[Debian] OpenSSL

Introduction

Openssl est une bibliothèque standard de chiffrement disponible sur les systèmes Unix.

Chiffrement de fichier

Vous pouvez chiffrer un fichier en AES-256-CBC avec un mot de passe grâce à la commande suivante :

```
openssl enc -aes-256-cbc -salt -pbkdf2 -in <INPUT_FILE> -out <OUTPUT_FILE> -base64 -k <PASSPHRASE>
```

Et pour déchiffrer :

```
openssl enc -d -aes-256-cbc -pbkdf2 -in <INPUT_FILE> -out <OUTPUT_FILE> -base64 -k <PASSPHRASE>
```

Pour créer une paire de clés publique/privée RSA puis chiffrer un fichier avec, on peut utiliser openssl.

Tout d'abord, générez la paire de clés :

- Clé privée :

```
openssl genpkey -algorithm RSA -out priv.pem -aes-256-cbc -pass pass:mypass
```

- Clé publique :

```
openssl rsa -pubout -in key.pem -out pub.pem -passin pass:mypass
```

Puis chiffrez un fichier avec la clé publique :

```
openssl pkeyutl -encrypt -inkey pub.pem -pubin -in plain.txt -out cypher.enc && openssl base64 -in cypher.enc -out cypher.enc.base64
```

Puis pour déchiffrer avec la clé privée :

```
openssl base64 -d -in cypher.enc.base64 -out cypher.enc && openssl pkeyutl -decrypt -inkey priv.pem -in cypher.enc -out plain_decrypted.txt
```

Échange de clé

On peut utiliser **Diffie Hellman** pour échanger des clés.

Pour cela, commencez par créer un paramètre DH :

```
openssl genpkey -genparam -algorithm DH -out dhp.prem
```

Puis on peut générer une première clé :

```
openssl genpkey -paramfile dhp.pem -out dhkey1.pem
```

On peut afficher les informations de cette clé :

```
openssl pkey -in dhkey1.pem -text -noout
```

Et on peut créer une deuxième clé :

```
openssl pkey -in dhkey1.pem -pubout -out dhpub1.pem
```

On peut afficher les informations de cette clé :

```
openssl pkey -pubin -in dhpub1.pem -text
```

On peut ensuite créer une dérivation de ces clés :

```
openssl pkeyutil -derive -inkey dhkey1.pem -peerkey dhpub1.pem -out secret1.bin
```

```
openssl pkeyutil -derive -inkey dhkey2.pem -peerkey dhpub2.pem -out secret2.bin
```

On peut ensuite comparer les deux secrets et s'apercevoir qu'ils sont identiques :

```
cmp secret1.bin secret2.bin
```

Génération de clés ECC

Les clés ECC suivent les courbes elliptiques ce qui permet une taille de clé réduite en conservant un bon niveau de sécurité.

Création d'une clé ECC

```
openssl ecparam -genkey -name prime256v1 -out ecc.key
```

Demande de certificat

```
openssl req -new -sha256 -key ecc.key -nodes -out ecc.csr
```

Signer le certificat

Ici nous allons auto-signer le certificat, si vous utilisez une autorité de certification (CA), utilisez la clé privée du CA :

```
openssl req -x509 -sha256 -days 365 -key ecc.key -in ecc.csr -out ecc.crt
```

Extraire la clé publique

```
openssl ec -in net-sec.key -pubout
```

Revision #12
Created 1 October 2024 07:52:20 by Elieroc
Updated 1 October 2024 15:00:51 by Elieroc