Cryptographie Symétrique

- **Principe** : Utilise une clé partagée KKK pour chiffrer (CCC) et déchiffrer (MMM) les messages.
- **Caractéristiques** :
    - Chiffrement rapide, particulièrement en matériel.
    - Clés courtes : 128 à 256 bits.
    - Principal inconvénient : le besoin de partager une clé secrète.
- **Types de chiffrement** :
    - **Chiffrement à flot** : Génère une suite chiffrante pseudo-aléatoire combinée aux données via un XOR. Exemples : WEP, WPA, Bluetooth.
    - **Chiffrement par bloc** : Divise les données en blocs de taille fixe (64 ou 128 bits), chiffrés séparément. Exemples : AES, DES.
- **Sécurité** : Le chiffrement par flot peut être vulnérable si la même clé est réutilisée sans variation (vecteur d'initialisation - IV).

Cryptographie Asymétrique et OpenSSL

- **RSA** :
    - Utilise une paire de clés : une clé publique pour chiffrer et une clé privée pour déchiffrer.
    - Inconvénient : coûteux en calculs, nécessite de grandes clés.
- **OpenSSL** :
    - Bibliothèque open-source pour implémenter SSL/TLS.
    - Supporte de nombreux algorithmes de chiffrement, génération de clés, signatures numériques, et certificats X.509.

Hachage, Salage et Sécurité des Mots de Passe

- **Fonctions de hachage** :
    - Prend une entrée de taille variable et produit une empreinte de taille fixe.
    - Exemples : MD5, SHA-1, SHA-256.
- **Salage** :
    - Ajoute un sel aléatoire à un mot de passe avant de le hacher pour prévenir les attaques par tables arc-en-ciel.
    - Stocke le sel et l'empreinte hachée dans la base de données.

Codes Correcteurs d'Erreurs

- **Principe** : Détecter et corriger les erreurs dans les données transmises.
- **Exemples** :
    - **Code de parité** : Ajoute un bit indiquant si le nombre de bits 1 est pair ou impair.
    - **Code de répétition** : Répète chaque bit plusieurs fois.
    - **CRC (Contrôle de Redondance Cyclique)** : Ajoute un reste de division polynomiale à la séquence binaire.

    - **Code de Hamming** : Utilise des bits de parité pour détecter et corriger une erreur simple dans chaque bloc de données.

Signatures, Certificats et PKI (Public Key Infrastructure)

- **Signatures numériques** :
    - Garantissent l'authenticité et l'intégrité d'un message.
    - Utilisent une clé privée pour signer et une clé publique pour vérifier.
    - Exemple : Signature RSA.
- **Certificats numériques** :
    - Équivalents à des cartes d'identité numériques, contenant la clé publique et des informations sur le propriétaire.
    - Exemples de standards : X.509, OpenPGP.
- **Infrastructure à Clés Publiques (PKI)** :
    - Ensemble de composants pour créer, gérer, et révoquer des certificats à clés publiques.
    - Composants : Autorité d'enregistrement (RA), Autorité de certification (CA), Autorité de validation (VA), et archivage.