

PLEX on Ubuntu virtual machine (ARM64)

82.64.87.133

Création de la VM :

2c

1gb RAM

30GB disque

Accès au disque de la Freebox coché

User = plex

Password à renseigner

Debian !

Premier lancement de la VM :

- Connexion avec le user ;
- ls -l /mnt/SSD/
- sudo mkdir /mnt/NAS
- ls -l /mnt/NAS/
- sudo apt-get update && sudo apt-get upgrade && sudo apt-get dist-upgrade
- Installer curl :
 - sudo apt-get install curl
- Installer les outils CIFS (client SMB...)
 - sudo apt-get install cifs-utils
- Installer gnupg :
 - sudo apt-get install gnupg
- Créer un nouveau fichier qui contiendra les credentials d'accès au NAS :
 - sudo vi /home/plex/.smbcredentials
 - *Contenu à renseigner :*
 - username=plex
 - password=...
 - Enregistrer les modifications
- Tant que notre VM tourne, Identifier l'IP et l'adresse MAC de notre VM, et définir un baux statique car nous utiliserons un vrai client SSH et pas la booze de console bugguée (la mienne sera 192.168.1.99 pour la VM)
- Eteindre la VM ;
- Dans la configuration cloud-init, remplacer le bloc "mounts" par celui-ci :
 - *mounts:*
 - - ['//192.168.1.254/SSD', '/mnt/SSD', cifs, 'vers=1.0,guest,uid=1000,gid=1000', '0', '0']
 - - ['//192.168.1.100/Commun', '/mnt/NAS', cifs, '_netdev,vers=3.0,credentials=/home/

plex/.smbcredentials,iocharset=utf8,uid=1000,gid=1000', '0', '0']

- Relancer la VM sans ouvrir la console proposée sur la Freebox et patienter 20 secondes ;
- Se connecter avec un vrai client SSH (Putty, MacOS, Linux, ...) en lançant la commande :
 - ssh 192.168.1.99 -l plex
 - Saisir "yes"
 - Saisir le mot de passe associé au user Plex
 - Lancer la commande pour reconfigurer la bonne locale :
 - sudo dpkg-reconfigure locales
 - Descendre dans la liste jusqu'à se positionner sur la fr_FR.UTF-8 UTF-8
 - Appuyer une fois sur la touche ESPACE pour cocher la locale et appuyer une fois sur la touche TAB, puis ENTREE
 - Sélectionner à nouvelle la même locale en tant que défaut locale
 - Puis faire en sorte que le message ne soit plus affiché en lançant la commande :
 - sudo touch /var/lib/cloud/instance/locale-check.skip
- Quitter la connexion SSH et relancer une nouvelle connexion :
 - exit
 - ssh 192.168.1.99 -l plex
 - Saisir le mot de passe...
- Editer le fstab pour monter automatiquement les volumes SSD et NAS :
 - sudo vi /etc/fstab
 - Vérifier que vous avez bien vos deux lignes avec exactement la même chose que votre config-cloud dans l'interface VM de la Freebox ;
 - Il se peut que la casse soit perdue (lettres en minuscule alors que vous aviez mis des majuscules dans la cloud-config), alors procéder aux corrections si besoin ;
- Enregistrer les modifications ;
- Redémarrer la VM :
 - sudo shutdown -r now
- Vérifier si nos volumes sont bien montés :
 - ls -l /mnt/SSD
 - ls -l /mnt/NAS

Installation de Plex :

- Se rendre sur le site <https://www.plex.tv/media-server-downloads/>
- Sélectionner Linux dans Plex Media Server puis Choose Distribution > ARMv8 > un téléchargement débutera mais l'idée était surtout de récupérer le lien de la dernière version de Plex, en l'occurrence :
 - https://downloads.plex.tv/plex-media-server-new/1.19.1.2645ccb6eb67e/debian/plexmediaserver_1.19.1.2645ccb6eb67e_arm64.deb
 - Ce qui signifie que nous allons installer ce package deb depuis la VM directement !

- sudo wget https://downloads.plex.tv/plex-media-server-new/1.19.1.2645ccb6eb67e/debian/plexmediaserver_1.19.1.2645ccb6eb67e_arm64.deb
- Installer le package :
 - sudo dpkg -i [plexmediaserver_1.19.1.2645ccb6eb67e_arm64.deb](https://downloads.plex.tv/plex-media-server-new/1.19.1.2645ccb6eb67e/debian/plexmediaserver_1.19.1.2645ccb6eb67e_arm64.deb)
 - Identifier le metada dir parmi les informations affichées après la fin de l'installation : /var/lib/plexmediaserver/Library/Application Support
 - Idem pour le Created symlink : /lib/systemd/system/plexmediaserver.service
 - Vérifier si Plex tourne :
 - sudo systemctl status plexmediaserver
 - Vous devriez voir un truc comme "Active: **active (running)**"
- Activer la mise à jour simplifiée de Plex :
 - sudo vi /etc/apt/sources.list.d/plexmediaserver.list
 - Décommenter la ligne "#deb ..." en retirant le "#" puis enregistrer les modifications ;
 - Puis importer la clé publique de Plex :
 - sudo wget -q <https://downloads.plex.tv/plex-keys/PlexSign.key> -O - | sudo apt-key add -
 - Vous devriez obtenir un "OK".
- Si vous vous apprêtez à configurer Plex depuis un autre réseau de celui de la VM (via VPN par exemple), vous devez ajouter la plage d'adresse des IP VPN dans la configuration de Plex :
 - Commencer par stopper Plex :
 - sudo systemctl stop plexmediaserver
 - Editer la configuration de Plex :
 - sudo vi /var/lib/plexmediaserver/Library/Application\ Support/Plex\ Media\ Server/Preferences.xml
 - Ajouter la plage d'adresse des IPs du VPN avant la fin de la configuration ;
 - allowedNetworks="192.168.1.0/24,192.168.27.0/24"
 - Enregistrer les modifications ;
- Redémarrer la VM et patienter 20 secondes :
 - sudo shutdown -r now
- Se reconnecter en SSH ;
- Vérifier si Plex tourne :
 - sudo systemctl status plexmediaserver
 - Vous devriez voir un truc comme "Active: **active (running)**"

Mise en place du reverse proxy avec Nginx (pouvoir utiliser Plex via notre domaine freeboxos) :

- Commencer par stopper Plex :
 - sudo systemctl stop plexmediaserver
- Installer les dépendances suivantes :
 - sudo apt-get install software-properties-common
 - sudo apt install nginx

- Stopper le service NGINX car nous allons modifier sa configuration :
 - sudo systemctl stop nginx.service
- Prendre connaissance du nombre de coeurs associés à notre VM :
 - grep processor /proc/cpuinfo | wc -l
 - Garder en tête ce chiffre représentant le nombre de coeurs de notre VM ;
- Modifier le contenu du fichier de configuration NGINX :
 - sudo vi /etc/nginx/nginx.conf
 - Remplacer la valeur associée à la clé "worker_processes" par le nombre de coeurs de la VM récupéré dans l'étape précédente (2 pour moi) ;
 - Décommenter la ligne "#server_tokens off;" afin qu'elle soit active ;
- Créer le virtual host plex :
 - sudo vi /etc/nginx/sites-available/plex
 - Incrire le contenu suivant **en modifiant seulement la valeur en gras** par la vôtre :

```
upstream plex_backend {
    server      localhost:32400;
    keepalive   32;
}

map $http_upgrade $connection_upgrade {
    default      upgrade;
    "           close;
}

server {
    listen       80;
    server_name  aureltrouts.freeboxos.fr;

    # Some players don't reopen a socket and playback stops totally instead of resuming after an extended pause (e.g. Chrome)
    send_timeout 100m;

    # Faster resolving, improves stapling time.
    # Timeout and nameservers may need to be adjusted for your location
    Cloudflare's have been used here.
    resolver      1.1.1.1 1.0.0.1 valid=300s;
    resolver_timeout 10s;

    # Plex has A LOT of javascript, xml and html. This helps a lot, but if it causes playback issues with devices turn it off. (Haven't encountered any yet)
    gzip          on;
    gzip_vary     on;
    gzip_min_length 1000;
    gzip_proxied  any;
```

```

gzip_types      text/plain text/css text/xml application/xml text/javascript
application/x-javascript image/svg+xml;
gzip_disable    "MSIE [1-6].";

# Nginx default client_max_body_size is 1MB, which breaks Camera Upload
feature from the phones.
# Increasing the limit fixes the issue. Anyhow, if 4K videos are expected to be
uploaded, the size might need to be increased even more
client_max_body_size 20M;

# Forward real ip and host to Plex
proxy_set_header Host $host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;
proxy_set_header Host $server_addr;
proxy_set_header Referer $server_addr;
proxy_set_header Origin $server_addr;

# Plex headers
proxy_set_header X-Plex-Client-Identifier $http_x_plex_client_identifier;
proxy_set_header X-Plex-Device $http_x_plex_device;
proxy_set_header X-Plex-Device-Name $http_x_plex_device_name;
proxy_set_header X-Plex-Platform $http_x_plex_platform;
proxy_set_header X-Plex-Platform-Version $http_x_plex_platform_version;
proxy_set_header X-Plex-Product $http_x_plex_product;
proxy_set_header X-Plex-Token $http_x_plex_token;
proxy_set_header X-Plex-Version $http_x_plex_version;
proxy_set_header X-Plex-Nocache $http_x_plex_nocache;
proxy_set_header X-Plex-Provides $http_x_plex_provides;
proxy_set_header X-Plex-Device-Vendor $http_x_plex_device_vendor;
proxy_set_header X-Plex-Model $http_x_plex_model;

#Websockets
proxy_http_version 1.1;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection "upgrade";

#Buffering off send to the client as soon as the data is received from Plex.
proxy_redirect      off;
proxy_buffering     off;

location / {
    proxy_pass      http://plex_backend;
}
}

```

- Activer ce nouveau virtual host :
 - sudo ln -s /etc/nginx/sites-available/plex /etc/nginx/sites-enabled/plex
- Relancer le service NGINX :
 - sudo systemctl start nginx.service
- Relancer plex :
 - sudo systemctl start plexmediaserver

Mise en place du protocole HTTPS et HTTP/2 pour Plex :

- Commencer par stopper Plex et NGINX :
 - sudo systemctl stop nginx.service
 - sudo systemctl stop plexmediaserver
- Installer les dépendances suivantes :
 - sudo apt install certbot
 - sudo apt install python3-certbot-nginx
- Créer le répertoire destiné au fichier temporaire de Let's Encrypt :
 - sudo mkdir -p /var/www/letsencrypt
- Créer le fichier temporaire :
 - sudo vi /etc/nginx/snippets/letsencrypt
 - Renseigner le contenu suivant :

```
location ^~ /.well-known/acme-challenge/ {
    default_type "text/plain";
    root      /var/www/letsencrypt;
}
```

 - Inclure ce nouveau fichier à la configuration NGINX de notre virtual host :
 - sudo vi /etc/nginx/sites-available/plex
 - Ajouter la ligne suivante à la fin du bloc server :
 - include /etc/nginx/snippets/letsencrypt;
 - Ajouter une redirection dans le paramétrage de la Freebox en forwardant ce qui vient de 80 de l'extérieur vers 80 vers l'intérieur et sur notre VM ; Il nous faut ABSOLUMENT le port 80 pour permettre à Let's Encrypt de nous générer nos certificats ;
 - Démarrer NGINX (pas Plex) :
 - sudo systemctl start nginx.service
 - Générer les certificats :
 - sudo certbot certonly --webroot -w /var/www/letsencrypt --agree-tos --hsts --staple-ocsp --no-eff-email --email **mottier.aurelien.nas@gmail.com** -d **aureltrouts.freeboxos.fr** --rsa-key-size 4096
 - Si Let's Encrypt a pu valider le domaine (et donc généré les certificats), vous aurez un message comme celui-ci :
 - **IMPORTANT NOTES:**
 - *Congratulations! Your certificate and chain have been saved at ...*
 - Identifier le répertoire contenant nos certificats :
 - **/etc/letsencrypt/live/aureltrouts.freeboxos.fr/**

- Identifier les certificats contenus :
 - ***cert.pem*** : le certificat du domaine **aureltrouts.freeboxos.fr**
 - ***chain.pem*** : le certificat *Let's Encrypt*
 - ***fullchain.pem*** : les certificats *cert.pem* et *chain.pem* combinés
 - ***privkey.pem*** : la clé privée du certificat.
- Générer une nouvelle clé Diffie-Hellman (DH) afin d'améliorer encore plus la sécurité du protocole SSL/TLS :
 - `sudo openssl dhparam -out /etc/ssl/certs/dhparam.pem 4096`
 - Patienter car cette opération peu prendre 20 minutes...
- Réajuster les permissions de cette nouvelle clé DH :
 - `sudo chmod 600 /etc/ssl/certs/dhparam.pem`
- Renouveler automatiquement les certificats fournis par Let's Encrypt (qui expirent tous les 90 jours) :
 - `sudo crontab -e`
 - `30 5 * * 1 /usr/bin/certbot renew >> /var/log/le-renew.log`
(tous les lundis à 05h30 du matin)
- Modifier la règle de forwarding de la freebox pour forward le port 443 et non le port 80 comme jusqu'à présent ;
- Mettre à jour la configuration du virtual host, en apportant les changements suivants (j'ai gardé **443** pour permettre le renouvellement de certificats...) :
 - `listen 443 ssl http2;`
 - Ajouter la ligne suivante juste en dessous de notre listen (cette nouvelle ligne ressemble beaucoup mais nous devons l'ajouter) :
 - `listen [::]:443 ssl http2;`
 - Puis ajouter le gros bloc :

```
# SSL certificates to enable HTTPS :
ssl_certificate      /etc/letsencrypt/live/aureltrouts.freeboxos.fr/
fullchain.pem;
ssl_certificate_key  /etc/letsencrypt/live/aureltrouts.freeboxos.fr/
privkey.pem;
ssl_trusted_certificate /etc/letsencrypt/live/aureltrouts.freeboxos.fr/
chain.pem;
ssl_dhparam           /etc/ssl/certs/dhparam.pem;
ssl_protocols         TLSv1.3;
ssl_prefer_server_ciphers off;
ssl_stapling          on;
ssl_stapling_verify   on;
ssl_session_timeout   1d;
ssl_session_cache     shared:MozSSL:10m;
ssl_session_tickets   off;

# HSTS (nginx_http_headers_module is required) (63072000 seconds)
add_header Strict-Transport-Security "max-age=63072000" always;
```

- Enfin, supprimer l'ancienne version du protocole HTTP, en retirant la

ligne suivante :

- proxy_http_version 1.1;
- Enregistrer les modifications et quitter l'édition du fichier ;
- Tester la validité de notre configuration NGINX :
 - sudo nginx -t
- Si le test est concluant, démarrer NGINX :
 - sudo systemctl start nginx.service
- Démarrer Plex :
 - sudo systemctl start plexmediaserver
- Tester le support TLS 1.3 soit supporté :
 - sudo curl -I -v --tlsv1.3 --tls-max 1.3 <https://aureltrouts.freeboxos.fr>
(...left intact)
- Vérifier que le TLS 1.2, bien que toujours d'actualité, ne soit pas supporté :
 - sudo curl -I -v --tlsv1.2 --tls-max 1.2 <https://aureltrouts.freeboxos.fr>
(...alert protocol version)
- Vérifier que les TLS 1.0 et 1.1 ne sont pas supportés (obsolètes) :
 - sudo curl -I -v --tlsv1 --tls-max 1.0 <https://aureltrouts.freeboxos.fr>
(...no protocols available)
 - sudo curl -I -v --tlsv1.1 --tls-max 1.1 <https://aureltrouts.freeboxos.fr>
(...no protocols available)

Ajout du service de récupération des sous-titres :

- Identifier la dernière version stable en allant ici : <https://github.com/pannal/Sub-Zero.bundle/releases/latest>
 - Dans mon cas : 2.6.5.3223
- Se rendre dans le dossier Plex :
 - cd /var/lib/plexmediaserver/Library/Application\ Support/Plex\ Media\ Server/Plug-ins/
- Valoriser les valeurs en gras par la bonne version et lancer la commande ainsi constituée :
 - wget <https://github.com/pannal/Sub-Zero.bundle/releases/download/2.6.5.3223/Sub-Zero.bundle-2.6.5.3223.zip>
- Dézipper le contenu :
 - unzip Sub-Zero.bundle-**2.6.5.3223**.zip
- Supprimer le ZIP :
 - rm Sub-Zero.bundle-**2.6.5.3223**.zip

Configuration de Plex :

- Se rendre sur la page suivante : <https://aureltrouts.freeboxos.fr/>
 - Vérifier la connexion HTTPS en cliquant sur le cadenas à gauche de l'URL dans le navigateur ;
 - Ignorer le Plex Pass (fermer la fenêtre) ;
 - Trouver un nom pour Plex : vm-plex (exemple)
 - Décocher la "Allow me to access my media outside my home", car nous allons d'abord nous concentrer sur le réseau interne ;
 - Ne pas ajouter de library pour le moment ;

- Procéder aux paramétrages personnels en passant en revue TOUTES les rubriques à gauche (et bien penser à désactiver le transcodage) ;
- Procéder à l'ajout d'une première PETITE library ;
- Attendre la fin de l'indexation ;
- **TODO** tester avec un client et checker que nous avons bien du direct play (dashboard Plex) quelque soit le flux lu par le client (SH, HD ou UHD) ;

Mise à jour de Plex :

- Se connecter en SSH sur la VM ;
- Stopper le service Plex :
 - sudo systemctl stop plexmediaserver
- Faire une sauvegarde de notre configuration quelque part sur la Freebox :
 - cd (rien)
 - sudo cp /var/lib/plexmediaserver/Library/Application\ Support/Plex\ Media\ Server/Preferences.xml ./
- Eteindre la VM proprement ;
- Faire une sauvegarde de la VM complète ;
- Relancer la VM et se connecter en SSH ;
- Stopper le service Plex et NGINX :
 - sudo systemctl stop plexmediaserver
 - sudo systemctl stop nginx.service
- Lancer la mise à jour de Plex (la mise à jour de Plex sera greffée par dessus l'ancienne) :
 - sudo apt-get update && sudo apt-get upgrade
- Vérifier que nous n'avons pas perdu notre ancienne configuration :
 - sudo vi /var/lib/plexmediaserver/Library/Application\ Support/Plex\ Media\ Server/Preferences.xml
 - Nous devrions retrouver le même fichier qu'avant, avec les mêmes informations ;
- Relancer NGINX et Plex :
 - sudo systemctl start nginx.service
 - sudo systemctl start plexmediaserver
- Vérifier si Plex tourne :
 - sudo systemctl status plexmediaserver
 - Vous devriez voir un truc comme "Active: **active (running)**"
- Se connecter à <https://aureltrouts.freeboxos.fr> et valider que :
 - Nous sommes toujours en HTTPS ;
 - Nous pouvons toujours voir les libraires ;
 - Nous avons d'ailleurs plus le message de demande de mise à jour sur Plex ;