Hidden Service Tor serveur Apache.

I Installation de Tor.

```
sudo apt update
sudo apt upgrade
sudo apt install tor
```

II Configuration des Hidden Service.

Éditons le fichiers de configuration.

```
sudo nano /etc/tor/torrc
```

Il faut y dé-commenter les lignes

```
HiddenServiceDir /var/lib/tor/hidden_service/
HiddenServicePort 80 127.0.0.1:80
```

Il faut maintenant redémarrer tous les services Tor :

```
sudo systemctl reload tor
```

ce qui créera le répertoire `/var/lib/tor/hidden_service` et le peuplera des trois fichiers `hostname`, `private_key` et `public_key`. Pour vérifier faites :

```
sudo ls /var/lib/tor/hidden_service
```

Le fichier `hostname` contient le nom de domaine de votre site sur le réseau Tor une adresse de la forme `nomdedomaine.onion`. Pour voir votre nom de domaine faites :

```
sudo nano /var/lib/tor/hidden_service/hostname
```

Les deux autres fichiers,`private_key` et `public_key`, sont des identifiants uniques de votre Hidden Service.

Sauvegarder vos identifiants de Hidden Service Tor sur clef USB.

Les fichiers dans `/var/lib/tor/other_hidden_service` c'est-à-dire `hostname`, `private_key` et `public_key` seront indispensables. Si vous avez l'intention de garder votre service un peu longtemps vous avez intérêt à les sauvegarder.

Branchez une clef USB sur le serveur. Puis sauvegardez les fichiers sur la clef USB (appelée ici `maclefusb`) à la racine de votre clé USB :

```
sudo cp /var/lib/tor/hidden_service/hostname /media/pi/maclefusb/
    hostname
sudo cp /var/lib/tor/hidden_service/hs_ed25519_public_key /media/pi/
    maclefusb/hs_ed25519_public_key
sudo cp /var/lib/tor/hidden_service/hs_ed25519_secret_key /media/pi/
    maclefusb/hs_ed25519_secret_key
```

Les fichiers dans */var/lib/tor/hidden_ service* sont dans le groupe et l'utilisateur *debian-tor*.

Remplacer les identifiants de Hidden Service Tor.

Vous pouvez également remplacez les fichiers par d'autres que vous auriez précédemment sauvegardés comme expliqué dans le cas d'un nom de domaine onion personnalisé.

III Installation de Apache.

```
13 sudo apt install apache2
```

IV Répertoire accueillant le site.

Créer un répertoire pour accueillir tous les documents du site

```
14 mkdir /home/pi/nomdedomaine
```

Explications.

- Pour le `nomdedomaine` le choisi celui qui apparaît dans le fichier `hostname` sous la forme `nomdedomaine.onion`, par souci de cohérence mais ce n'est pas obligatoire. Ce n'est qu'un dossier.

Apache2 cherche les fichiers du site dans `/var/www` il faut donc faire un lien vers le répertoire que nous avons créé :

```
15 sudo ln -s /home/pi/nomdedomaine /var/www
```

V Créer un virtualhost pour le site.

Supprimer le virtualhost par défaut.

Enlever le virtualhost par défaut de Debian :

```
16 sudo a2dissite 000-default.conf
```

Créer un nouveau virtualhost.

Ouvrez le fichier du virtualhost par défaut :

```
17 sudo nano /etc/apache2/sites-available/000-default.conf
```

Et modifiez-le comme suit pour ajouter le nom de domaine de votre site qui est dans le fichier `hostname` et qui, ci-dessous, est appelé `nomdedomaine` :

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
ServerName nomdedomaine.onion

ServerAdmin webmaster@localhost
DocumentRoot /var/www/nomdedomaine

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Il faudrait modifier ce passge pour ajouter le HTTPS. La prochaine fois.

Il n'est évidemment pas question de laisser une adresse mail à moins qu'elle ne soit anonymisée.

Il faut ensuite enregistrer ce fichier modifié sous le nom `nomdedomaine.onion.conf` dans `/etc/apache2/sites-available/`.

VI Activer le virtualhost.

Puis on active ce virtualhost :

```
sudo a2ensite nomdedomaine.onion.conf
```

Redémarrons apache2 pour que tout soit activé :

```
sudo systemctl reload apache2
```

VII Imposer à apache2 l'utilisation de tor.

Ouvrir le fichier de configuration des ports sur apache2

```
sudo nano /etc/apache2/ports.conf
```

et le modifier comme suit :

```
# If you just change the port or add more ports here, you will likely
    also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 127.0.0.1:80

<IfModule ssl_module>
Listen 443
</IfModule>

<IfModule mod_gnutls.c>
Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Il ne reste plus qu'à relancer apache2

```
sudo systemctl reload apache2
```

VIII Réglages du serveur Apache2.

Il est intéressant d'empêcher la navigation dans l'arborescence du site.

L'idée est d'interdire l'accès à des répertoires avec les fichiers `.htaccess` mais ceux-ci sont par défaut inactifs. Pour rendre ces fichiers opérants il faut ouvrir le fichier apache2.conf

```
sudo nano /etc/apache2/apache2.conf
```

Puis aller jusqu'à la section

```
<Directory /var/www/>
Options Indexes FollowSymLinks
AllowOverride None
Require all granted
</Directory>
```

et remplacer le `None` en `All` (faire fonctionner htaccess dans le répertoire `/var/www/`) :

```
<Directory /var/www/>
Options Indexes FollowSymLinks
AllowOverride All
Require all granted
</Directory>
```

Pour empêcher la possibilité de naviguer dans les répertoires du site dans `home/pi/nomdededomaine` créez un fichier `.htaccess`

```
79 sudo nano /home/pi/nomdedomaine/.htaccess
```

écrire dedans

```
80 Options -Indexes
```

il est possible d'installer des `.htaccess` afin d'empêcher l'accès à un dossier particulier.

Il ne reste plus qu'à mettre un fichier `index.html` dans le dossier `/home/pi/nomdedomai` et votre site web sur Tor est en ligne (accessible uniquement via le réseau Tor évidemment) après un dernier petit reboot.