Sécurité Informatique
Notions et Pratiques avec OpenSSL
TP 1

1 Importance de la sécurité

La sécurité informatique est essentielle pour :

- **Protéger la confidentialité** des données (empêcher l'accès non autorisé).
- **Garantir l'intégrité** (assurer que les données n'ont pas été altérées).
- **Assurer l'authenticité** (vérifier l'identité des utilisateurs et serveurs).
- **Renforcer la confiance** (clients, partenaires, transactions en ligne).

2 Le protocole HTTPS

a) HTTPS = **HTTP + TLS/SSL**.

b) Permet :

- Chiffrement des échanges → confidentialité
- Vérification du certificat → authenticité
- Contrôle de l'intégrité des messages

Exemple : lorsqu'un utilisateur accède à une banque en ligne, HTTPS empêche l'espionnage ou la modification des données échangées.

3 Les trois piliers :

a) **Confidentialité** : seules les personnes autorisées peuvent lire l'information.

b) **Intégrité** : les données ne sont pas modifiées pendant la transmission.

c) **Authenticité** : garantir que l'émetteur ou le serveur est bien celui qu'il prétend être.

4 OpenSSL Outils pratiques

a) Chiffrement symétrique (rapide, même clé pour chiffrer/déchiffrer)

Chiffrement avec AES-256

openssl enc -aes-256-cbc -salt -in clair.txt -out chiffre_sym.txt -k motdepasse

Déchiffrement

openssl enc -d -aes-256-cbc -in chiffre_sym.txt -out dechiffre.txt -k motdepasse

b) Chiffrement asymétrique (clé publique/privée)

 # Génération d'une paire de clés RSA

openssl genpkey -algorithm RSA -out cle_privee.pem -pkeyopt rsa_keygen_bits:2048

openssl rsa -in cle_privee.pem -pubout -out cle_publique.pem

 # Chiffrement avec clé publique

openssl rsautl -encrypt -pubin -inkey cle_publique.pem -in clair.txt -out chiffre_asym.txt

 # Déchiffrement avec clé privée

openssl rsautl -decrypt -inkey cle_privee.pem -in chiffre_asym.txt -out dechiffre.txt

c) Fonction de hachage (empreinte unique)

openssl dgst -sha256 fichier.txt

d) Signature numérique

Création d'une signature avec clé privée

openssl dgst -sha256 -sign cle_privee.pem -out signature.bin fichier.txt

Vérification avec clé publique

openssl dgst -sha256 -verify cle_publique.pem -signature signature.bin fichier.txt

e) Certificat numérique (auto-signé pour test)

Générer une clé privée

openssl genrsa -out serveur.key 2048

Générer une demande de certificat (CSR)

openssl req -new -key serveur.key -out serveur.csr

Générer un certificat auto-signé valide 365 jours

openssl x509 -req -days 365 -in serveur.csr -signkey serveur.key -out serveur.crt