null
OIDC_ISSUER=null
OIDC_ISSUER_DISCOVER=false
OIDC_PUBLIC_KEY=null
OIDC_AUTH_ENDPOINT=null
OIDC_TOKEN_ENDPOINT=null
OIDC_ADDITIONAL_SCOPES=null
OIDC_DUMP_USER_DETAILS=false
OIDC_USER_TO_GROUPS=false
OIDC_GROUPS_CLAIM=groups
OIDC_REMOVE_FROM_GROUPS=false

# Disable default third-party services such as Gravatar and Draw.IO
# Service-specific options will override this option
DISABLE_EXTERNAL_SERVICES=false

# Use custom avatar service, Sets fetch URL
# Possible placeholders: ${hash} ${size} ${email}
# If set, Avatars will be fetched regardless of DISABLE_EXTERNAL_SERVICES option.
# Example:
AVATAR_URL=https://seccdn.libravatar.org/avatar/${hash}?s=${size}&d=identicon
AVATAR_URL=
```

```
# Enable diagrams.net integration
# Can simply be true/false to enable/disable the integration.
# Alternatively, It can be URL to the diagrams.net instance you want to use.
# For URLs, The following URL parameters should be included:
embed=1&proto=json&spin=1&configure=1
DRAWIO=true

# Default item listing view
# Used for public visitors and user's without a preference.
# Can be 'list' or 'grid'.
APP_VIEWS_BOOKS=list
APP_VIEWS_BOOKSHELVES=grid
APP_VIEWS_BOOKSHELF=grid

# Use dark mode by default
# Will be overridden by any user/session preference.
APP_DEFAULT_DARK_MODE=false

# Page revision limit
# Number of page revisions to keep in the system before deleting old revisions.
# If set to 'false' a limit will not be enforced.
REVISION_LIMIT=100

# Recycle Bin Lifetime
# The number of days that content will remain in the recycle bin before
# being considered for auto-removal. It is not a guarantee that content will
# be removed after this time.
# Set to 0 for no recycle bin functionality.
# Set to -1 for unlimited recycle bin lifetime.
RECYCLE_BIN_LIFETIME=30

# File Upload Limit
# Maximum file size, in megabytes, that can be uploaded to the system.
FILE_UPLOAD_SIZE_LIMIT=50

# Export Page Size
# Primarily used to determine page size of PDF exports.
# Can be 'a4' or 'letter'.
EXPORT_PAGE_SIZE=a4

# Allow <script> tags in page content
# Note, if set to 'true' the page editor may still escape scripts.
ALLOW_CONTENT_SCRIPTS=false
```

Indicate if robots/crawlers should crawl your instance.
Can be 'true', 'false' or 'null'.
The behaviour of the default 'null' option will depend on the 'app-public' admin setting.
Contents of the robots.txt file can be overridden, making this option obsolete.
ALLOW_ROBOTS=null

Allow server-side fetches to be performed to potentially unknown
and user-provided locations. Primarily used in exports when loading
in externally referenced assets.
Can be 'true' or 'false'.
ALLOW_UNTRUSTED_SERVER_FETCHING=false

A list of hosts that BookStack can be iframed within.
Space separated if multiple. BookStack host domain is auto-inferred.
For Example: ALLOWED_IFRAME_HOSTS="https://example.com https://a.example.com"
Setting this option will also auto-adjust cookies to be SameSite=None.
ALLOWED_IFRAME_HOSTS=null

A list of sources/hostnames that can be loaded within iframes within BookStack.
Space separated if multiple. BookStack host domain is auto-inferred.
Can be set to a lone "*" to allow all sources for iframe content (Not advised).
Defaults to a set of common services.
Current host and source for the "DRAWIO" setting will be auto-appended to the sources
configured.
ALLOWED_IFRAME_SOURCES="https://*.draw.io https://*.youtube.com https://*.youtube-
nocookie.com https://*.vimeo.com"

The default and maximum item-counts for listing API requests.
API_DEFAULT_ITEM_COUNT=100
API_MAX_ITEM_COUNT=500

The number of API requests that can be made per minute by a single user.
API_REQUESTS_PER_MIN=180

Enable the logging of failed email+password logins with the given message.
The default log channel below uses the php 'error_log' function which commonly
results in messages being output to the webserver error logs.
The message can contain a %u parameter which will be replaced with the login
user identifier (Username or email).
LOG_FAILED_LOGIN_MESSAGE=false
LOG_FAILED_LOGIN_CHANNEL=errorlog_plain_webserver

Alter the precision of IP addresses stored by BookStack.
Should be a number between 0 and 4, where 4 retains the full IP address
and 0 completely hides the IP address. As an example, a value of 2 for the

```
# IP address '146.191.42.4' would result in '146.191.x.x' being logged.  
# For the IPv6 address '2001:db8:85a3:8d3:1319:8a2e:370:7348' this would result as:  
# '2001:db8:85a3:8d3:x:x:x:x'  
IP_ADDRESS_PRECISION=4
```

Docker

WordPress

Voici un compose pour installer WordPress.

Ne pas oublier de personnaliser les noms, mot de passe ainsi que le port souhaiter.

```
version: '2'

services:
  db:
    image: mysql:latest
    volumes:
      - db_data:/var/lib/mysql
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: ${MYSQL_DATABASE_PASSWORD}
      MYSQL_DATABASE_PASSWORD: mot-de-passe
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: wordpress

  wordpress:
    image: wordpress:latest
    ports:
      - 22650:80
    restart: always
    environment:
      WORDPRESS_DB_HOST: db:3306
      WORDPRESS_DB_USER: wordpress
      WORDPRESS_DB_PASSWORD: wordpress
    volumes:
      - html:/var/www/html

volumes:
  db_data:
  html:
```

Pour augmenter la taille de téléversement, il suffit de rajouter ces 2 lignes dans le fichier `.htaccess` :

```
php_value post_max_size 128M
php_value upload_max_filesize 128M
```

Migrer le nom de domaine :

Une fois le changement fait, il suffit de lancer cette requête SQL en adaptant les bonnes valeurs :

```
# Changer l'URL du site
UPDATE wp_options
SET option_value = replace(option_value, 'http://www.ancien-site.fr', 'http://www.nouveau-site.fr')
WHERE option_name = 'home'
OR option_name = 'siteurl';

# Changer l'URL des GUID
UPDATE wp_posts
SET guid = REPLACE (guid, 'http://www.ancien-site.fr', 'http://www.nouveau-site.fr');

# Changer l'URL des médias dans les articles et pages
UPDATE wp_posts
SET post_content = REPLACE (post_content, 'http://www.ancien-site.fr', 'http://www.nouveau-site.fr');

# Changer l'URL des données meta
UPDATE wp_postmeta
SET meta_value = REPLACE (meta_value, 'http://www.ancien-site.fr', 'http://www.nouveau-site.fr');
```

Ensuite, se rendre à la fin du fichier `wp-config.php` et rajouter les 2 lignes suivantes :

```
define( 'WP_SITEURL', 'http://' . $_SERVER['SERVER_NAME'] . " " );
define( 'WP_HOME', 'http://' . $_SERVER['HTTP_HOST'] . " " );
```

De cette manière, le site est accessible depuis n'importe quelle source.

Astuce

Intégrer une vidéo Invidious

Pour intégrer une vidéo Invidious dans un article, il suffit d'ajouter un bloc HTML et d'y mettre le code suivant :


```
<iframe width="720" height="480" src="LIEN_INTEGRATION_INVIDIOUS;" frameborder="0"
allowfullscreen></iframe>
```

Remplacer LIEN_INTEGRATION_INVIDIOUS par le lien d'intégration de la vidéo.

Docker

Jellyfin

Jellyfin est une application de Streaming vidéo, musique.

Docker-compose :

```
services:
  jellyfin:
    image: lscr.io/linuxserver/jellyfin:latest
    container_name: jellyfin
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Europe/Paris
      - LANG=fr_FR.UTF-8
      - LANGUAGE=fr_FR.UTF-8
    volumes:
      - config:/config
      - /path/to/tvseries:/data/tvshows
      - /path/to/movies:/data/movies
    devices:
      - /dev/dri:/dev/dri
    ports:
      - 8096:8096
    restart: unless-stopped
volumes:
  config:
```

Pour les volumes, on peut soit les personnaliser dans le compose comme proposer, ou alors créer un autre point de montage.

Autre :

Lien XMLTV

Réglage du XMLTV :

Catégories de films :

Action|Drama|Drame|Familial|Fantastique|Fantasy|film : biographie|film : comédie|film : comédie dramatique|film : comédie musicale|film : comédie policière|film : comédie romantique|film : comédie sentimentale|film : court métrage|film : court métrage burlesque|film : court métrage d'animation|film : court métrage dramatique|film : drame|film : essai|film : mélodrame|film : péplum|film : thriller|film : thriller politique|film : western|film à sketches|film catastrophe|film d'action|film d'animation|film d'aventures|film de guerre|film de science-fiction|film de suspense|film d'épouvante|film d'espionnage|film d'horreur|film documentaire|film érotique|film fantastique|film historique|film noir|film policier|film pornographique|film pour la jeunesse|Thriller|Téléfilm|téléfilm : unknown|téléfilm d'action|téléfilm d'animation|téléfilm d'aventures|téléfilm de science-fiction|téléfilm de suspense|téléfilm d'horreur|téléfilm dramatique|téléfilm érotique|téléfilm fantastique|téléfilm historique|téléfilm humoristique|téléfilm policier|téléfilm sentimental|Romance|Science-Fiction|Science-Fiction & Fantastique|Sci-Fi|Action & Adventure|Adventure|Animation|Aventure|Comédie|Comedy|Crime|Fantastique|Fantasy

Catégories jeunesse :

jeunesse : dessin animé dessin animé|jeunesse : dessin animé jeunesse|jeunesse : dessin animé manga|jeunesse : émission jeunesse

Catégories des informations :

news|journalism|documentary|current affairs|journal|documentaire|info|information|informations|magazine|actualité|magazine animalier|magazine culinaire|magazine culturel|magazine d'actualité|magazine de découvertes|magazine de géopolitique|magazine de la décoration|magazine de la gastronomie|magazine de la mer|magazine de la mode|magazine de la nature|magazine de la santé|magazine de l'art de vivre|magazine de l'automobile|magazine de l'économie|magazine de l'environnement|magazine de reportages|magazine de services|magazine de société|magazine de télé-achat|magazine des beaux-arts|magazine d'information|magazine du cinéma|magazine du consommateur|magazine du court métrage|magazine du jardinage|magazine du jeu vidéo|magazine du multimédia|magazine du show-biz|magazine du

tourisme|magazine éducatif|magazine historique|magazine jeunesse|magazine judiciaire|magazine littéraire|magazine musical|magazine politique|magazine pornographique|magazine régional|magazine religieux|magazine scientifique|magazine sportif|Documentaire|documentaire animalier|documentaire art de vivre|documentaire autre|documentaire aventures|documentaire beaux-arts|documentaire chasse|documentaire cinéma|documentaire civilisations|documentaire culture|documentaire découvertes|documentaire économie|documentaire environnement|documentaire fiction|documentaire gastronomie|documentaire histoire|documentaire justice|documentaire médias|documentaire multimédia|documentaire musique|documentaire nature|documentaire pêche|documentaire politique|documentaire religions|documentaire santé|documentaire sciences et technique|documentaire société|documentaire sport|documentaire télé réalité|documentaire voyage|Documentary

Catégories des sports :

Sport|sport : automobilisme|sport : baseball|sport : basket-ball|sport : boxe|sport : catch|sport : cérémonie|sport : cyclisme|sport : équitation|sport : escalade|sport : fitness|sport : football|sport : formule 1|sport : formule 2|sport : formule 3|sport : formule e|sport : golf|sport : handball|sport : handisport|sport : hippisme|sport : hockey sur glace|sport : indycar|sport : jt sport|sport : judo|sport : motocross|sport : motocyclisme|sport : multisports|sport : rallye|sport : rugby|sport : rugby à xiii|sport : snooker|sport : speedway|sport : tennis|sport : tir à l'arc|sport : trail|sport : triathlon|sport : voile|sport : voitures de tourisme|sport : vtt|sports extrêmes|sports mécaniques

Collabora

Collabora Online est une puissante suite Office collaborative qui prend en charge tous les principaux formats de fichiers, de documents, de feuilles de calcul et de présentation, que vous pouvez intégrer dans votre propre infrastructure. Collabora Online offre la sécurité et la souveraineté des données et est parfaitement adapté aux exigences d'un environnement de travail distribué moderne. Offrant une apparence et une sensation familière, Collabora Online représente une véritable alternative aux autres solutions de grandes marques, vous donnant contrôle et flexibilité.

Compose

Voici le compose pour collabora :

```
version: "2.1"
services:
  collabora:
    image: collabora/code
    container_name: collabora
    environment:
      - username=utilisateur
      - password=mot-de-passe
    volumes:
      - /etc/localtime:/etc/localtime:ro
    ports:
      - 9980:9980
    restart: unless-stopped
```

Ne pas oublier de créer un sous-domaine.

Extension

Coter Nextcloud, il faut installer le connecteur <https://apps.nextcloud.com/apps/richdocuments>

Home Assistant

Installation :

Pour le compose, il faut que Home Assistant ait accès à tout le réseau local pour pouvoir détecter et gérer vos différents équipements connecter.

Voici le compose :

```
version: '3.7'

services:
  homeassistant:
    image: homeassistant/home-assistant:latest
    container_name: home-assistant
    restart: unless-stopped
    volumes:
      - config:/config
    network_mode: host
    environment:
      - TZ=Europe/Paris
      - PUID=1000
      - PGID=1000
volumes:
  config:
```

Configuration :

Interface de configuration :

Une fois déployé, on peut se connecter via l'adresse <http://ip:8123>

Une fois sur l'interface, on peut créer le compte et les différentes informations.

Changement du port :

Pour changer le port, rendez-vous dans le fichier `configuration.yml`.

Ajouter la ligne suivante en respectant les espaces et bien sûr le port souhaité :

```
#port
server_port: 9000
```

Serveur Proxy :

Si vous passez par un serveur proxy comme Nginx Proxy Manager, il faut ajouter les lignes suivantes en précisant l'adresse IP du proxy :

```
# Nginx proxy manager
http:
  use_x_forwarded_for: true
  trusted_proxies:
    - 192.168.1.2
    - 172.0.0.2
```

Restauration d'un autre Home Assistant :

Il suffit de copier et restaurer le contenu du dossier racine. Dans le doute, faire une sauvegarde de ce dossier avant.

```
automations.yaml
backups
blueprints
.cloud
configuration.yaml
deps
.HA_VERSION
home-assistant.log.fault
home-assistant_v2.db
home-assistant_v2.db-shm
home-assistant_v2.db-wal
```



```
scenes.yaml
scripts.yaml
secrets.yaml
.storage
tts
```

Carte

Pour rajouter une image en vue de créer une carte, il faut créer un répertoire `www` dans `/config`, lors de la configuration, il faudra renseigner le chemin `/local` pour `/config/www`.

Docker

Mealie

Un gestionnaire de recettes et un planificateur de repas auto-hébergés avec un backend RestAPI et une application frontale réactive construite en Vue pour une expérience utilisateur agréable pour toute la famille.



Voici un docker-compose :

```
version: "3.1"

services:
  mealie:
    container_name: mealie
    image: hkotel/mealie:latest
    restart: unless-stopped
    ports:
      - 9925:80
    environment:
      PUID: 1000
      PGID: 1000
```

TZ: Europe/Paris

Est utiliser pour les notifications

BASE_URL: http://localhost:8080

Configuration

Rendre les recettes public

RECIPE_PUBLIC: 'true'

Afficher les valeurs nutritionnel

RECIPE_SHOW_NUTRITION: 'true'

???

RECIPE_SHOW_ASSETS: 'true'

Définire la vue en paysage

RECIPE_LANDSCAPE_VIEW: 'true'

Désactiver les commentaires

RECIPE_DISABLE_COMMENTS: 'false'

???

RECIPE_DISABLE_AMOUNT: 'false'

Autres configurations possible

Unicorn

WEB_CONCURRENCY: 2

WORKERS_PER_CORE: 0.5

MAX_WORKERS: 8

volumes:

- app-data:/app/data

volumes:

app-data:

Pour la première connexion, le compte est :

Utilisateur :

Mot de passe :

Wireguard avec WebUI

Docker-compose

Voici un compose pour WireGuard avec une interface web :

(Explication pour le mot de passe depuis la version 14 plus bas).

```
version: '3.3'
services:
  wg-easy:
    container_name: wireguard-ui
    cap_add:
      - NET_ADMIN
      - SYS_MODULE
    environment:
      - WG_HOST=ip.public
      - WG_PORT=port.udp.externe.du.conteneur
      - PASSWORD_HASH=$$2b$$12$$coPqCsPtcFO.Ab99xyIBNOW4.lu7OOA2/ZlboHN6/oyxca3MWo7fW
      - WG_PERSISTENT_KEEPALIVE=20
      - WG_ALLOWED_IPS=0.0.0.0/0, ::/0
      - WG_DEFAULT_DNS=1.1.1.1, 1.0.0.1
    volumes:
      - data:/etc/wireguard
    ports:
      - 51820:51820/udp
      - 51821:51821/tcp
    sysctls:
      - net.ipv4.conf.all.src_valid_mark=1
      - net.ipv4.ip_forward=1
    restart: unless-stopped
    image: ghcr.io/wg-easy/wg-easy:latest
  volumes:
    data:
```

Ne pas oublier d'ouvrir de port **udp** du routeur.

Il est important en cas d'utilisation d'un serveur DNS en docker, de créer un réseau commun au 2 conteneurs et de mettre l'ip interne du docker.

Exemple :

```
version: '3.3'
services:
  wg-easy:
    container_name: wireguard-ui
    cap_add:
      - NET_ADMIN
      - SYS_MODULE
    environment:
      - WG_HOST=ip.public
      - WG_PORT=port.udp.externe.du.conteneur
      - PASSWORD_HASH=$$2b$$12$$coPqCsPtcFO.Ab99xyIBNOW4.lu7OOA2/ZIboHN6/oyxca3MWo7fW
      - WG_PERSISTENT_KEEPALIVE=20
      - WG_ALLOWED_IPS=0.0.0.0/0, ::/0
      - WG_DEFAULT_DNS=192.168.12.2
    networks:
      - dns-network
    volumes:
      - data:/etc/wireguard
    ports:
      - 51820:51820/udp
      - 51821:51821/tcp
    sysctls:
      - net.ipv4.conf.all.src_valid_mark=1
      - net.ipv4.ip_forward=1
    restart: unless-stopped
    image: ghcr.io/wg-easy/wg-easy:latest
volumes:
  data:
networks:
  dns-network:
    name: dns-network
    driver: bridge
```

Génération de mot de passe.

Depuis la version 14 la variable d'environnement `PASSWORD` a été remplacé par `PASSWORD_HASH`. Un outil est mis à disposition dans le conteneur pour avoir un mot de passe brouillé en partant d'un mot de passe ne clair.

Une fois en `BASH` dans le conteneur, lancer la commande suivante en remplaçant `mon_mot_de_passe` par le mot de passe désiré :

```
wgpw mon_mot_de_passe
```

Ensuite, on obtient un résultat sous la forme suivante :

```
PASSWORD_HASH='$2b$12$coPqCsPtcFO.Ab99xyIBNOW4.lu7OOA2/ZIboHN6/oyxca3MWo7fW'
```

Il y a une subtilité, comme nous utilisons un docker-compose, il faut retirer les simples côtes (') à l'avant et arrière et également rajouter un \$ supplémentaire à chaque \$ présent.

Ce qui donne :

```
PASSWORD_HASH=$$2b$$12$$coPqCsPtcFO.Ab99xyIBNOW4.lu7OOA2/ZIboHN6/oyxca3MWo7fW
```

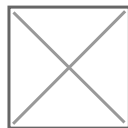
Documentation officielle : https://github.com/wg-easy/wg-easy/blob/master/How_to_generate_an_bcrypt_hash.md

Cloud-commander

Cloud-commander, est un service qui sert de gestionnaire de fichier web, il a l'avantage d'être en 2 colonnes, très pratique pour déplacer des fichiers d'un endroit à l'autre sans devoir changer de dossier continuellement.

Il intègre une visionneuse d'image ainsi qu'un éditeur de texte, un terminal est également intégré.

Interface :



Docker-compose :

Voici un exemple de mon docker-compose, je crée un volume pour la sauvegarde des préférences :

```
version: '2'
services:
  cloudcmd:
    container_name: cloud-commander
    image: coderaiser/cloudcmd
    #environment:
    ports:
      - 8000:8000
    volumes:
      - ./srv
      - config:/root
    restart: unless-stopped
volumes:
  config:
```

Bien sûr, on peut rajouter des volumes qui pointe vers un endroit particulier du système de fichier.

Il y a également des variables d'environnement que l'on peut rajouter et qui sont détaillées sur le site officiel cloudcmd.io.

Pi-Hole

Docker-compose :

```
version: "3"

# More info at https://github.com/pi-hole/docker-pi-hole/ and https://docs.pi-hole.net/
services:
  pihole:
    container_name: pihole
    image: pihole/pihole:latest
    # For DHCP it is recommended to remove these ports and instead add: network_mode: "host"
    # network_mode: host
    ports:
      - 53:53/tcp
      - 53:53/udp
      #- 67:67/udp # Only required if you are using Pi-hole as your DHCP server
      - 56880:80/tcp
    environment:
      TZ: Europe/Paris
      WEBPASSWORD: password
      WEBTHEME: default-darker
      # WEB_PORT: 5400 # En cas de networkmode: host
    # Volumes store your data between container upgrades
    volumes:
      - data:/etc/pihole
      - dnsmasq:/etc/dnsmasq.d
    # https://github.com/pi-hole/docker-pi-hole#note-on-capabilities
    #cap_add:
    # - NET_ADMIN # Required if you are using Pi-hole as your DHCP server, else not needed
    restart: unless-stopped

volumes:
  data:
```

dnsmasq:

Liste de blocage :

Liste par défaut :

<https://raw.githubusercontent.com/StevenBlack/hosts/master/hosts>

Liste que j'ai rajouté :

<https://easylist.to/easylist/easylist.txt> ☐ easylist
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/filters-2020.txt> ☐ Ublock filters
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/filters-2021.txt> ☐ Ublock filters
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/filters-2022.txt> ☐ Ublock filters
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/filters-2023.txt> ☐ Ublock filters
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/filters.txt> ☐ Ublock filters
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/badware.txt> ☐ Ublock Badware
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/privacy.txt> ☐ Ublock Privacy
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/quick-fixes.txt> ☐ Ublock Quick Fixes
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/resource-abuse.txt> ☐ Ublock Resource-abuse
<https://github.com/uBlockOrigin/uAssets/raw/master/filters/unbreak.txt> ☐ Ublock unbreak
<https://easylist.to/easylist/easyprivacy.txt> ☐ easyprivacy
<https://malware-filter.gitlab.io/malware-filter/urlhaus-filter-domains.txt> ☐ Online Malicious URL Blocklist
<https://pgl.yoyo.org/adserver/serverlist.php?hostformat=hosts> ☐ Ublock Peter Lowe's Ad and tracking server list

Wireguard :

Comme expliquer dans WireGuard, voici le compose pour Pi-Hole :

version: "3"

More info at <https://github.com/pi-hole/docker-pi-hole/> and <https://docs.pi-hole.net/>

services:

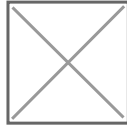
pihole:

```
container_name: pihole
image: pihole/pihole:latest
# For DHCP it is recommended to remove these ports and instead add: network_mode: "host"
# network_mode: host
ports:
  - 53:53/tcp
  - 53:53/udp
# - 67:67/udp # Only required if you are using Pi-hole as your DHCP server
  - 56880:80/tcp
environment:
  TZ: Europe/Paris
  WEBPASSWORD: password
  WEBTHEME: default-darker
# WEB_PORT: 5400 # En cas d'utilisation de network_mode: host
# Volumes store your data between container upgrades
networks:
  dns_network:
    ipv4_address: 192.168.12.2
volumes:
  - data:/etc/pihole
  - dnsmasq:/etc/dnsmasq.d
# https://github.com/pi-hole/docker-pi-hole#note-on-capabilities
#cap_add:
# - NET_ADMIN # Required if you are using Pi-hole as your DHCP server, else not needed
restart: unless-stopped

volumes:
  data:
  dnsmasq:
networks:
  dns_network:
    name: dns_network
    driver: bridge
  ipam:
    config:
      - subnet: 192.168.12.0/24
```

Restreamer

Restreamer est une application qui est un service autohébergé de streaming comme Twitch, il a l'avantage de pouvoir proposer une interface web et de pouvoir renvoyer le stream sur d'autre plate-forme.



Docker-compose :

```
version: '3.3'
services:
  restreamer:
    restart: unless-stopped
    container_name: restreamer
    ports:
      - 8080:8080
      - 8181:8181
      - 1935:1935
      - 6000:6000/udp
    volumes:
      - config:/core/config
      - data:/core/data
    privileged: false
    image: datarhei/restreamer:latest
volumes:
  config:
  data:
```

Ne pas oublier de personnaliser les ports externes.

[Github](#)

Docker

Pyload

Pyload est un gestionnaire de téléchargement via interface web.

Pyload est la version stable.

Compose :

```
version: "2.1"
services:
  pyload:
    image: lscr.io/linuxserver/pyload:latest
    container_name: pyload
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Europe/Paris
    volumes:
      - config:/config
      - downloads:/downloads
    ports:
      - 8000:8000
      # - 7227:7227 #optional
    restart: unless-stopped
volumes:
  config:
  downloads:
```

Compte par défaut:

Utilisateur :

mot de passe :

[Page Linuxserver](#)

Pyload-ng est la version de développement.

Compose :

version: "2.1"

services:

 payload:

 image: lscr.io/linuxserver/payload-ng:latest

 container_name: payload-ng

 environment:

 - PUID=1000

 - PGID=1000

 - TZ=Europe/Paris

 volumes:

 - config:/config

 - downloads:/downloads

 ports:

 - 8000:8000

- 7227:7227 #optional

 restart: unless-stopped

volumes:

 config:

 downloads:

[DockerHub](#)

Heimdall

Comme son nom l'indique, Heimdall Application Dashboard est un tableau de bord pour toutes vos applications web. Il n'a pas besoin d'être limité aux applications, vous pouvez ajouter des liens vers tout ce que vous voulez.

Heimdall est une solution élégante pour organiser toutes vos applications web. Il est dédié à cet objectif afin que vous ne perdiez pas vos liens dans une mer de signets.

Pourquoi ne pas l'utiliser comme page de démarrage de votre navigateur ? Il a même la capacité d'inclure une barre de recherche utilisant Google, Bing ou DuckDuckGo.

Docker-compose :

```
version: "2.1"
services:
  heimdall:
    image: lscr.io/linuxserver/heimdall:latest
    container_name: heimdall
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Europe/Paris
    volumes:
      - config:/config
    ports:
      - 80:80
      - 443:443 # port https
    restart: unless-stopped
volumes:
  config:
```

Adguard

AdGuard Home est un logiciel permettant de bloquer les publicités et le suivi sur l'ensemble du réseau. Après l'avoir configuré, il couvrira tous vos appareils domestiques, et vous n'aurez besoin d'aucun logiciel client pour cela. Pour en savoir plus, consultez notre dépôt Github officiel.

Docker-compose :

```
version: '3.3'
services:
  adguardhom:
    container_name: adguardhome
    restart: unless-stopped
    volumes:
      - workdir:/opt/adguardhome/work
      - confdir:/opt/adguardhome/conf
    ports:
      - 53:53/tcp
      - 53:53/udp
      - 67:67/udp # Serveur DHCP
      - 68:68/udp # Serveur DHCP
      - 5480:5480/tcp
      #- 443:443/tcp
      #- 443:443/udp
      - 3000:3000/tcp
      - 853:853/tcp
      - 784:784/udp
      - 853:853/udp
      - 8853:8853/udp
      - 5443:5443/tcp
      - 5443:5443/udp
    image: adguard/adguardhome:latest
volumes:
  workdir:
  confdir:
```


Avec la configuration supplémentaire pour WireGuard :

```
version: '3.3'
services:
  adguardhom:
    container_name: adguardhome
    restart: unless-stopped
    volumes:
      - workdir:/opt/adguardhome/work
      - confdir:/opt/adguardhome/conf
    networks:
      dns-network:
        ipv4_address: 192.168.12.2
    ports:
      - 53:53/tcp
      - 53:53/udp
      - 67:67/udp # Serveur DHCP
      - 68:68/udp # Serveur DHCP
      - 5480:5480/tcp
      #- 443:443/tcp
      #- 443:443/udp
      - 3000:3000/tcp
      - 853:853/tcp
      - 784:784/udp
      - 853:853/udp
      - 8853:8853/udp
      - 5443:5443/tcp
      - 5443:5443/udp
    image: adguard/adguardhome:latest
  volumes:
    workdir:
    confdir:
  networks:
    dns-network:
      name: dns-network
      driver: bridge
    ipam:
```

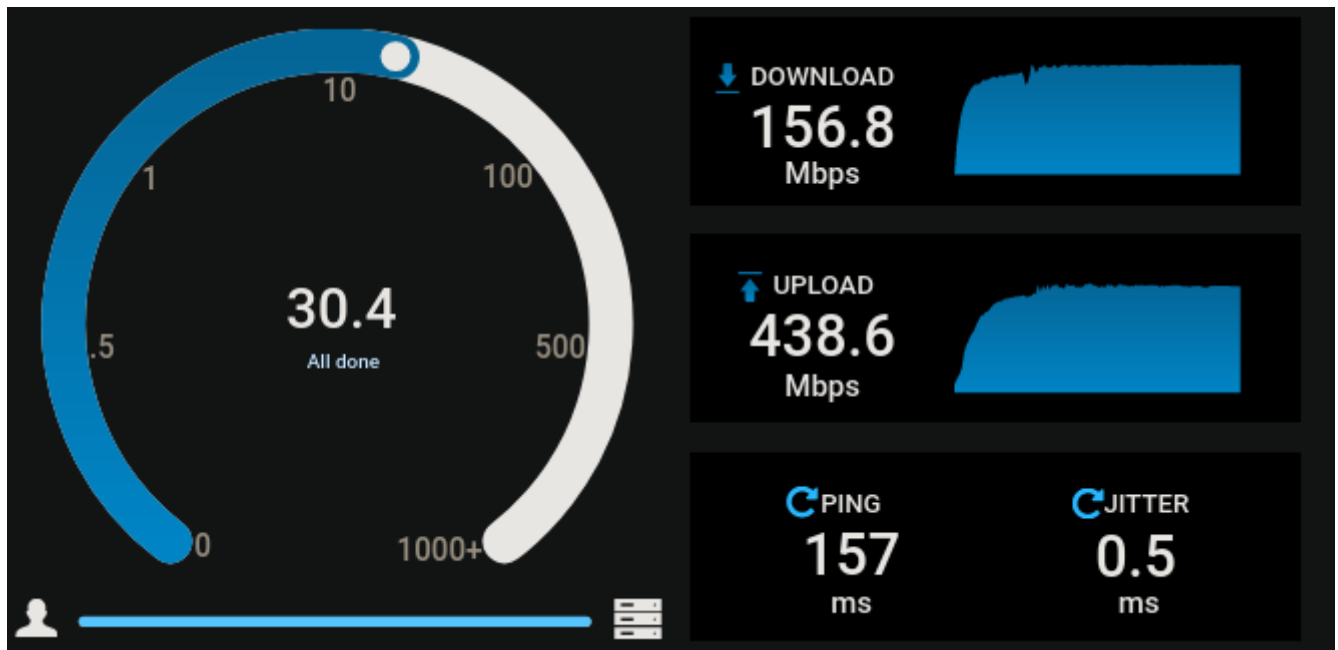
config:

- subnet: 192.168.12.0/24

Docker

OpenSpeedTest

OpenSpeedTest est un logiciel permettant de faire un test de vitesse entre un appareil et le serveur su lequel il est installé.

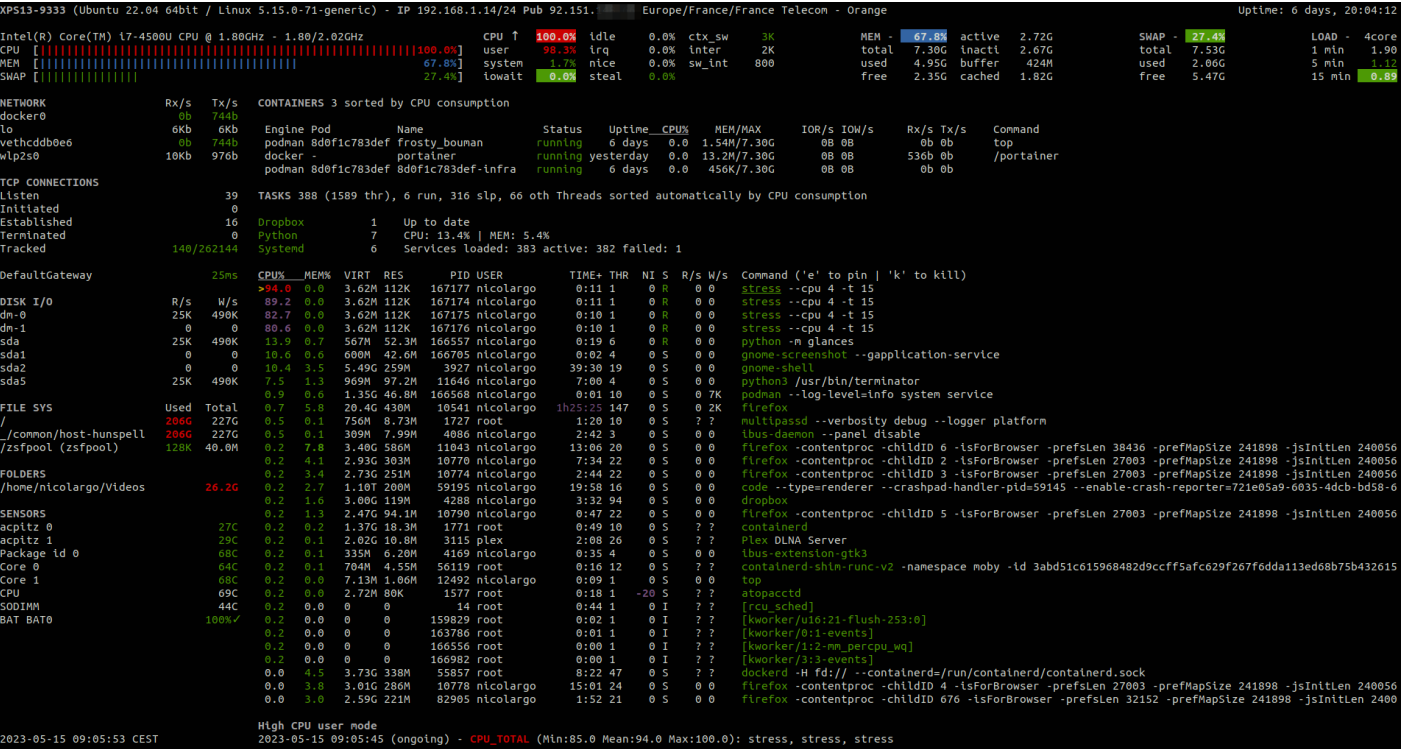


Voici le compose :

```
version: '3.3'
services:
  speedtest:
    restart: unless-stopped
    container_name: openspeedtest
    ports:
      - 3000:3000 # HTTP
      #- 3001:3001 # HTTPS
    volumes:
      - lestencrypt:/var/log/letsencrypt
    image: openspeedtest/latest
  volumes:
    lestencrypt:
# source https://github.com/openspeedtest/Docker-Image?tab=readme-ov-file#or-use-docker-compose.yml
```


Glances

Glances est un logiciel qui permet d'avoir un monitoring du serveur en temps réel.



Voici le compose :

services:

glances:

image: nicolargo/glances:latest-full

pid: host

ports:

- 61208:61208

volumes:

- /var/run/docker.sock:/var/run/docker.sock

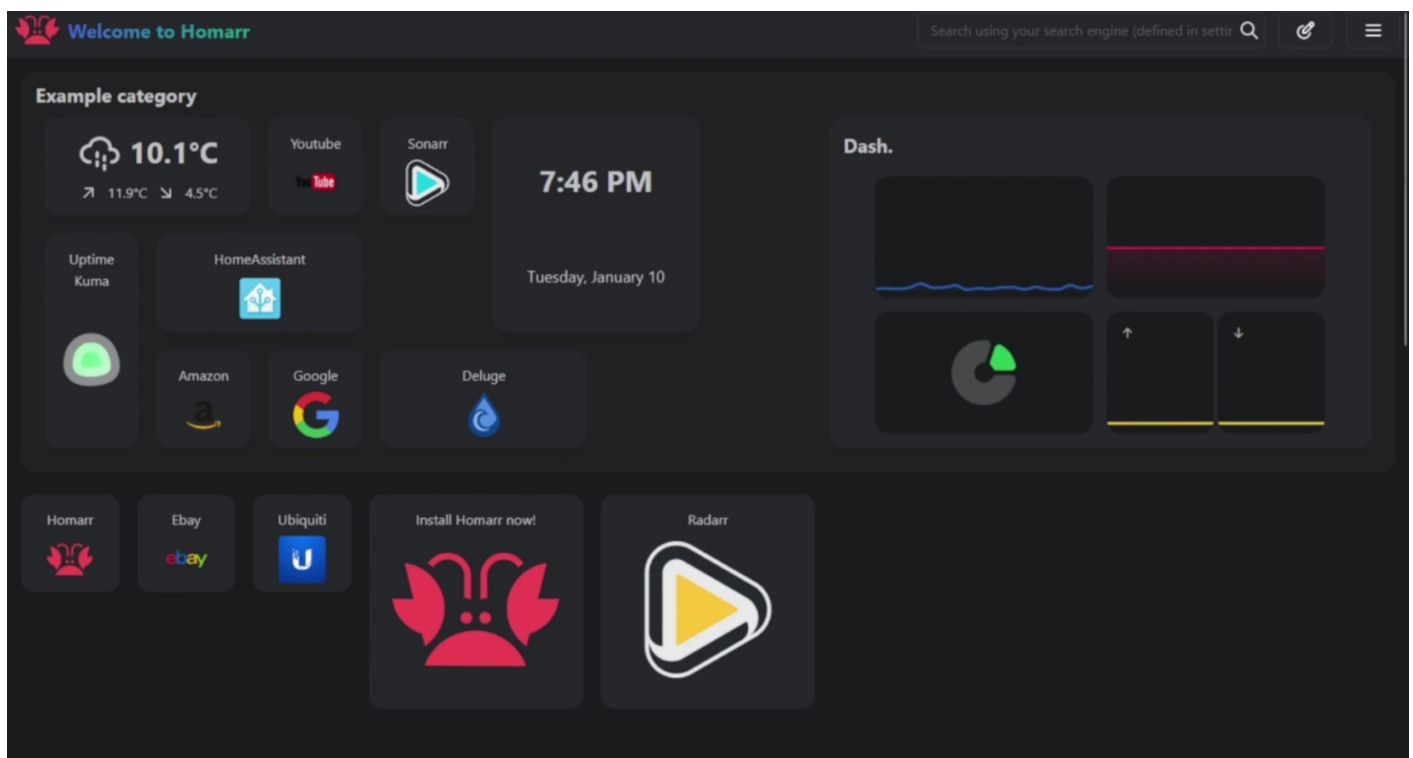
environment:

- "GLANCES_OPT=-w"

restart: unless-stopped

Homarr

Simplifiez la gestion de votre serveur avec Homarr, un tableau de bord élégant et moderne qui met toutes vos applications et services à portée de main. Avec Homarr, vous pouvez accéder et tout contrôler en un seul endroit pratique. Homarr s'intègre de manière transparente aux applications que vous avez ajoutées, vous fournissant des informations précieuses et vous donnant un contrôle total. L'installation est un jeu d'enfant et Homarr prend en charge un large éventail de méthodes de déploiement.



Voici le compose :

```
#-----#
#   Homarr - A simple, yet powerful dashboard for your server.   #
#-----#

services:
  homarr:
    container_name: homarr
    image: ghcr.io/homarr-labs/homarr:latest
    restart: unless-stopped
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock # Facultatif, uniquement si vous souhaitez l'intégration de
```

Docker.

- appdata:/appdata

environment:

- SECRET_ENCRYPTION_KEY=collez_votre_clé_de_64_caractères_ici

ports:

- '7576:7575'

volumes:

appdata:

⚠ Veillez à faire cette commande pour générer une clé de 64 caractères :

```
openssl rand -hex 32
```

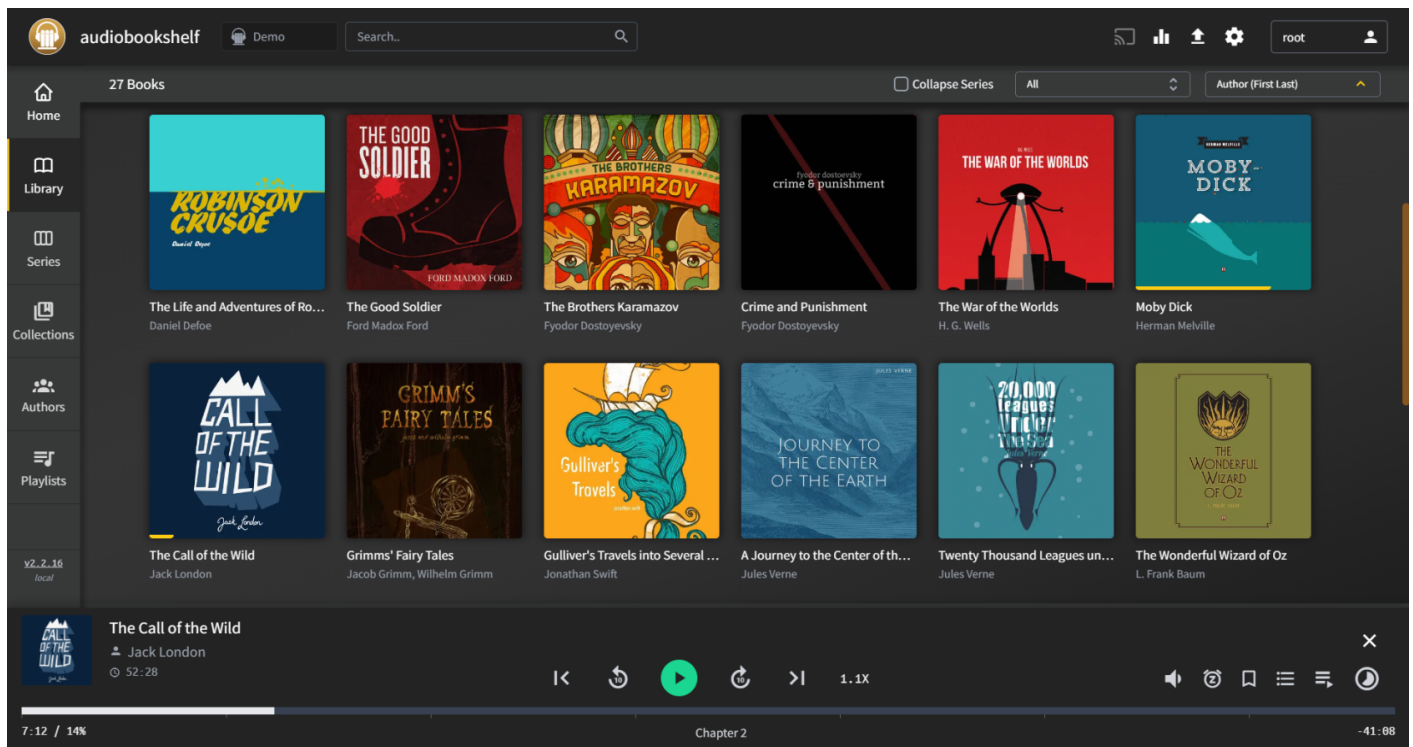
Source.

AudioBookShelf

Audiobookshelf est un serveur de livres audio et de podcasts auto-hébergé.

Caractéristiques

- Entièrement **open source**, y compris l'application [Android et iOS](#) (*en version bêta*)
- Diffusez tous les formats audio à la volée
- Recherchez et ajoutez des podcasts pour télécharger des épisodes avec téléchargement automatique
- Prise en charge multi-utilisateurs avec autorisations personnalisées
- Maintient la progression par utilisateur et se synchronise sur tous les appareils
- Détecte automatiquement les mises à jour de la bibliothèque, pas besoin de réanalyser
- Téléchargez des livres et des podcasts avec le téléchargement groupé de dossiers par glisser-déposer
- Sauvegardez vos métadonnées + sauvegardes quotidiennes automatisées
- Application Web progressive (PWA)
- Prise en charge de Chromecast sur l'application Web et l'application Android
- Récupérer des métadonnées et des pochettes à partir de plusieurs sources
- Éditeur de chapitre et recherche de chapitre (à l'aide de [l'API Audnexus](#))
- Fusionnez vos fichiers audio en un seul m4b
- Intégrez des métadonnées et une image de couverture dans vos fichiers audio (à l'aide [de Tone](#))
- Prise en charge de base des ebooks et des liseuses
 - Epub, pdf, cbr, cbz
 - Envoyer un ebook à un appareil (c'est-à-dire Kindle)
- Ouvrir les flux RSS pour les podcasts et les livres audio



Voici le compose :

version: "3.7"

services:

audiobookshelf:

image: ghcr.io/advplyr/audiobookshelf:latest

ports:

- 80:80

volumes:

- audiobooks:/audiobooks
- podcasts:/podcasts
- config:/config
- metadata:/metadata

restart: unless-stopped

volumes:

audiobooks:

podcasts:

config:

metadata:

[GitHub](#)

FreshRSS

FreshRSS est un agrégateur de flux RSS auto-hébergé.

Il est léger, facile à utiliser, puissant et personnalisable.

Il s'agit d'une application multi-utilisateur avec un mode de lecture anonyme. Il prend en charge les balises personnalisées. Il existe une API pour les clients (mobiles) et une interface de ligne de commande.


Grâce au standard WebSub, FreshRSS est capable de recevoir des notifications push instantanées de sources compatibles, telles que Friendica, WordPress, Blogger, Medium, etc.


FreshRSS prend en charge nativement le Web scraping de base, basé sur XPath, pour les sites Web ne fournissant aucun flux RSS/Atom.

Différentes méthodes de connexion sont prises en charge : formulaire Web (incluant une option anonyme), authentification HTTP (compatible avec la délégation proxy), OpenID Connect.

Enfin, FreshRSS prend en charge les extensions pour un réglage ultérieur.

- Site officiel : <https://freshrss.org>
- D  mo : <https://demo.freshrss.org>
- Licence : [GNU AGPL 3](#)



Subscriptions management 

≡ Main stream

★ Favourites (1)

Comics ▾

Development ▴

🔗 Docker Blog

🌐 Fedora Magazine

⚙️ ★ Mozilla Hacks - the Web d...

Filter

Statistics

See website

Manage


Actualize

Mark as read


🏠 📧 ★ ☆ ▾ Mark as read ▾ ≡ ⋮ 📖 📡 ⬆️ ↺

Today — September 25th 2014 Your RSS feeds

🏠 ☆ 🇸🇦 Sam & Max : Python, Djang... Obin a été migré


September 25th 2014 at 10:09 

🏠 ☆ 🇷🇺 Le blog des bits - Pascal Jeh... Plug-in flash


September 25th 2014 at 00:00 

Yesterday — September 24th 2014 Your RSS feeds


🏠 ★ 🐋 Docker Blog Docker Hub Official Repos: Announcing Language Stacks

September 24th 2014 at 23:21 


🏠 ☆ 🌐 Fedora Magazine GNOME 3.14 released — all this goodness will be in Fedora 21

September 24th 2014 at 20:44 

🏠 ☆ 🌐 Fedora Magazine Flaw CVE-2014-6271 discovered in the Bash shell — update your Fedora systems


September 24th 2014 at 17:45 

🏠 ☆ 🇸🇦 Sam & Max : Python, Djang... Un bot qui tweete 4992 Super-Héros à raison de un toutes les 3,14 heures peut-il sauver Internet ?


September 24th 2014 at 08:43 

Before yesterday Your RSS feeds


🏠 ☆ 🌐 Fedora Magazine Fedora 21 Alpha released

September 23rd 2014 at 17:24 


🏠 ☆ 🇸🇦 Sam & Max : Python, Djang... Le blog a été migré (enfin presque)

September 23rd 2014 at 14:18 


🏠 ☆ 🇷🇺 Le blog des bits - Pascal Jeh... Mise en veille

September 23rd 2014 at 00:00 


🏠 ☆ 🐋 Docker Blog Report: Burrito Quest I

September 22nd 2014 at 18:29 


🏠 ☆ 🌐 Fedora Magazine Florida is back on the Map for Linux and Open Source conventions with FOSSETCON 2

September 20th 2014 at 21:10 


🏠 ☆ 🌐 Fedora Magazine Must-have GNOME extension: gTile

September 19th 2014 at 13:45 

🏠 ☆ ★ Mozilla Hacks - the Web de... Low price smartphones - memory management and optimization on Firefox OS

September 19th 2014 at 13:44 

🏠 ☆ 🌐 Fedora Magazine F21 Alpha on Tuesday, Fedora UK podcast, Flock bids, Fedora store, and badges!

September 18th 2014 at 23:34 

< ▴ >

Voici le compose :

```

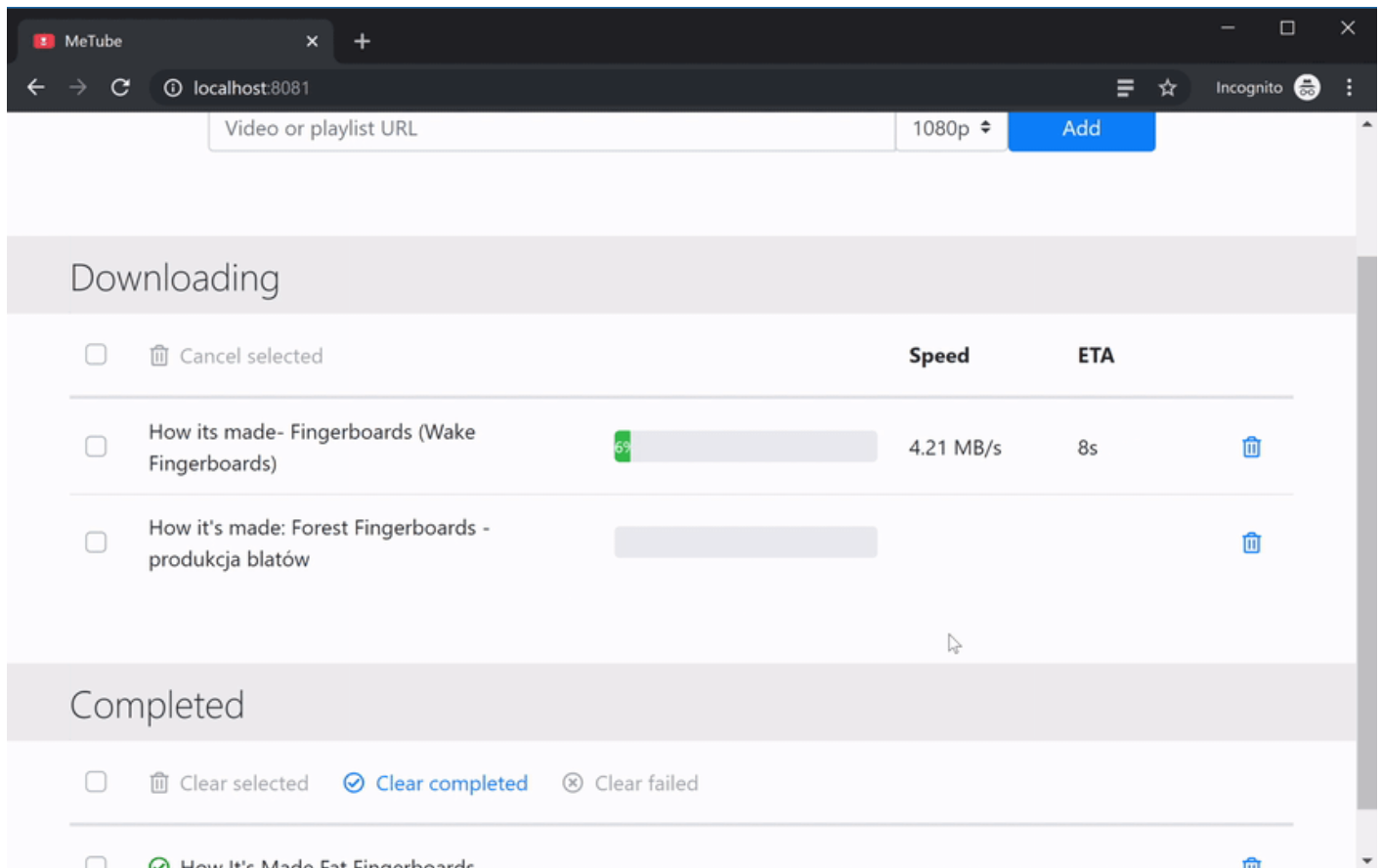
version: "2.1"
services:
  freshrss:
    image: lscr.io/linuxserver/freshrss:latest
    container_name: freshrss
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Europe/Paris
    volumes:
      - config:/config
    ports:
      - 80:80
    restart: unless-stopped
volumes:
  config:

```

Docker

MeTube

MeTube permet de télécharger des vidéos YouTube, aussi bien l'audio que la vidéo.



Compose :

```
version: "3"
services:
  metube:
    image: ghcr.io/alexta69/metube
    container_name: metube
    restart: unless-stopped
    ports:
      - "8081:8081"
    volumes:
      - /path/to/downloads:/downloads
```

Source

Docker

Vaultwarden

Vaultwarden est la partie serveur auto-héberger pour Bitwarden, un gestionnaire de mot de passe.

Voici le compose :

```
version: '3.7'

services:
  vaultwarden:
    image: vaultwarden/server:latest
    container_name: vaultwarden
    ports:
      - 80:80
    environment:
      - WEBSOCKET_ENABLED=true # Enable WebSocket notifications.
      - SIGNUPS_ALLOWED=true
      - WEB_VAULT_ENABLED=true
      - INVITATIONS_ALLOWED=false
    volumes:
      - data:/data
    restart: unless-stopped
volumes:
  data:
```

Dans le cas où l'interface admin est activé et des paramètres changés, les variables d'environnement du docker-compose ne sont plus pris en compte, les paramètres peuvent être édités dans le fichier `config.json` qui se trouve dans `/data`.

[GitHub](#)

Yourls Raccourcisseur de lien

Yourls est un raccourcisseur de lien.

Le docker-compose :

```
version: '3.1'
services:
  yourls:
    image: yourls
    restart: unless-stopped
    ports:
      - 8080:80
    environment:
      YOURLS_DB_PASS: example
      YOURLS_SITE: https://example.com
      YOURLS_USER: example_username
      YOURLS_PASS: example_password

  mysql:
    image: mysql
    restart: unless-stopped
    volumes:
      - mysql:/var/lib/mysql
    environment:
      MYSQL_ROOT_PASSWORD: example
      MYSQL_DATABASE: yourls
volumes:
  mysql:
```

Modifier bien sûr les infos des variables d'environnement.

Pour y accéder, il faut rajouter `/admin` à la fin du lien, exemple : `http://localhost:8080/admin/`, ou `http://<host-ip>:8080/admin/`.

Il y a également une extension pour le navigateur web : [Firefox](#), [Chromium](#).

[Hub-Docker](#)

Docker

Nginx-RTMP

Nginx-RTMP permet de streamer un flux vidéo sur le réseau.

```
version: "2.1"
services:
  nginx-rtmp:
    image: tiangolo/nginx-rtmp
    container_name: nginx-rtmp
    ports:
      - 1935:1935
    restart: unless-stopped
```

Pour steamer via OBS, il suffit d'entrer l'adresse suivante : `rtmp://ip:1935/live`

Pou lire le flux via VLC, il faut ouvrir un flux réseau et indiqué la même adresse que pour OBS :

`rtmp://ip:1935/live`

Docker

Srt-live-server

Srt-live-serveur permet de faire du streaming, mais via l'UDP.

Docker-compose x86_64 :

```
# https://hub.docker.com/r/baluzs/srt-live-server
version: '3'
services:
  srt:
    image: baluzs/srt-live-server
    ports:
      - 1935:1935/udp
    restart: unless-stopped
    volumes:
      - sls:/etc/sls
      - logs:/logs
volumes:
  logs:
  sls:
```

Docker-compose ARM64 :

```
# https://github.com/ravenium/srt-live-server
version: '3'
services:
  srt:
    image: ravenium/srt-live-server:latest
    ports:
      - 1935:1935/udp
    restart: unless-stopped
    volumes:
      - logs:/logs
volumes:
  logs:
```

Pour l'utilisation :

Coté OBS : `srt://ip.serveur:port?streamid=input/live/lenomdustream`

Coté VLC : `srt://ip.serveur:port?streamid=output/live/lenomdustream`

Docker

Filegator

FileGator est une application Web gratuite, open source et auto-hébergée pour gérer des fichiers et des dossiers.



[Site Officiel](#)

Docker-compose.

```
services:
  filegator:
    build: .
    image: filegator/filegator:latest
    restart: unless-stopped
    ports:
      - 8080:8080
    volumes:
      - repository:/var/www/filegator/repository
      - private:/var/www/filegator/private
volumes:
  private:
  repository:
```

Des volumes peuvent être rajouter, ils doivent l'etre dans `/var/www/filegator/repository` du conteneur.

Le compte par défaut est :

Utilisateur : `admin`

Mot de passe : `admin123`

Stirling-PDF

Stirling-PDF est une application web locale qui permet de manipuler des fichiers PDF. Elle est basée sur Docker et offre de nombreuses fonctionnalités, notamment la division, la fusion, la conversion, la réorganisation, l'ajout d'images, la rotation, la compression et bien plus encore.

Cette application est sécurisée et ne stocke aucune donnée à l'extérieur. Les fichiers et PDF sont soit stockés sur votre appareil, soit conservés en mémoire serveur uniquement pendant l'exécution de la tâche, soit stockés temporairement dans un fichier durant l'exécution de la tâche. Les fichiers téléchargés sont supprimés du serveur une fois la tâche terminée.

```
version: '3.3'
services:
  stirling-pdf:
    image: frooodle/s-pdf:latest
    restart: unless-stopped
    network_mode: bridge
    ports:
      - 8080:8080
    volumes:
      - trainingData:/usr/share/tesseract-ocr/5/tessdata #Required for extra OCR languages
      - extraConfigs:/configs
#   - /location/of/customFiles:/customFiles/
#   - /location/of/logs:/logs/
    environment:
      - DOCKER_ENABLE_SECURITY=false
volumes:
  trainingData:
  extraConfigs:
```

[Github](#)

Joal

Il fait croire aux trackers torrent que vous partagez des fichiers pour augmenter votre ratio et tricher (alternative à ratiomaster).



Le Docker compose :

```
services:
  joal:
    image: anthonyraymond/joal:latest
    container_name: joal
    restart: unless-stopped
    volumes:
      - data:/data
    ports:
      - 8080:8080
    command:
      - "--joal-conf=/data"
      - "--spring.main.web-environment=true"
      - "--server.port=8080"
      - "--joal.ui.path.prefix=joal"
      - "--joal.ui.secret-token=YourToken"
    volumes:
      data:
```

La ligne 15 token correspond au mot de passe de la page d'administration, remplacer YourToken par un mot de passe.

Une fois le conteneur lancer la première fois, il faut l'arrêter et placer le contenu de l'archive [joal.tar.gz](#) dans le volume `data`.

L'accès de la page de Joal se fait via le lien `http://ip:8080/joal/ui`, il faut bien sûr remplacer ip par celui du serveur.