THEORY - Cryptographie

LA POLICE DU VOCABULAIRE....2
LA TRIADE CIA (ET SES AMIS)....2
Les 3 Piliers (CIA)....2
Confidentialité (Confidentiality)....2
Intégrité (Integrity)....2
Disponibilité (Availability)....2
Les Extensions Critiques (Authenticité & Non-Répudiation)....3
LE PRINCIPE DE KERCKHOFFS (La Transparence)....3
LE MUSÉE DES HORREURS (Histoire & Faiblesses)....4
LA SCYTALE LACÉDÉMONIENNE (Le Bâton de Sparte)....4
LE CHIFFRE DE CÉSAR (Le Décalage)....4
LE CHIFFRE DE VIGENÈRE (Le Polyalphabétique)....4
LA STÉGANOGRAPHIE (L'Art de l'Invisible)....5
LA VITESSE LUMIÈRE (Chiffrement Symétrique)....6
LE CONCEPT : UNE SEULE CLÉ POUR LES GOUVERNER TOUS....6
LA MÉCANIQUE INTERNE (Sous le capot)....6
Le XOR (OU Exclusif) - L'arme absolue....6
Confusion & Diffusion (L'Effet Avalanche)....6
Le Réseau de Feistel....6
LES ALGORITHMES (Le Bon, la Brute et le Truand)....7
LE GRAAL : LE MASQUE JETABLE (One-Time Pad)....7
BEST PRACTICES....7
LE DOUBLE VERROU (Chiffrement Asymétrique)....8
LE CONCEPT : LA BOÎTE AUX LETTRES....8
La Clé Publique (Public Key)....8
La Clé Privée (Private Key)....8
Le Processus (Eve & Wall-e)....8
LES ALGORITHMES (Les Poids Lourds Mathématiques)....9
RSA (Rivest-Shamir-Adleman) - Le Patriarche....9
Diffie-Hellman (DH) – L'Échangeur....9
ECC (Elliptic Curve Cryptography) & Ed25519 - Le Futur....9
ASYMÉTRIQUE vs SYMÉTRIQUE (Le Choc des Titans)....9
L'APPROCHE HYBRIDE (PGP & SSL)....9
EMPREINTES & CERTITUDES (Intégrité & Auth)....10
LE HACHAGE (L'Empreinte Numérique)....10
Les Propriétés Vitales....10
LE ZOO DES ALGORITHMES (Hachage)....10
LE MAC & HMAC (Sceau d'Intégrité)....10
LA SIGNATURE NUMÉRIQUE (Le Sceau Royal)....11
Tableau Récapitulatif....11
SUR LE TERRAIN (Protocoles & Infrastructure)....12
SSH (SECURE SHELL) - LA LIGNE DE VIE DE L'ADMIN....12
Le Danger : Man-in-the-Middle (MITM)....12
Le Protocole de Vérification (Qui est qui ?)....12
SSL / TLS (HTTPS) - LE WEB SÉCURISÉ....13
Le "Handshake" (La Poignée de Main)....13
PKI (PUBLIC KEY INFRASTRUCTURE) - LA HIÉRARCHIE....13
PGP (PRETTY GOOD PRIVACY) - LE WEB OF TRUST....13
KERBEROS - LE CHIEN DE GARDE....13
L'ART DE LA GUERRE (Cryptanalyse & Attaques)....14
LA FORCE BRUTE (La Méthode "Bourrin")....14
LES ATTAQUES INTELLIGENTES (Le Cerveau)....14
Attaque par Dictionnaire....14
Tables Arc-en-Ciel (Rainbow Tables)....14
Analyse Fréquentielle....14
LES ATTAQUES MATHÉMATIQUES (La Chirurgie)....15
LES ATTAQUES PHYSIQUES (Side-Channel)....15
L'ATTAQUE ULTIME : LE THÉORÈME DE LA CLÉ À MOLETTE....15

LA POLICE DU VOCABULAIRE

Concept	Terme APPROUVÉ (✓)	Terme INTERDIT / FAUX (×)	Définition Technique
Action de protéger	**Chiffrement** (Encrypt)	Cryptage / Coder	Transformation d'un message clair en message inintelligible à l'aide d'une **clé**.
Action d'ouvrir (Légitime)	**Déchiffrement** (Decrypt)	Décryptage / Décoder	Reconstitution du message clair à l'aide de la **clé de déchiffrement**.
Action d'ouvrir (Attaque)	**Décryptement** (Cracking)	Crackage / Hacking	Retrouver le message clair **SANS connaître la clé**. C'est le travail du cryptanalyste.
Le résultat	Cryptogramme	Message crypté	Le message chiffré, illisible.

LA TRIADE CIA (ET SES AMIS)

Les 3 Piliers (CIA)

Confidentialité (Confidentiality)

- *Définition :* L'information n'est accessible qu'aux personnes autorisées.
- *L'ennemi :* L'écoute réseau (Sniffing), le vol de données.
- *La solution :* Le **Chiffrement**.

Intégrité (Integrity)

- *Définition :* Garantir que les données n'ont pas été modifiées (accidentellement ou par malveillance) durant le transport ou le stockage.
- *L'ennemi :* L'altération, la corruption de données.
- *La solution :* Le **Hachage** (Hash).

Disponibilité (Availability)

- *Définition :* Le système doit fonctionner quand on en a besoin. (Souvent considéré "hors-sujet" purement crypto, mais vital en sécu globale).

Les Extensions Critiques (Authenticité & Non-Répudiation)

Pour que la crypto soit utile légalement et techniquement, on ajoute deux concepts :

Authenticité (Authentication) :

- *Définition :* Prouver l'identité d'une personne ou d'une machine. "Est-ce bien Alice qui me parle ?"
- *La solution :* Signatures numériques, Certificats, Mots de passe.

Non-répudiation :

- *Définition :* Empêcher quelqu'un de nier avoir commis une action.
- *Exemple :* Si tu signes un contrat numériquement avec ta clé privée, tu ne peux pas dire "C'était pas moi". C'est l'équivalent numérique de la lettre recommandée avec accusé de réception.

LE PRINCIPE DE KERCKHOFFS (La Transparence)

C'est la règle d'or qui sépare la bonne crypto de la mauvaise crypto.

Le Principe : La sécurité d'un cryptosystème ne doit reposer QUE sur le secret de la **clé**, jamais sur le secret de l'algorithme.

Ce que ça veut dire : Tout le monde sait comment fonctionne AES. L'algorithme est public, testé par des milliers de chercheurs. La seule chose qu'on cache, c'est ta clé de 256 bits.

> **Analogie Culturelle (Serrurier) :** Tout le monde sait comment fonctionne une serrure à goupilles (le mécanisme est connu). Pourtant, sans TA clé, on ne rentre pas chez toi. Si la sécurité reposait sur le fait de cacher la serrure derrière un pot de fleurs, ce serait de la "sécurité par l'obscurité". Et c'est nul.

LE MUSÉE DES HORREURS (Histoire & Faiblesses)

LA SCYTALE LACÉDÉMONIENNE (Le Bâton de Sparte)

Ne riez pas, c'était le top de la technologie en -400 av. J.C.

- **Le Principe :** On prend un bâton d'un diamètre précis. On enroule une bandelette de cuir autour. On écrit le message dans la longueur. Une fois déroulée, la bandelette ne montre qu'une suite de lettres incohérentes.
- **Le Déchiffrement :** Le destinataire doit posséder un bâton **strictement identique** (même diamètre) pour enrouler la bandelette et lire le message.
- **La Faiblesse (Fatale) :** La sécurité repose entièrement sur le matériel (le diamètre du bâton). Si l'ennemi vole le bâton ou teste plusieurs diamètres, le message est lu. C'est l'ancêtre de la "Sécurité par l'obscurité".

LE CHIFFRE DE CÉSAR (Le Décalage)

Utilisé par Jules César pour ses correspondances militaires. Niveau de sécurité : École Primaire.

- **Le Principe (Substitution Monoalphabétique) :** On remplace chaque lettre par une autre située un nombre fixe de crans plus loin dans l'alphabet.
    - Exemple (Décalage +3) : A devient D, B devient E…
- **ROT13 :** Une variante célèbre encore utilisée (pour les blagues ou spoilers sur le net) qui décale de 13 lettres.
- **L'Attaque (Analyse Fréquentielle) :** C'est la mort de ce chiffrement. Dans chaque langue, certaines lettres reviennent tout le temps (le "E" en français ou anglais).
    - *La méthode :* Si dans le message chiffré, la lettre "X" apparaît le plus souvent, alors X = E. On en déduit le décalage et tout le reste tombe.

LE CHIFFRE DE VIGENÈRE (Le Polyalphabétique)

Le "César" sous stéroïdes. Une tentative intelligente de corriger le tir.

- **Le Principe :** Au lieu d'un seul décalage pour tout le texte, on change de décalage à chaque lettre en utilisant un **Mot-Clé**.
    - Si le mot-clé est "BAC", on décale la 1ère lettre selon B, la 2ème selon A, la 3ème selon C, et on recommence.
- **L'Avantage :** La même lettre du message clair (ex: deux "E") peut être chiffrée par deux lettres différentes. Cela brouille les pistes et résiste à l'analyse de fréquence simple.
- **La Réalité :** Ça a tenu longtemps, mais ça reste cassable (Méthode Kasiski). Si la clé est courte, les répétitions finissent par apparaître.

LA STÉGANOGRAPHIE (L'Art de l'Invisible)

Attention à la confusion ! Ce n'est PAS de la cryptographie au sens strict.

- **La Définition :** L'art de **cacher** l'existence même du message.
    - *Cryptographie :* "Je vois un coffre-fort, je ne peux pas l'ouvrir." (Message brouillé).
    - *Stéganographie :* "Je ne vois même pas le coffre-fort." (Message caché).
- **Les Techniques :**
    - Encre invisible.
    - Cacher du texte dans les pixels d'une image (LSB).
    - Tatouage numérique (Watermarking).
    - Cacher un message dans un texte anodin (ex: lire la première lettre de chaque mot).

Anecdote Culturelle (Mr. Robot) : Elliot grave ses données sur des CD audio de musique qu'il range dans un classeur. Un policier voit un album de Pink Floyd, alors que c'est un fichier chiffré. C'est de la stéganographie.

LA VITESSE LUMIÈRE (Chiffrement Symétrique)

LE CONCEPT : UNE SEULE CLÉ POUR LES GOUVERNER TOUS

Le principe est simple : la même clé sert à chiffrer (fermer) et à déchiffrer (ouvrir).

- **L'Analogie :** C'est la clé de ta chambre d'hôtel. La réception (Expéditeur) a la clé, toi (Destinataire) as la clé. C'est rapide, efficace.
- **La Vitesse :** C'est le "Faucon Millenium" de la crypto. Ça va vite, très vite. Idéal pour chiffrer des disques durs entiers ou des flux vidéo en temps réel.
- **Le Danger :** Si tu perds la clé ou si on l'intercepte pendant l'échange, c'est fini. C'est le problème majeur de la distribution des clés.

> **Règle de Survie :** Ne jamais envoyer une clé symétrique par email en clair ("Tiens, voici la clé pour ouvrir le fichier"). C'est comme laisser ta clé sous le paillasson avec un panneau néon "Bienvenue".

LA MÉCANIQUE INTERNE (Sous le capot)

Comment transforme-t-on "Bonjour" en bouillie illisible ? Avec des maths binaires.

Le XOR (OU Exclusif) - L'arme absolue

C'est l'opération reine. Simple, réversible, et destructrice pour la structure du langage.

- **Table de vérité :**
  - 0 XOR 0 = 0
  - 1 XOR 1 = 0 (Identiques = 0)
  - 0 XOR 1 = 1 (Différents = 1)
- **Propriété Magique :** $A \oplus A = 0$. Si tu fais XOR deux fois avec la même valeur, tu retrouves le message original. C'est pour ça que la même clé chiffre ET déchiffre.

Confusion & Diffusion (L'Effet Avalanche)

Pour qu'un chiffrement soit bon, il doit respecter deux principes de Claude Shannon :

1. **Confusion (Substitution / S-Box) :** Remplacer des morceaux de bits par d'autres selon une table complexe. Ça casse la logique.
2. **Diffusion (Permutation / P-Box) :** Mélanger l'ordre des bits. Si tu changes **1 seul bit** en entrée, **50% des bits** de sortie doivent changer.
   - *C'est l'Effet d'Avalanche :* Une boule de neige devient une catastrophe.

Le Réseau de Feistel

Une architecture utilisée par les vieux algos (DES).

- On coupe le message en deux blocs (Gauche / Droite).
- On "cuisine" la moitié Droite avec la clé, puis on l'ajoute à la moitié Gauche (XOR).
- On inverse Gauche et Droite.
- On répète ça 16 fois (16 tours).

LES ALGORITHMES (Le Bon, la Brute et le Truand)

Algorithme	Statut	Bloc / Clé	Description & Usage
DES (Data Encryption Standard)	☠ **MORT**	64 bits / 56 bits	L'ancêtre. Cassé. Trop court. À utiliser uniquement pour étudier l'histoire.
3DES (Triple DES)	⚠ **SURVIE**	64 bits / 112 or 168 bits	On applique DES trois fois de suite (Encrypt-Decrypt-Encrypt) pour rallonger la clé. Lent mais robuste. Utilisé dans les vieux systèmes bancaires par nostalgie.
AES (Rijndael)	🚀 **STANDARD**	128 bits / 128-256 bits	Le roi actuel. Choisi par le NIST. Rapide, inviolable (pour l'instant). Utilise des matrices de substitution et de mélange (MixColumns)

LE GRAAL : LE MASQUE JETABLE (One-Time Pad)

C'est le seul chiffrement **mathématiquement inviolable**.

- **Condition 1 :** La clé doit être **aussi longue** que le message.
- **Condition 2 :** La clé doit être totalement **aléatoire**.
- **Condition 3 :** La clé ne sert qu'une **seule fois** (Jetable).

Pourquoi on ne l'utilise pas partout ? Parce que si tu as un fichier de 1 To à chiffrer, il te faut une clé de 1 To à échanger secrètement avec ton correspondant. C'est l'enfer logistique.

BEST PRACTICES

- **Utilise AES-256.** Point final. Ne cherche pas l'originalité.
- **Mode de chiffrement :** Ne jamais utiliser le mode **ECB** (Electronic Code Book) car il laisse voir les motifs (on voit le pingouin Linux même chiffré). Utilise **CBC** ou **GCM**.
- **Génération de clé :** int getRandomNumber() { return 4; } n'est PAS un générateur aléatoire valide. Utilise des vrais générateurs d'entropie (CSPRNG).

LE DOUBLE VERROU (Chiffrement Asymétrique)

LE CONCEPT : LA BOÎTE AUX LETTRES

L'asymétrie a été inventée pour résoudre le problème de l'échange de clés. Au lieu d'une seule clé, chaque utilisateur possède un **couple de clés** (Key Pair) mathématiquement liées mais distinctes.

La Clé Publique (Public Key)

- *Rôle :* **CHIFFRER** (Fermer).
- *Distribution :* On la donne à tout le monde. On l'affiche sur son front, sur son site web, dans un annuaire.
- *Analogie :* C'est un cadenas ouvert. N'importe qui peut le prendre, fermer une boîte avec, et le verrouiller.

La Clé Privée (Private Key)

- *Rôle :* **DÉCHIFFRER** (Ouvrir).
- *Distribution :* **JAMAIS**. Elle reste chez toi, protégée par une *passphrase* ou sur une carte à puce.
- *Analogie :* C'est l'unique clé plate qui peut rouvrir le cadenas.

Le Processus (Eve & Wall-e)

- Eve veut écrire un message à Wall-e.
- Eve récupère la **Clé Publique de Wall-e**.
- Eve chiffre le message avec la **Clé Publique de Wall-e**.
- Une fois chiffré, **même Eve ne peut plus le lire**.
- Wall-e reçoit le message d’Eve et utilise sa **Clé Privée** pour le déchiffré.
- Wall-e peut lire le message.

LES ALGORITHMES (Les Poids Lourds Mathématiques)

Ici, on ne mélange pas juste des bits (XOR). On fait des maths complexes (Arithmétique modulaire).

RSA (Rivest-Shamir-Adleman) - Le Patriarche

Inventé en 1977. C'est le standard historique.

Le Moteur : Repose sur la difficulté de factoriser le produit de deux très grands **nombres premiers**.

La Taille : Pour être sécurisé aujourd'hui, une clé RSA doit faire au minimum **2048 bits**, idéalement **4096 bits**.

Performance : C'est lent. Très lent. Ça demande beaucoup de ressources CPU.

Diffie-Hellman (DH) – L'Échangeur

Ce n'est pas fait pour chiffrer un fichier, mais pour se mettre d'accord sur une clé secrète en public.

Le Principe : Eve et Wall-e mélangent leurs secrets mathématiques (exponentiation modulaire) pour arriver à un résultat commun, sans que quelqu'un qui écoute puisse deviner ce résultat.

Faiblesse : Vulnérable à l'attaque *Man-in-the-Middle* (MITM) s'il n'y a pas d'authentification (signature).

ECC (Elliptic Curve Cryptography) & Ed25519 - Le Futur

Le Principe : Utilise des courbes mathématiques au lieu des nombres premiers simples.

Avantage : Beaucoup plus performant et nécessite des clés plus petites pour la même sécurité. (Une clé ECC 256 bits est plus forte qu'une clé RSA 2048 bits).

Ed25519 : L'algorithme de référence actuel pour les clés SSH (rapide, court, sûr).

ASYMÉTRIQUE vs SYMÉTRIQUE (Le Choc des Titans)

Pourquoi s'embêter avec deux systèmes ? Parce que l'un est agile, l'autre est sécurisé pour l'échange.

Caractéristique	**Symétrique (AES)**	**Asymétrique (RSA)**
Clés	1 seule (Partagée)	2 (Publique / Privée)
Vitesse	Très Rapide (Faucon Millenium)	Lent (Destroyer Stellaire)
Taille de clé	128 / 256 bits	2048 / 4096 bits
Usage	Chiffrer les données (Disque, Flux)	Échanger la clé symétrique / Signer
Problème	Comment échanger la clé ?	Gourmand en calcul

L'APPROCHE HYBRIDE (PGP & SSL)

Comme RSA est trop lent pour chiffrer tout un film ou un disque dur, on utilise la méthode hybride. C'est le fonctionnement de **PGP** et de **HTTPS**.

Génération : Eve génère une clé **Symétrique** temporaire (clé de session).

Chiffrement Données : Elle chiffre le gros fichier avec cette clé Symétrique (Rapide).

Chiffrement Clé : Elle chiffre la clé Symétrique avec la **Clé Publique (RSA)** de Wall-e.

Envoi : Elle envoie le paquet (Fichier chiffré + Clé chiffrée).

Réception : Wall-e utilise sa **Clé Privée** pour récupérer la clé Symétrique, puis utilise la clé Symétrique pour lire le fichier.

EMPREINTES & CERTITUDES (Intégrité & Auth)

LE HACHAGE (L'Empreinte Numérique)

Le hachage (Hashing) n'est **PAS** du chiffrement. On ne déchiffre pas un Hash. C'est une fonction à sens unique (One-way function).

Le Principe : On prend un fichier de n'importe quelle taille (un mot de passe ou un film 4K) et on le passe dans une moulinette mathématique.

Le Résultat : On obtient une chaîne de caractères de taille fixe (ex: 256 bits) appelée **Empreinte**, **Condensat**, **Digest** ou **Hash**.

Les Propriétés Vitales

- **Irréversibilité :** On ne peut pas retrouver le fichier original à partir du Hash. C'est comme essayer de reconstituer une vache à partir d'un steak haché.
- **Déterminisme :** Le même fichier donnera toujours, exactement, le même Hash.
- **Effet d'Avalanche :** Si tu changes **un seul bit** dans le fichier original (une virgule, un pixel), le Hash final change totalement (environ 50% des bits de sortie changent) .
- **Anti-Collision :** Il doit être mathématiquement impossible de trouver deux fichiers différents qui donnent le même Hash.

LE ZOO DES ALGORITHMES (Hachage)

Comme pour le chiffrement, il y a les champions et les cadavres.

Algorithme	Taille (bits)	Statut	Usage / Note
MD5	128	CASSÉ	Obsolète. Des collisions sont possibles. À utiliser juste pour vérifier un téléchargement, jamais pour la sécurité.
SHA-1	160	OBSOLÈTE	Standard du gouvernement US longtemps, mais présente des faiblesses aujourd'hui.
SHA-256 (SHA-2)	256	STANDARD	Le standard actuel. Utilisé par **Bitcoin**, SSL, et pour les mots de passe.
X11	Variable	MINING	Utilisé par la crypto **Dash**. Enchaînement de **11 algos** différents (Blake, BMW, etc.). Plus complexe que SHA-256.

LE MAC & HMAC (Sceau d'Intégrité)

Le problème du Hash simple, c'est que si un pirate intercepte ton message, il peut le modifier ET recalculer le nouveau Hash. Ni vu ni connu. Pour empêcher ça, on utilise un **MAC** (Message Authentication Code).

- **Le Principe :** C'est un Hash + une **Clé Secrète**.
- **HMAC (Keyed-Hash MAC) :** On mélange le message avec une clé secrète avant de hacher.
    - Seul celui qui possède la clé secrète peut vérifier si le Hash est bon.
- **Résultat :** Garantit l'**Intégrité** (pas modifié) ET l'**Authenticité** (vient de quelqu'un qui a la clé).

LA SIGNATURE NUMÉRIQUE (Le Sceau Royal)

C'est l'application suprême de la cryptographie asymétrique. Attention, c'est le **processus inverse** du chiffrement de confidentialité .

Le Processus (Eve Signe pour Wall-e)

1. **Hachage :** Eve calcule le Hash (Condensat) de son message.
2. **Signature :** Eve chiffre ce Hash avec sa **CLÉ PRIVÉE**.
   - *Note :* On ne chiffre pas tout le message (trop lent), juste son empreinte.
3. **Envoi :** Eve envoie le Message (clair ou chiffré) + La Signature (Hash chiffré).
4. **Vérification (Wall-e) :**
   - Wall-e déchiffre la signature avec la **CLÉ PUBLIQUE** de Eve. Il obtient le "Hash Original".
   - Wall-e calcule lui-même le Hash du message reçu.
   - Il compare les deux.
5. **Conclusion :** Si les Hashs sont identiques, cela prouve deux choses :
   - **Intégrité :** Le message n'a pas changé d'un iota.
   - **Authenticité & Non-Répudiation :** Seule Eve (qui a la clé privée) a pu créer cette signature.

Tableau Récapitulatif

Objectif	**Clé utilisée par l'émetteur**	**Clé utilisée par le récepteur**
Confidentialité (Secret)	Clé **Publique** du destinataire	Clé **Privée** du destinataire
Authentification (Signature)	Clé **Privée** de l'émetteur	Clé **Publique** de l'émetteur

SUR LE TERRAIN (Protocoles & Infrastructure)

SSH (SECURE SHELL) - LA LIGNE DE VIE DE L'ADMIN

Le protocole standard pour gérer des machines à distance.

Il combine tout : Diffie-Hellman pour l'échange, RSA/Ed25519 pour l'auth, et AES pour le tunnel.

Le Danger : Man-in-the-Middle (MITM)

Le protocole d'échange de clés (Diffie-Hellman) est génial, mais il a un défaut : il ne vérifie pas l'identité.

- *Le Scénario :* Tu crois parler au Serveur A, mais tu parles à Eve (l'attaquante). Eve relaie tout au Serveur A. Elle lit tout.
- *La Parade :* L'Authentification des machines.

Le Protocole de Vérification (Qui est qui ?)

Qui est vérifié ?	**Par quel moyen ?**	**Détail Technique**
Le Serveur	Fingerprint	À la 1ère connexion, le client affiche l'empreinte (Hash) de la clé publique du serveur. Tu DOIS vérifier qu'elle correspond à la vraie. Si ça change un jour : ALERTE ROUGE (ou réinstallation).
Le Client	Clé Privée	Le serveur envoie un "challenge" chiffré avec ta clé publique. Seule ta Clé Privée peut le résoudre. C'est la preuve que c'est toi.
Protection Locale	Passphrase	Mot de passe qui chiffre ta clé privée sur TON disque dur. Si on vole ton PC, la clé est illisible sans ce code.

SSL / TLS (HTTPS) - LE WEB SÉCURISÉ

L'architecture qui protège tes achats sur Amazon.

C'est l'application parfaite de l'approche **Hybride**.

Le "Handshake" (La Poignée de Main)

1. Client Hello : "Bonjour, je veux du HTTPS."
2. Server Hello + Certificat : "Ok. Voici mon Certificat (Carte d'identité validée par une autorité) contenant ma Clé Publique."
3. Vérification : Le navigateur vérifie si le Certificat est valide (pas périmé, émis par une autorité de confiance).
4. Échange de Clé (Session Key) :
    - Le client génère une clé Symétrique (AES) aléatoire.
    - Il la chiffre avec la Clé Publique du serveur.
    - Il l'envoie.
5. Tunnel Chiffré : Le serveur déchiffre la clé AES avec sa clé Privée. Maintenant, les deux utilisent cette clé AES pour toute la suite (Vitesse maximale).

PKI (PUBLIC KEY INFRASTRUCTURE) - LA HIÉRARCHIE

Comment savoir si la Clé Publique de Google est vraiment celle de Google ?

Si n'importe qui peut créer une clé publique, comment faire confiance ? C'est le rôle de la **PKI**.

- **CA (Certificate Authority) :** L'entité "Dieu". C'est une organisation (ex: Verisign, Let's Encrypt) en qui les navigateurs ont confiance par défaut.
- **Le Certificat Numérique :** C'est une Clé Publique **signée** par une CA.
    - *Analogie :* Ta carte d'identité. C'est un bout de plastique avec ta photo, mais elle a valeur légale car elle est signée par la "République Française" (CA).

PGP (PRETTY GOOD PRIVACY) - LE WEB OF TRUST

L'alternative rebelle à la PKI.

Utilisé pour chiffrer les emails ou signer des fichiers.

- Fonctionnement Hybride :
    1. Eve génère une clé de session unique.
    2. Eve chiffre le message avec cette clé (Symétrique).
    3. Eve chiffre la clé de session avec la Clé Publique de Wall-e (RSA) .
    4. Elle envoie le tout (Message chiffré + Clé chiffrée).
- **Confiance (Web of Trust) :** Pas d'autorité centrale (CA). Je fais confiance à la clé de Wall-e parce que Eve l'a signée, et je fais confiance à Eve.

KERBEROS - LE CHIEN DE GARDE

Le standard en entreprise (Windows AD).

- **Principe :** Authentification par **Tickets**.
- **Fonctionnement :** Tu ne donnes pas ton mot de passe au serveur final. Tu t'identifies une fois auprès du "Key Distribution Center" (KDC), qui te donne un "Ticket". Tu présentes ce ticket aux serveurs pour entrer .
- **Sécurité :** Tout repose sur des clés symétriques et des horodatages stricts.

L'ART DE LA GUERRE (Cryptanalyse & Attaques)

LA FORCE BRUTE (La Méthode "Bourrin")

Essayer toutes les clés possibles jusqu'à ce que ça marche.

- **Le Principe :** On teste la clé 000...01, puis 000...02, etc.
- **La Réalité Mathématique :**
  - Contre un code PIN (4 chiffres) : C'est instantané (10^4).
  - Contre **AES-128** : Il faut tester 2128 possibilités ($3,4 \times 10^{38}$). Même avec tous les ordinateurs de la Terre, il faut des milliards d'années.
- **Conclusion :** Inefficace contre les algorithmes modernes (AES), mais fatal contre les mots de passe faibles.

LES ATTAQUES INTELLIGENTES (Le Cerveau)

Quand la porte est trop solide (AES), on cherche à deviner la clé autrement.

Attaque par Dictionnaire

- **Cible :** Les mots de passe humains.
- **Méthode :** On teste tous les mots du dictionnaire, les prénoms, les dates, et les variantes ("password123").
- **Contre-mesure :** Utiliser des mots de passe longs, aléatoires et salés (Salt).

Tables Arc-en-Ciel (Rainbow Tables)

- **Type :** Compromis Temps / Mémoire.
- **Méthode :** Au lieu de recalculer le Hash de chaque mot de passe à chaque fois (ce qui prend du temps), le pirate pré-calcule des milliards de Hashs et les stocke dans des tables géantes.
- **L'Attaque :** Il compare ton Hash volé avec sa base de données. Si ça matche, il a ton mot de passe instantanément.
- **Contre-mesure :** Le **SALT** (Grain de sel). Ajouter des caractères aléatoires avant de hacher rend les tables pré-calculées inutiles.

Analyse Fréquentielle

- **Cible :** Chiffrements anciens (César, Vigenère).
- **Méthode :** On compte l'apparition des lettres. Si 'X' apparaît 15% du temps, c'est probablement un 'E'.
- **Efficacité :** Nulle contre AES ou RSA.

LES ATTAQUES MATHÉMATIQUES (La Chirurgie)

Pour les cryptographes de haut vol (NSA, chercheurs).

- **Cryptanalyse Linéaire :** Cherche des approximations linéaires entre le texte clair et le chiffré. Efficace contre le vieux **DES**, mais inefficace contre les systèmes modernes.
- **Cryptanalyse Différentielle :** On regarde comment une modification mineure dans l'entrée change la sortie. Nécessite énormément de données.

LES ATTAQUES PHYSIQUES (Side-Channel)

Quand on ne peut pas casser les maths, on attaque la physique de la machine.

On appelle ça l'attaque par **Canal Auxiliaire**. On n'attaque pas l'algorithme, mais son implémentation "dans le monde réel" .

- **Timing Attack :** Si le processeur met 0.001s de plus pour vérifier un mot de passe incorrect qui commence par la bonne lettre, on peut deviner la lettre.
- **Consommation Électrique / Bruit :** En analysant les fluctuations de courant ou le bruit du ventilateur/processeur, on peut parfois déduire les opérations mathématiques en cours (et donc la clé).

Note : C'est de la haute voltige. Réservé aux laboratoires ou aux espions qui ont accès physique à votre serveur.

L'ATTAQUE ULTIME : LE THÉORÈME DE LA CLÉ À MOLETTE

La "Rubber-Hose Cryptanalysis". La fin de partie.

C'est l'attaque la plus efficace, la moins chère et la plus redoutée. Elle contourne 100% des pare-feux et des cryptages RSA-4096 bits.

- **Le Scénario :** L'attaquant n'essaie pas de casser le code avec un supercalculateur à 1 million de dollars. Il achète une clé à molette à 5 dollars.
- **L'Action :** Il menace (ou frappe) le détenteur de la clé jusqu'à ce qu'il donne le mot de passe.
- **La Leçon :** La cryptographie protège les données logiques, pas les êtres humains. La faille se situe souvent entre la chaise et le clavier (Ingénierie Sociale, Phishing, Menace).