Tutoriel : Configuration d'un Reverse Proxy HTTPS avec Nginx sur Ubuntu Server

Introduction

Ce tutoriel explique comment configurer un serveur Nginx comme reverse proxy pour rediriger automatiquement le trafic HTTP vers HTTPS, en utilisant un certificat SSL pour sécuriser les communications.

Prérequis :

- Ubuntu Server (version 18.04 ou plus récente)
- Accès root ou utilisateur avec privilèges sudo
- Connaissance de base des commandes Linux

1. Installation des paquets nécessaires

Commencez par mettre à jour la liste des paquets et installer les logiciels requis :

```
sudo apt update
sudo apt install -y nginx openssl
```

2. Création du certificat SSL avec mkcert

Pour sécuriser votre site, vous avez besoin d'un certificat SSL. Pour ce tutoriel, nous utiliserons mkcert, qui crée des certificats de développement localement approuvés :

```
# Installation des dépendances
sudo apt install -y libnss3-tools

# Téléchargement et installation de mkcert
wget https://github.com/FiloSottile/mkcert/releases/download/v1.4.3/mkcert-v1.4.3-linux-amd64
chmod +x mkcert-v1.4.3-linux-amd64
sudo mv mkcert-v1.4.3-linux-amd64 /usr/local/bin/mkcert

# Installation de l'autorité de certification locale
mkcert -install

# Création du répertoire pour les certificats
sudo mkdir -p /etc/nginx/ssl

# Génération du certificat pour votre domaine
```

```
mkcert -key-file /etc/nginx/ssl/ipssibankso.key -cert-file /etc/nginx/ssl/ipssibankso.crt ipss

# Ajustement des permissions
sudo chmod 644 /etc/nginx/ssl/ipssibankso.crt
sudo chmod 600 /etc/nginx/ssl/ipssibankso.key
```

L'avantage de mkcert est qu'il installe automatiquement une autorité de certification locale sur votre machine. Cela signifie que les certificats générés seront considérés comme fiables par votre navigateur, sans aucun avertissement de sécurité.

3. Création du contenu web de démonstration

Créez un répertoire pour votre site web et une page HTML de démonstration :

```
# Création du répertoire pour le site
sudo mkdir -p /var/www/ipssibankso

# Création de la page HTML
sudo nano /var/www/ipssibankso/index.html
```

Ajoutez le contenu HTML suivant :

```
<!DOCTYPE html>
<html>
<head>
    <title>ipssibankso.com - Site Sécurisé</title>
    <style>
        body { font-family: Arial, sans-serif; margin: 40px; line-height: 1.6; }
        h1 { color: #0066cc; }
        .secure { color: green; font-weight: bold; }
        .box { border: 1px solid #ddd; padding: 20px; border-radius: 5px; background-color: #f
    </style>
</head>
<body>
    <h1>Bienvenue sur ipssibankso.com</h1>
    <div class="box">
        <h2>Site configuré avec <span class="secure">HTTPS sécurisé</span></h2>
        <p>Caractéristiques de cette configuration :</p>
        <ul>
            <li>Reverse proxy avec Nginx</li>
            <li>Redirection automatique HTTP vers HTTPS</li>
            <li>Certificat SSL (auto-signé pour cette démonstration)</li>
            <li>Configuration complète pour un site web sécurisé</li>
        </ul>
        <p>Ce site est maintenant accessible via HTTPS et affiche le cadenas de sécurité.</p>
    </div>
```

```
</body>
</html>
```

4. Configuration de Nginx comme reverse proxy

Créez un fichier de configuration pour votre site :

```
sudo nano /etc/nginx/sites-available/ipssibankso.conf
```

Ajoutez la configuration suivante :

```
# Redirection HTTP vers HTTPS
server {
    listen 80;
    server_name ipssibankso.com www.ipssibankso.com;
    return 301 https://$host$request_uri;
}

# Configuration HTTPS
server {
    listen 443 ssl;
    server_name ipssibankso.com www.ipssibankso.com;

    # Certificats SSL
    ssl_certificate /etc/nginx/ssl/ipssibankso.crt;
    ssl_certificate_key /etc/nginx/ssl/ipssibankso.key;

    # Paramètres SSL recommandés
    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_prefer_server_ciphers on;
    ssl_session_timeout 1d;
    ssl_session_cache shared:SSL:10m;

    # Racine du site web
    root /var/www/ipssibankso;
    index index.html;

    location / {
        try_files $uri $uri/ =404;

        # Pour un reverse proxy vers une application backend, décommentez ces lignes
        # proxy_pass http://localhost:8080;
        # 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;
    }
}
```

5. Activation de la configuration Nginx

Activez votre configuration et redémarrez Nginx :

```
# Créer un lien symbolique vers sites-enabled
sudo ln -s /etc/nginx/sites-available/ipssibankso.conf /etc/nginx/sites-enabled/

# Vérifier la configuration
sudo nginx -t

# Si tout est OK, redémarrer Nginx
sudo systemctl restart nginx
```

6. Configuration du fichier hosts local

Pour tester localement, ajoutez votre domaine dans le fichier hosts :

```
sudo nano /etc/hosts
```

Ajoutez cette ligne :

```
127.0.0.1 ipssibankso.com www.ipssibankso.com
```

7. Test et vérification

Accédez à votre site via un navigateur web :

1. Ouvrez un navigateur sur votre machine virtuelle
2. Accédez à http://ipssibankso.com
3. Vous devriez être automatiquement redirigé vers https://ipssibankso.com
4. Si vous avez correctement installé mkcert, vous ne devriez pas voir d'avertissement de sécurité
5. Vérifiez la présence du cadenas dans la barre d'adresse

8. Exposition externe avec ngrok (optionnel)

Pour rendre votre site accessible depuis Internet :

```
# Installation de ngrok (si pas déjà fait)
wget https://bin.equinox.io/c/bNyj1mQVY4c/ngrok-v3-stable-linux-amd64.tgz
tar xvzf ngrok-v3-stable-linux-amd64.tgz
sudo mv ngrok /usr/local/bin/
```

```
# Configuration de ngrok (nécessite une inscription sur ngrok.com)
ngrok config add-authtoken VOTRE_TOKEN

# Lancement du tunnel
ngrok http 80
```

Mise à jour de la configuration Nginx pour ngrok :

```
sudo nano /etc/nginx/sites-available/ipssibankso.conf
```

Ajoutez ces blocs à la fin du fichier :

```
# Pour l'URL ngrok
server {
    listen 80;
    server_name ~^.*\.ngrok-free\.app$;
    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl;
    server_name ~^.*\.ngrok-free\.app$;

    ssl_certificate /etc/nginx/ssl/ipssibankso.crt;
    ssl_certificate_key /etc/nginx/ssl/ipssibankso.key;

    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_prefer_server_ciphers on;

    root /var/www/ipssibankso;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

Redémarrez Nginx :

```
sudo nginx -t
sudo systemctl restart nginx
```

Explication technique

Qu'est-ce qu'un reverse proxy ?

Un reverse proxy est un serveur qui se place entre les clients et les serveurs d'applications. Il intercepte les requêtes des clients, les transmet aux serveurs appropriés, puis renvoie les réponses aux clients.

Avantages d'un reverse proxy :

- **Sécurité** : Masque les serveurs internes, ajoute une couche de protection
- **Équilibrage de charge** : Distribue les requêtes entre plusieurs serveurs
- **Mise en cache** : Améliore les performances en stockant les réponses fréquemment demandées
- **SSL Termination** : Gère le chiffrement/déchiffrement, soulageant les serveurs d'applications

Redirection HTTP vers HTTPS

La configuration mise en place force toutes les connexions HTTP (non sécurisées) à être redirigées vers HTTPS (sécurisées). Cette redirection est réalisée avec la directive `return 301 https://$host$request_uri;` .

Certificats SSL

Pour un environnement de production, il est recommandé d'utiliser un certificat émis par une autorité de certification reconnue (comme Let's Encrypt).

Le certificat généré avec mkcert dans ce tutoriel est parfait pour des tests ou des environnements de développement. Contrairement aux certificats auto-signés traditionnels, ceux créés par mkcert sont automatiquement approuvés par votre navigateur local sans générer d'avertissement, car mkcert installe sa propre autorité de certification dans le magasin de certificats de votre système.

Conclusion

Vous avez maintenant configuré avec succès un reverse proxy Nginx qui :

- Redirige automatiquement tout le trafic HTTP vers HTTPS
- Utilise un certificat SSL pour sécuriser les communications
- Peut servir de proxy pour vos applications backend
- Est accessible localement via ipssibankso.com et potentiellement depuis Internet via ngrok

Cette configuration est idéale pour sécuriser vos applications web et les rendre accessibles de manière sécurisée.