CRYPTOGRAPHIE

Chiffrement contemporain

AES, RSA, clés publiques — ce qui protège réellement vos communications.

2 h | Dès 14 ans | Fiche C2

01 Objectif

À la fin de cette séquence : Distinguer chiffrement symétrique et asymétrique, expliquer comment HTTPS protège une connexion, et comprendre ce que représente une clé de 256 bits.

02 L'essentiel

Symétrique : la même clé pour chiffrer et déchiffrer

AES-256 est le standard mondial. Une clé de 256 bits représente 2^{256} combinaisons — plus que le nombre d'atomes dans l'univers observable. Rapide, mais les deux parties doivent partager la clé sans se rencontrer : le problème fondamental de l'échange de clés.

Asymétrique : une clé publique, une clé privée

RSA et la cryptographie sur courbes elliptiques (ECC) résolvent l'échange de clés. Vous publiez votre clé publique — chacun peut l'utiliser pour vous chiffrer un message — mais seule votre clé privée peut le déchiffrer. La sécurité repose sur la difficulté mathématique de factoriser de très grands nombres.

TLS — la poignée de main invisible

Quand le cadenas s'affiche, HTTPS a fait en quelques millisecondes : vérification du certificat, échange de clé asymétrique, puis session symétrique. Ce mariage des deux méthodes est la base de tout commerce en ligne et de toute messagerie sécurisée.

Les hachages — empreintes à sens unique

SHA-256 transforme n'importe quel fichier en une empreinte de 64 caractères hexadécimaux. Changer un seul bit change entièrement l'empreinte. On ne peut pas retrouver le fichier depuis l'empreinte : ils servent à vérifier l'intégrité, jamais à stocker un secret récupérable.

03 Le schéma

✕ Symetrique (AES-256)

- Meme cle pour chiffrer et dechiffrer
- Tres rapide — gigaoctets par seconde
- Probleme : echanger la cle de facon sure
- Usage : chiffrement de fichiers, session HTTPS
- Ex : disque chiffre, VPN

✓ Asymetrique (RSA / ECC)

- Cle publique pour chiffrer, privee pour dechiffrer
- 1000× plus lent qu'AES
- La cle publique peut etre publiee partout
- Usage : echange de cle, signatures numeriques
- Ex : HTTPS, SSH, email signe

Dans la pratique, les deux sont combinés : l'asymétrique établit un secret commun, le symétrique chiffre le contenu. C'est exactement le protocole TLS utilisé à chaque connexion HTTPS.

04 À retenir

- **AES-256** : symétrique, 2^{256} combinaisons, indéchiffrable par force brute avec la technologie actuelle.
- **RSA/ECC** : asymétrique, résout l'échange de clés sans se rencontrer, mais lent.
- **TLS** combine les deux : asymétrique pour la poignée de main, symétrique pour la session.
- **Un hachage ne chiffre pas** : il empreinte sans retour possible. SHA-256 ≠ AES.

05 Exercice

Dans la console du navigateur (F12 → Console), tapez : `window.crypto.subtle.generateKey({name:'AES-GCM',length:256},true,['encrypt','decrypt']).then(k⇒crypto.subtle.exportKey('raw',k)).then(b⇒console.log(btoa(String.fromCharCode(...new Uint8Array(b)))))`. Vous venez de générer une vraie clé AES-256 dans votre navigateur. Elle n'a jamais circulé sur le réseau.

Indice — La clé générée est unique, aléatoire et ne réapparaîtra jamais. C'est exactement ce que fait votre navigateur à chaque nouvelle session HTTPS.

06 Vérification des acquis

▶ **Pourquoi ne pas tout chiffrer en asymétrique ?**

▶ **Qu'est-ce qu'un certificat TLS ?**

Chaque fiche est autonome et conçue pour être imprimée (mise en page papier optimisée).