Création de Certificats et d'Hôtes Virtuels HTTPS dans Apache

Apache HTTPD permet de configurer plusieurs **hôtes virtuels HTTPS**, chacun avec un certificat SSL/TLS dédié, pour sécuriser les échanges entre les clients et le serveur. Voici une procédure complète pour créer des certificats et configurer des hôtes virtuels HTTPS.

---

1. Création des Certificats SSL/TLS

Option 1 : Certificat Auto-signé (Pour des Tests Internes)

1. **Créer une Clé Privée**

```
openssl genrsa -out /etc/ssl/private/example.key 2048
```

2. **Créer une CSR (Certificate Signing Request)**

```
openssl req -new -key /etc/ssl/private/example.key -out
/etc/ssl/certs/example.csr
```

   Pendant la création, remplissez les informations, notamment :

   - **Common Name (CN)** : `example.com` (nom de domaine de l'hôte virtuel).

3. **Signer le Certificat**

```
openssl x509 -req -days 365 -in /etc/ssl/certs/example.csr -signkey
/etc/ssl/private/example.key -out /etc/ssl/certs/example.crt
```

4. **Résultat**

   - **Clé privée** : `/etc/ssl/private/example.key`
   - **Certificat auto-signé** : `/etc/ssl/certs/example.crt`

Option 2 : Certificat Let's Encrypt (Pour un Site Public)

1. **Installer Certbot**

```
sudo apt update
sudo apt install certbot python3-certbot-apache
```

2. **Obtenir un Certificat SSL**

```
sudo certbot --apache -d example.com -d www.example.com
```

   - Certbot configure automatiquement Apache pour utiliser les certificats générés.

3. **Renouvellement Automatique**
   Let's Encrypt valide les certificats pour 90 jours. Configurez un renouvellement automatique :

```
sudo certbot renew --quiet
```

---

2. Création d'Hôtes Virtuels HTTPS

Chaque hôte virtuel peut avoir son propre certificat et configuration.

Configuration de Base pour un Hôte Virtuel HTTPS

1. **Créer un Fichier de Configuration**
   Fichier : `/etc/apache2/sites-available/example-https.conf`

```
<VirtualHost *:443>
    ServerName example.com
    ServerAlias www.example.com

    DocumentRoot /var/www/html/example

    SSLEngine On
    SSLCertificateFile /etc/ssl/certs/example.crt
    SSLCertificateKeyFile /etc/ssl/private/example.key

    <Directory /var/www/html/example>
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/example_error.log
    CustomLog ${APACHE_LOG_DIR}/example_access.log combined
</VirtualHost>
```

2. **Activer le Site et le Module SSL**

```
sudo a2enmod ssl
sudo a2ensite example-https
```

```
sudo systemctl reload apache2
```

3. **Créer le Répertoire DocumentRoot**

```
sudo mkdir -p /var/www/html/example
echo "Welcome to Example HTTPS Site" | sudo tee
/var/www/html/example/index.html
```

---

3. Configurer plusieurs Hôtes Virtuels HTTPS

Si vous avez plusieurs noms de domaine (`example1.com`, `example2.com`), configurez-les ainsi :

1. **Certificat pour Chaque Domaine**
   - Répétez la procédure de création de certificat pour chaque domaine (auto-signé ou Let's Encrypt).
2. **Fichier de Configuration pour Chaque Domaine**

Fichier : `/etc/apache2/sites-available/example1-https.conf`

```
<VirtualHost *:443>
    ServerName example1.com
    DocumentRoot /var/www/html/example1

    SSLEngine On
    SSLCertificateFile /etc/ssl/certs/example1.crt
    SSLCertificateKeyFile /etc/ssl/private/example1.key

    <Directory /var/www/html/example1>
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

Fichier : `/etc/apache2/sites-available/example2-https.conf`

```
<VirtualHost *:443>
    ServerName example2.com
    DocumentRoot /var/www/html/example2

    SSLEngine On
    SSLCertificateFile /etc/ssl/certs/example2.crt
    SSLCertificateKeyFile /etc/ssl/private/example2.key
```

```
    <Directory /var/www/html/example2>
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

3. **Activer les Sites**

```
sudo a2ensite example1-https
sudo a2ensite example2-https
sudo systemctl reload apache2
```

---

4. Rediriger HTTP vers HTTPS

1. **Créer un VirtualHost pour HTTP**
   Ajoutez un fichier `/etc/apache2/sites-available/redirect-http.conf` :

```
<VirtualHost *:80>
    ServerName example.com
    Redirect permanent / https://example.com/
</VirtualHost>
```

2. **Activer la Redirection**

```
sudo a2ensite redirect-http
sudo systemctl reload apache2
```

---

5. Vérification et Tests

1. **Test des Hôtes Virtuels**
   - Ouvrez un navigateur et accédez à `https://example.com`.
   - Utilisez `curl` pour vérifier les en-têtes HTTPS :

```
curl -I https://example.com
```

2. **Valider le Certificat**
   - Utilisez un outil comme SSL Labs pour tester la validité et la sécurité de votre certificat.

---

6. Renforcer la Sécurité du HTTPS

1. **Désactiver les Protocoles Obsolètes**
   Ajoutez dans chaque hôte virtuel HTTPS :

   ```
   SSLProtocol All -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
   ```

2. **Configurer les Chiffrements**
   Ajoutez :

   ```
   SSLCipherSuite HIGH:!aNULL:!MD5:!3DES
   SSLHonorCipherOrder On
   ```

3. **Activer HSTS (HTTP Strict Transport Security)**
   Ajoutez :

   ```
   Header always set Strict-Transport-Security "max-age=31536000;
   includeSubDomains; preload"
   ```

4. **Vérifiez avec un Testeur SSL**
   Utilisez SSL Labs pour détecter les failles potentielles.

---

7. Bonnes Pratiques

1. **Utiliser Let's Encrypt pour des Certificats Gratuits**
   - Automatiser leur renouvellement avec `certbot`.
2. **Activer les Logs**
   - Vérifiez les logs pour déboguer les problèmes :

     ```
     sudo tail -f /var/log/apache2/error.log
     ```
3. **Surveiller les Expirations**
   - Configurez des alertes pour être prévenu avant l'expiration des certificats.

Exemple Complet : Deux Hôtes HTTPS avec Redirection HTTP

Fichier : `/etc/apache2/sites-available/redirect-http.conf`

```
<VirtualHost *:80>
    ServerName example1.com
    Redirect permanent / https://example1.com/
</VirtualHost>

<VirtualHost *:80>
    ServerName example2.com
    Redirect permanent / https://example2.com/
</VirtualHost>
```

Fichier : `/etc/apache2/sites-available/example1-https.conf`

```
<VirtualHost *:443>
    ServerName example1.com
    DocumentRoot /var/www/html/example1

    SSLEngine On
    SSLCertificateFile /etc/ssl/certs/example1.crt
    SSLCertificateKeyFile /etc/ssl/private/example1.key

    <Directory /var/www/html/example1>
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

Fichier : `/etc/apache2/sites-available/example2-https.conf`

```
<VirtualHost *:443>
    ServerName example2.com
    DocumentRoot /var/www/html/example2

    SSLEngine On
    SSLCertificateFile /etc/ssl/certs/example2.crt
    SSLCertificateKeyFile /etc/ssl/private/example2.key

    <Directory /var/www/html/example2>
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

Activez tous les sites :

```
sudo a2ensite redirect-http
sudo a2ensite example1-https
sudo a2ensite example2-https
sudo systemctl reload apache2
```