Passage en https

CAS 1 : Serveur NGINX

Modification du fichier *pod/custom/settings_local.py*

ajouter les lignes suivantes :

```
SECURE_SSL_REDIRECT = True
SESSION_COOKIE_SECURE = True
CSRF_COOKIE_SECURE = True
```

Créez une clé privée et le certificat à l'aide de la commande OpenSSL. Il s'agit ici d'un certificat auto-signé. Personnellement, il ne me sert qu'à faire tourner mon serveur web en local pour du développement.

```
sudo mkdir /etc/nginx/certificate
cd /etc/nginx/certificate
sudo openssl req -new -newkey rsa:4096 -x509 -sha256 -days 365 -nodes -out nginx-certificate.crt -keyout nginx.
key
```

Renseignez les informations.

Modifiez ensuite ce fichier de configuration :

```
sudo vi /etc/nginx/sites-available/default
```

Comme ceci :

```
#Redirige les utilisateurs http vers https
server {
        listen 80 default_server;
        listen [::]:80 default_server;
        server_name _;
        return 301 https://$host$request_uri;
}

# https
server {
        listen 443 ssl default_server;
        listen [::]:443 ssl default_server;
        ssl_certificate /etc/nginx/certificate/nginx-certificate.crt;
        ssl_certificate_key /etc/nginx/certificate/nginx.key;
        root /var/www/html;
        index index.html index.htm index.nginx-debian.html;
        server_name _;
        location / {
                try_files $uri $uri/ =404;
        }
}
```

Dans *pod/custom/pod_nginx.conf*, ajoutez cette ligne pour ouvrir le port 443 (port https) :

```
listen 443;
```

Puis relancez les services.

```
sudo systemctl restart uwsgi-pod nginx
```

CAS 2 : serveur Apache

Mise en place d'un serveur front de type Apache: configuration des fichiers pod.conf et pod-ssl.conf

1. configuration du fichier pod.conf

```
<VirtualHost *:80>
  ServerAdmin pod@univ.fr
  ServerName pod.univ.fr

  DocumentRoot /var/www/html

  RewriteEngine On
  RewriteCond %{HTTPS} !on
  RewriteRule (.*) https://%{HTTP_HOST}/

  ErrorLog ${APACHE_LOG_DIR}/pod_error.log
  CustomLog ${APACHE_LOG_DIR}/pod_access.log combined
</VirtualHost>
```

2. configuration du fichier pod-ssl.conf

```
<VirtualHost *:443>
  ServerAdmin pod@univ.fr
  ServerName pod.univ.fr

  SSLEngine on
  SSLProtocol all -SSLv2 -SSLv3
  SSLCipherSuite ALL:!ADH:!EXPORT:!SSLv2:RC4+RSA:+HIGH:+MEDIUM:+LOW
  SSLProxyCheckPeerCN Off
  SSLProxyCheckPeerName Off
  SSLProxyVerify none

  SSLCertificateFile /etc/apache2/ssl/pod_univ.fr.crt
  SSLCertificateKeyFile /etc/apache2/ssl/pod_univ.fr.key
  SSLCertificateChainFile /etc/apache2/ssl/DigiCertCA.crt

  DocumentRoot /var/www/html
  Header always set X-Frame-Options "sameorigin"
  RewriteEngine On
  ProxyPassReverse / http://ip_priv_pod_apache/
  ProxyPass / http://ip_priv_pod_apache/
  ProxyPreserveHost On

  <Proxy *>
    Order deny,allow
    Allow from all
  </Proxy>
  <Directory />
    Options FollowSymLinks
    AllowOverride None
  </Directory>

  CustomLog /var/log/apache2/pod_access.log combined
  ErrorLog /var/log/apache2/pod_error.log
  ServerSignature Off
</VirtualHost>
```

relancer apache

sur le serveur front-pod-priv

ajouter les lignes suivantes dans *pod/custom/settings_local.py*

ajouter les lignes suivante :

```
SESSION_COOKIE_SECURE = True
CSRF_COOKIE_SECURE = True
USE_X_FORWARDED_HOST = True
SECURE_PROXY_SSL_HEADER = ('HTTP_X_FORWARDED_PROTO', 'https')
```

puis relancer le service

```
sudo systemctl restart uwsgi-pod
```

CAS 3 : Run server

(Adaptation de la doc suivante : https://timonweb.com/django/https-django-development-server-ssl-certificate/)

Il faut créer des certificats autosignés, **attention il faut se placer en dehors de podv3** :

```
(django_pod3) pod@pod:/usr/local/django_projects$ openssl req -newkey rsa:2048 -new -nodes -x509 -days 3650 -
keyout key.pem -out cert.pem
```

Ensuite, il faut installer des extensions:

```
(django_pod3) pod@pod:/usr/local/django_projects$ pip install django-extensions Werkzeug
```

Et modifier le fichier settings.py pour ajouter ""django_extensions"," dans la liste des applications installées.

```
(django_pod3) pod@pod:/usr/local/django_projects$ vim podv3/pod/settings.py
INSTALLED_APPS = [
    # [...]
    "django_extensions",
    # Pod Applications
        # [...]
]
```

Enfin, pour lancer le server, il suffit de se placer dans podv3 et de lancer la commande suivante :

```
(django_pod3) pod@pod:/usr/local/django_projects/podv3$ python3 manage.py runserver_plus pod.univ.fr:8000 --
insecure --cert-file ../cert.pem --key-file ../key.pem
```