Rappel des systèmes symétriques et asymétriques

Fonctions de hachage

Signature numérique

Cours 04

Ahmed Amou El Arby

Problématique

- Failles dans les protocoles de communication
  - Toute information circulant sur Internet peut être capturée et enregistrée et/ou modifiée

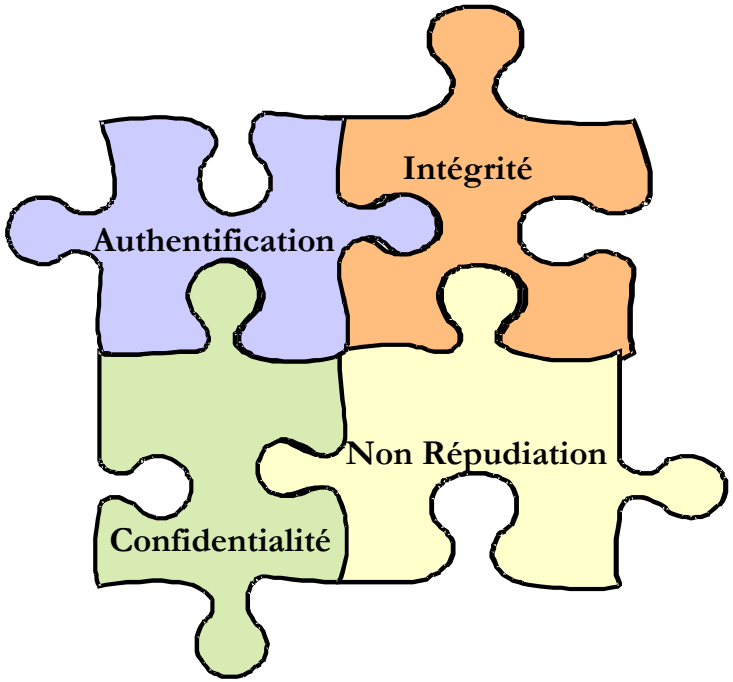
    Problème de confidentialité et d'intégrité
  - Toute personne peut falsifier son adresse IP (*spoofing*) ce qui engendre une fausse identification

    Problème d'authentification
  - Aucune preuve n'est fournie par Internet quant à la participation dans un échange électronique

    Problème d'absence de traçabilité

Cryptographie

- Science mathématique permettant d'effectuer des opérations sur un texte intelligible afin d'assurer une ou plusieurs propriétés de la sécurité de l'information.



Définition d'un crypto-système

Un crypto-système est décrit par cinq uplets (P,C,K,E,D), satisfaisant ces conditions:

- « P » est un ensemble fini de textes clairs (Plain text)
- « C » est un ensemble fini de textes cryptés (Cypher text)
- « K » est l'espace de clés (*key space*), représente un ensemble fini de clés possibles.
- Pour chaque k € K, il existe une fonction cryptage e_k € E, et une fonction de décryptage correspondante d_k € D
  - Les fonctions $e_k : P \rightarrow C$ et $d_k : C \rightarrow P$ doivent satisfaire:

$$d_k(e_k(x))=x \text{ pour chaque } x \in P$$

Principaux objectifs

- Le texte clair ne doit pas être facilement obtenu à partir d'un texte crypté.
- Les clés ne doivent pas être facilement obtenues à partir d'un texte crypté.
- L'espace des clés doit être assez large pour résister aux attaques *brute-force*.

Cryptanalyse

- Principes et méthodes permettant de trouver un message clair à partir d'un message crypté sans connaissance de la clé.
- Attaques classifiées selon le type de connaissance disponible pour l'intrus (c*ryptanalyst*).
- Connaissant C=E(P,K) mais pas K, l'objectif est de trouver P ou K.
- Types d'attaques de cryptanalyse:
  - Texte chiffré uniquement: uniquement C et E sont connus par l'intrus
  - Texte clair connu: Uniquement E, C, et quelques paires de messages clairs/cryptés avec K, sont connus par l'intrus
  - Texte clair choisi: E, C, sont connus, et P a été choisi par l'intrus.
  - …

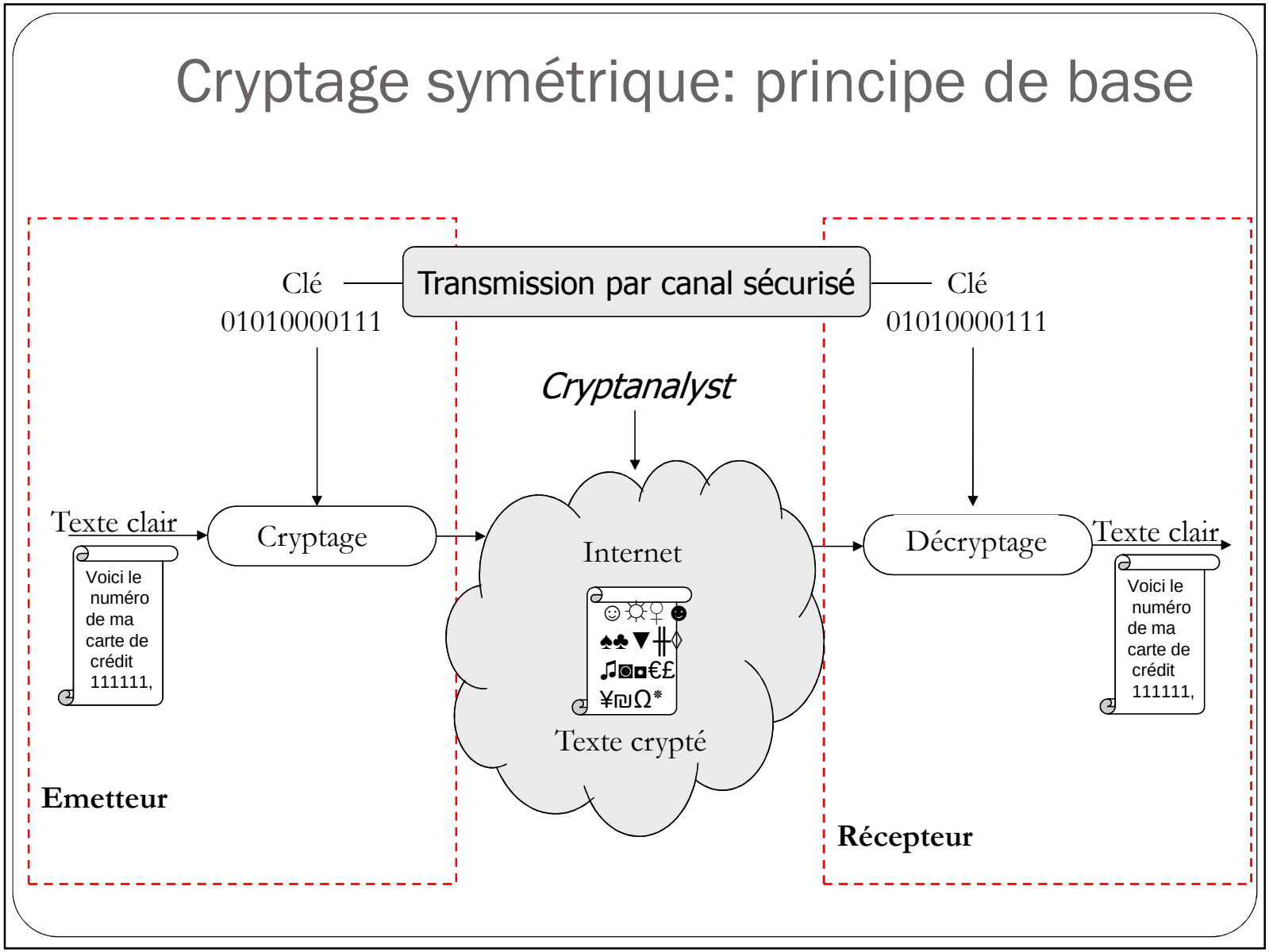
Cryptage symétrique

- Exigences:
  - Un algorithme de cryptage solide.
  - Une clé secrète partagée et connue entre l'émetteur et le récepteur.

    $Y = E_K(X)$

    $X = D_K(Y)$
- Suppose que l'algorithme de cryptage est connu à l'avance.
- Les clés sont distribuées à travers des canaux sécurisés.
- Exemples :
  - Algorithmes : DES, IDEA, AES
  - Taille des clés : 56-128-192-256-… bits

Cryptage symétrique: principe de base



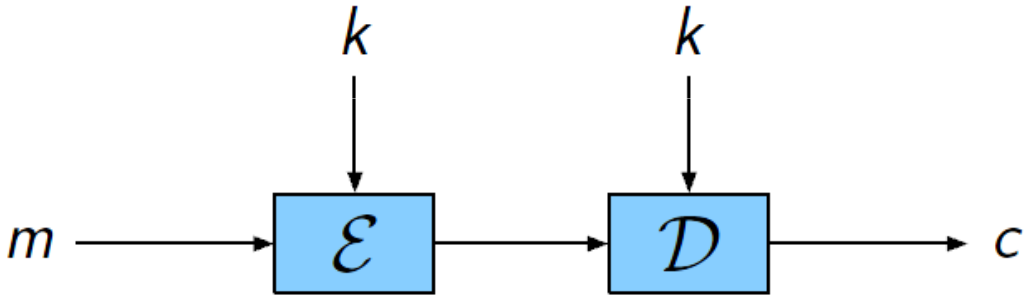
Cryptage symétrique: Modes Opérationnels

- Cryptage par flux (*Stream Cipher*)
  - Principe: Traite les éléments d'entrée de façon continue, produisant à la fois un élément de sortie (crypté).
  - La clé est aussi longue que le stream de données.
  - Mode adapté pour la communication en temps réel: Pas besoin d'attendre l'arrivé du block entier
  - Implémenté en général sur des supports hardware.
- Cryptage par bloc (*Bloc Cipher*)
  - Principe: Le texte est divisé en différents blocks de taille fixe. Un block est traité à la fois, produisant un block de données cryptées.
  - le block doit être entièrement disponible avant le traitement
  - La même fonction et la même clé est utilisée pour crypter les blocks successifs.
  - Implémentation d'une manière logicielle en générale.

Chiffrement Par Flux

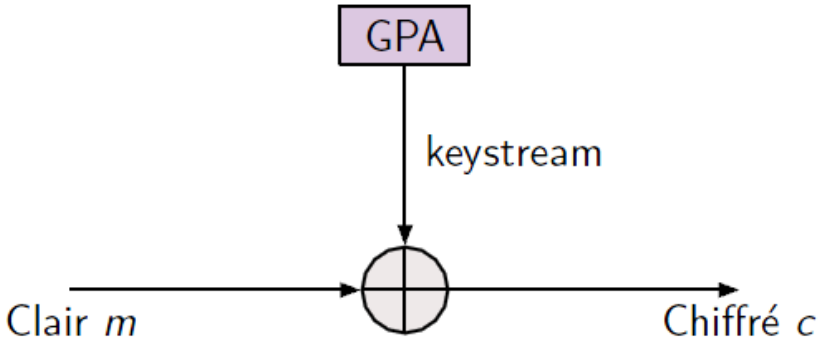
Chiffrement symétrique

La clé de chiffrement est la même que la clé de déchiffrement



Chiffrement Par Flux

- Caractéristiques
- Pas besoin de lire le message ni d'avoir sa longueur pour commencer a chiffrer
- Génération de pseudo-aléa, appelé flux de clé (keystream) que l'on combine (souvent par XOR) avec le flux de donnees



Chiffrement Par Flux

Les chiffrements par flots sont très utilisés pour protéger les données multimédia:

- RC4 (utilisé dans SSL et dans le WiFi 802.11)
- E0/1 (norme Bluetooth)
- A5/1, A5/2, A5/3 (utilisés dans le GSM)

Cryptographie Symétrique: opérations de base

- **Substitution**
  - Remplacement de chaque élément (bit, lettre, groupe de bits ou de lettres) dans le texte clair par un autre élément.
- **Transposition**
  - Réarrangement des éléments du texte clair

➔**La plupart des systèmes utilisent plusieurs étapes de transposition et de substitution.**

➔**Aucune information ne doit être perdue durant ces deux opérations**

Exemple de cryptage par substitution

Exemple: *Caesar's cipher*

- Etapes:
  - Clé = 3
  - Remplacer chaque lettre par celle qui la succède de trois (3).
  - a devient d, b devient e, …, y devient b, z devient c
  - L'algorithme peut être décrit comme suit:
    - C = E(p) = (p+3) mod (26)
- Problèmes rendant la cryptanalyse de cet algorithme simple:
  - Algorithme de cryptage et de décryptage connu.
  - Seulement 25 clés à essayer.
  - Le langage du message clair est connu et facilement identifiable.

Cryptographie Symétrique : exemples

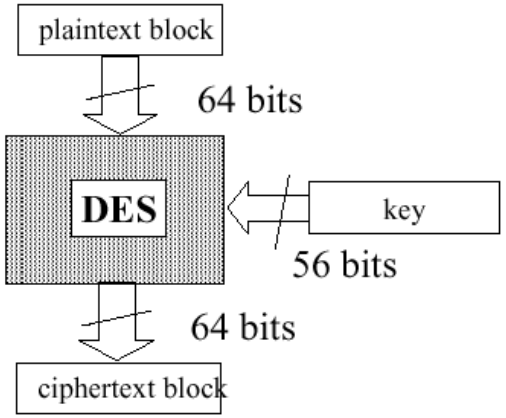
- **Algorithmes de chiffrement en continu (Stream Cipher)**
  - Exemple : RC4 (RSA Security)
    - Taille de la clé variable (128 bits en pratique).
- **Algorithmes de chiffrement par blocs (Block Cipher)**
  - Chiffrement par blocs de texte clair: 64 bits (DES), 128 bits (AES).
  - DES (clé 56 bits), 3DES (clé de 168 bits ou 112 bits).
  - RC2 (clé 128 bits), Blowfish (clé 128bits, jusqu'à 448 bits), AES (clé 128, 192, 256 bits).

Cryptage symétrique

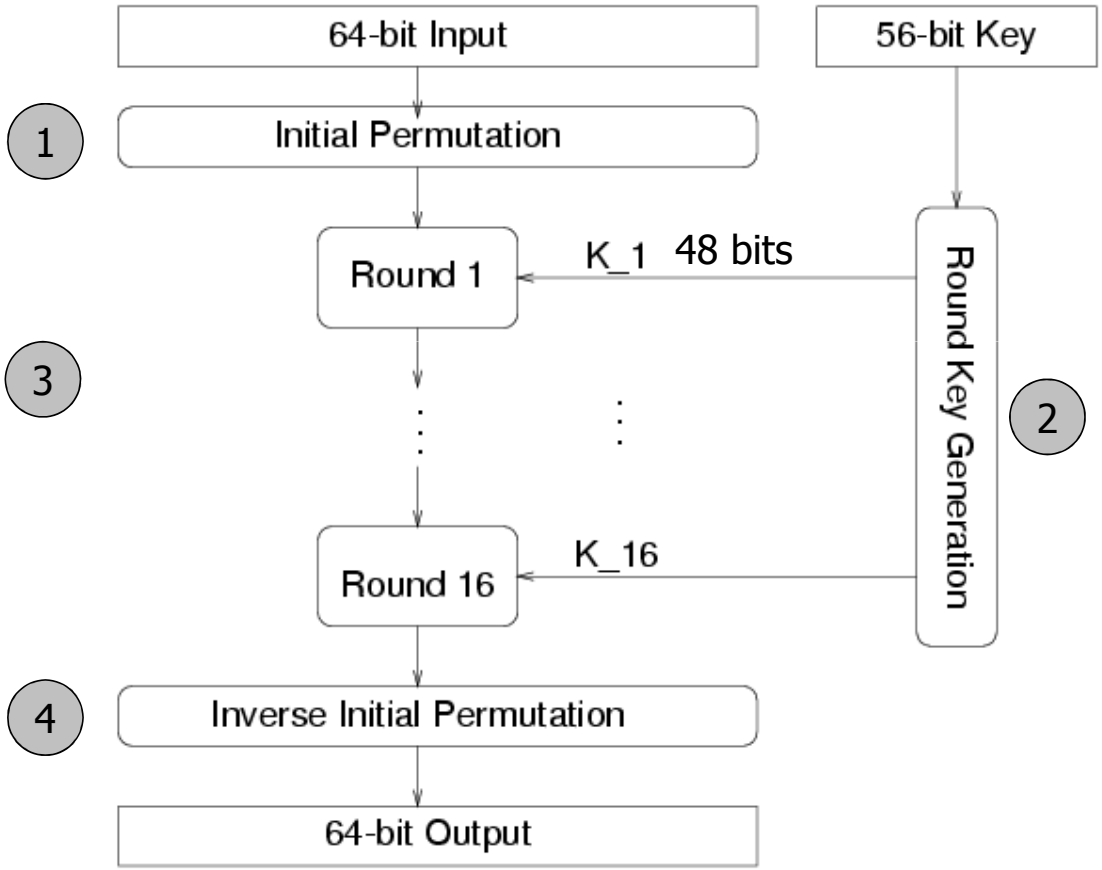
- Avantages
  - Confidentialité des données.
  - Rapidité, et facilité de mise en œuvre sur des circuits.
- Limitations
  - Problématique de l'échange de la clé de chiffrement
    - Établissement préalable d'un canal sûr pour la transmission de la clé
  - Une tierce partie ne peut pas s'assurer de l'authenticité des messages.
  - Problème de la distribution des clés de cryptage
    - Nécessité d'établir un canal sécurisé pour la transmission de la clé
    - Nombre de clés échangées (en n^2).

DES (Data Encryption Standard)

- L'algorithme de cryptage (Block cipher) à clés symétriques le plus utilisé.
- Crypte des blocks de 64 bits en utilisant des clés relativement courtes (taille effective 56-bit).
- Produit de transpositions et de substitutions.
- Implémentation facile en matériel.
  - Boites transposition P-Box
  - Boites de substitution S-Box



Algorithme DES

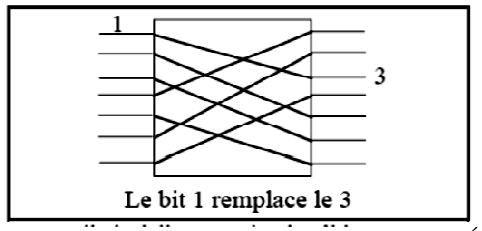


DES (étapes 1 et 3): P-Box

IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

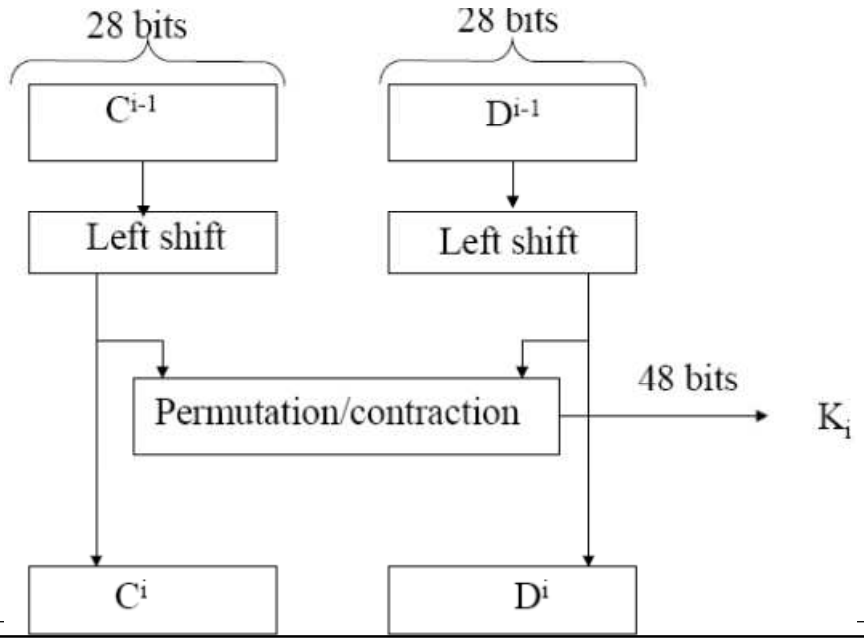
IP^{-1}							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

- Le bit numéro 1 deviendra à la position 58
- Implémentation simple en matériel

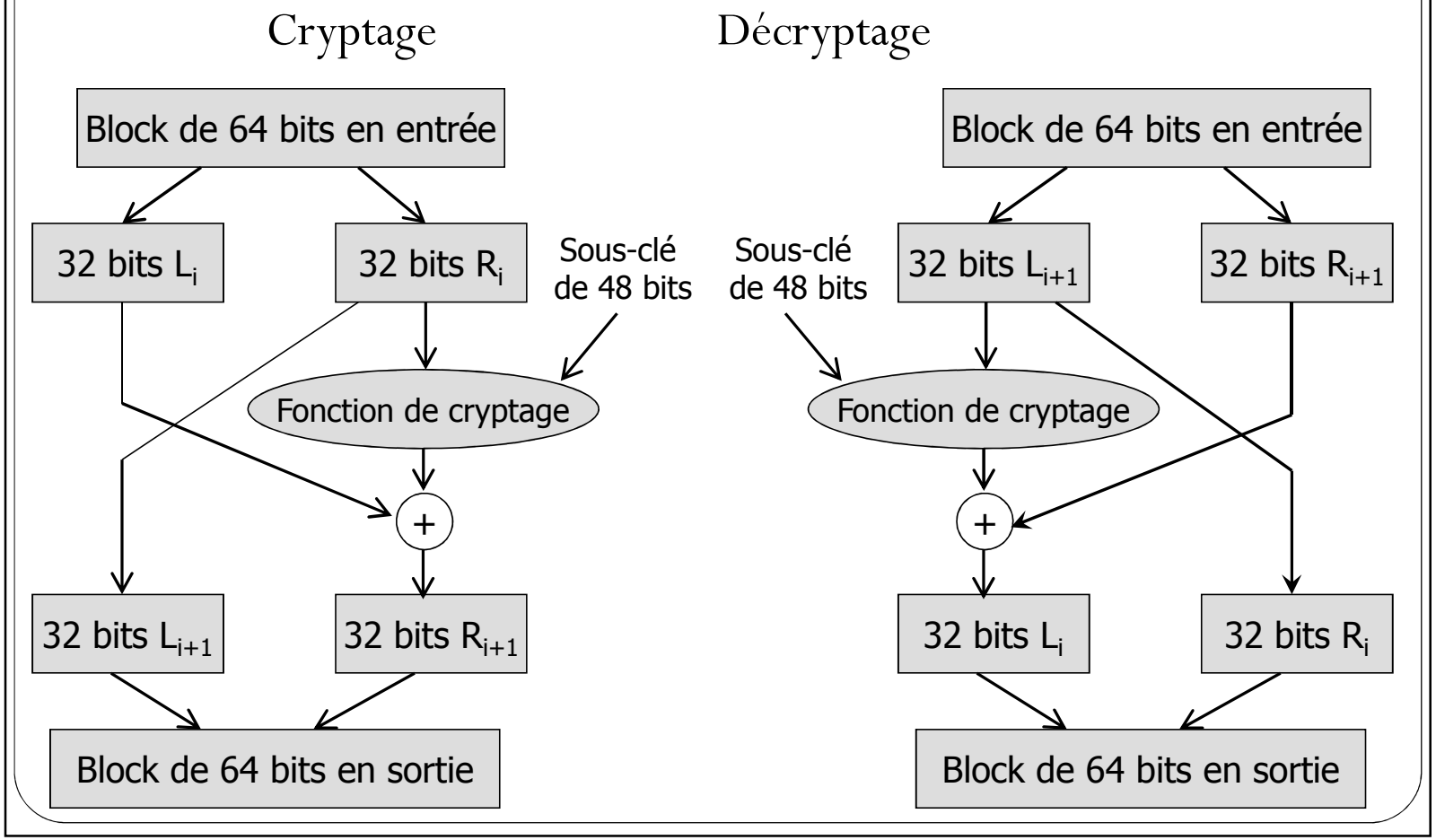


DES (étape 2)

- Les sous-clés *(Round keys)* sont générées à partir de la clé principale de 56 bits:
  - Diviser la clé de 56 bits en deux segments.
  - Rotation de chaque segment par un ou deux bits à droite.
  - Sélection de 24 bits de chaque segment.



DES (étape 3) Un tour DES (One DES round)



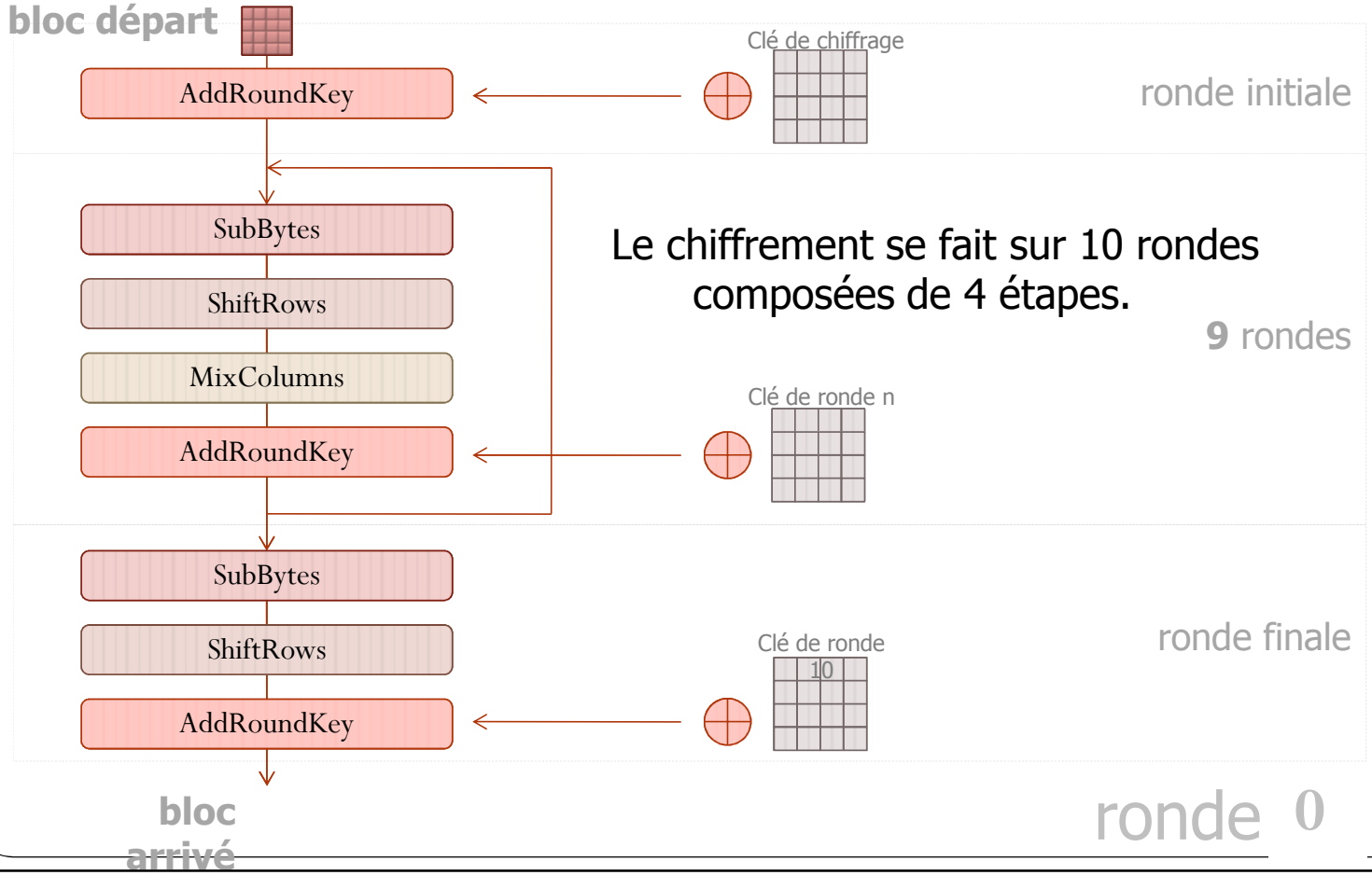
Comparatif DES AES

	DES	AES
Taille de clé	56 bits	128, 192 ou 256 bits
Type de chiffre	Chiffre à bloc symétrique	Chiffre à bloc symétrique
Taille de bloc	64 bits	128, 192 ou 256 bits
Resistance à la cryptanalyse	Vulnérable à la cryptanalyse linéaire et différentielle; tables de substitution faibles	Résistant contre des attaques différentielles, linéaires et par interpolation.
Sécurité	Prouvé comme inadéquat	Considéré sécurisé
Clés possibles	2^{56}	2^{128},2^{192} ou 2^{256}
Clés possibles composées de caractères ASCII affichables*	95^{7}	95^{16}, 95^{24} ou 95^{32}
Recherche sur toutes les clés possibles à 50 milliard de clés par seconde**	Pour une clé de 56 bits: 400 jours	Pour une clé de 128 bits: 5 x 1021 years

* - Il y a 95 caractères ASCII affichables
** - Temps affiché pour une recherche sur 100% des clés. ¨En théorie, une clé peut être trouvée après 50% de recherche.

Source : CISCO

AES: Aperçu général



AES: Domaines d'utilisation

- Chiffrement de documents confidentiels
  - L'algorithme AES est utilisé pour protéger des documents confidentiels.

The Committee on National Security Systems

- Voix sur IP: Skype
  - "Skype uses AES (Advanced Encryption Standard), [...] which is used by U.S. Government organizations to protect sensitive, information."

- WiFi: pour le WPA2 (norme IEEE 802.11i)
  - Utilise l'algorithme AES plutôt que RC4 (utilisé par WEP et WPA)

- Compression de fichiers: WinRAR et WinZip

Cryptographie asymétrique (1)

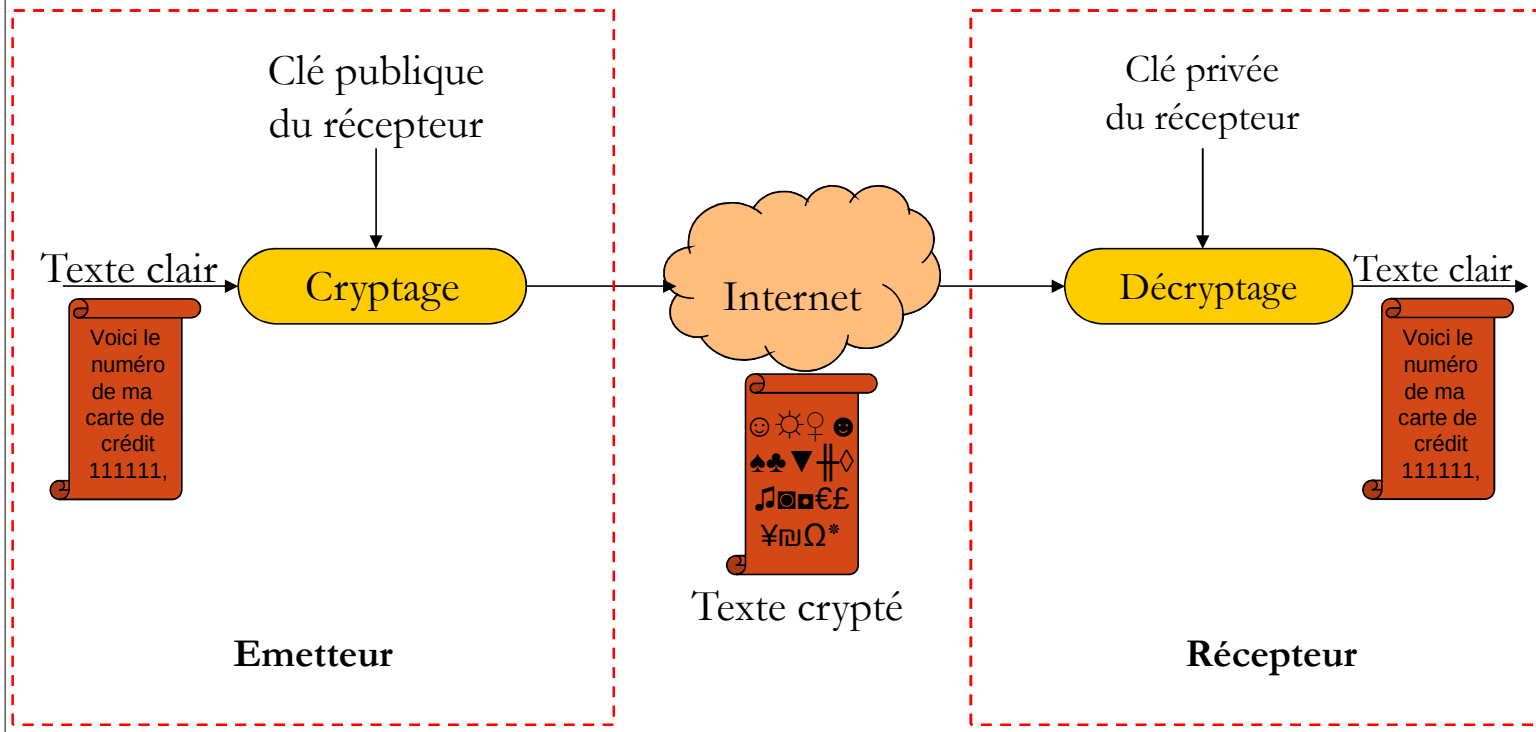
- Appelé aussi: *cryptographie à clé publique / à paire de clés / asymétrique*
- Représente une révolution dans l'histoire de la cryptographie
- Utilisation de deux clés:
  - Clé publique: Connue par tout le monde, et peut être utilisée pour crypter des messages ou pour vérifier la signature.
  - Clé privée: Connue par le récepteur uniquement, utilisée pour décrypter les messages, ou pour créer la signature.
- Si on crypte avec l'une de ces clés le décryptage se fait uniquement avec l'autre.
- Impossible de trouver la clé privée à partir de la clé publique.

Cryptographie asymétrique (2)

- Clés à grande taille (ex: RSA: 1024-2048-…).
- Fonction trappe à sens unique
  - K_{pr}: clé privée, K_{pu}: clé publique
  - $Y=f_{K_{pr}}(X)$ facile à calculer si K_{pr} et X sont connus.
  - $X=f_{K_{pu}}^{-1}(Y)$ facile si K_{pu} et Y sont connus, mais impossible si Y est connu et K_{pu} non connue.
- Utilisé généralement pour
  - Cryptage / décryptage: assurer la confidentialité.
  - Signature numérique: assurer l'authentification et la non répudiation.
  - Distribution de clés: se mettre d'accord sur une clé de session.

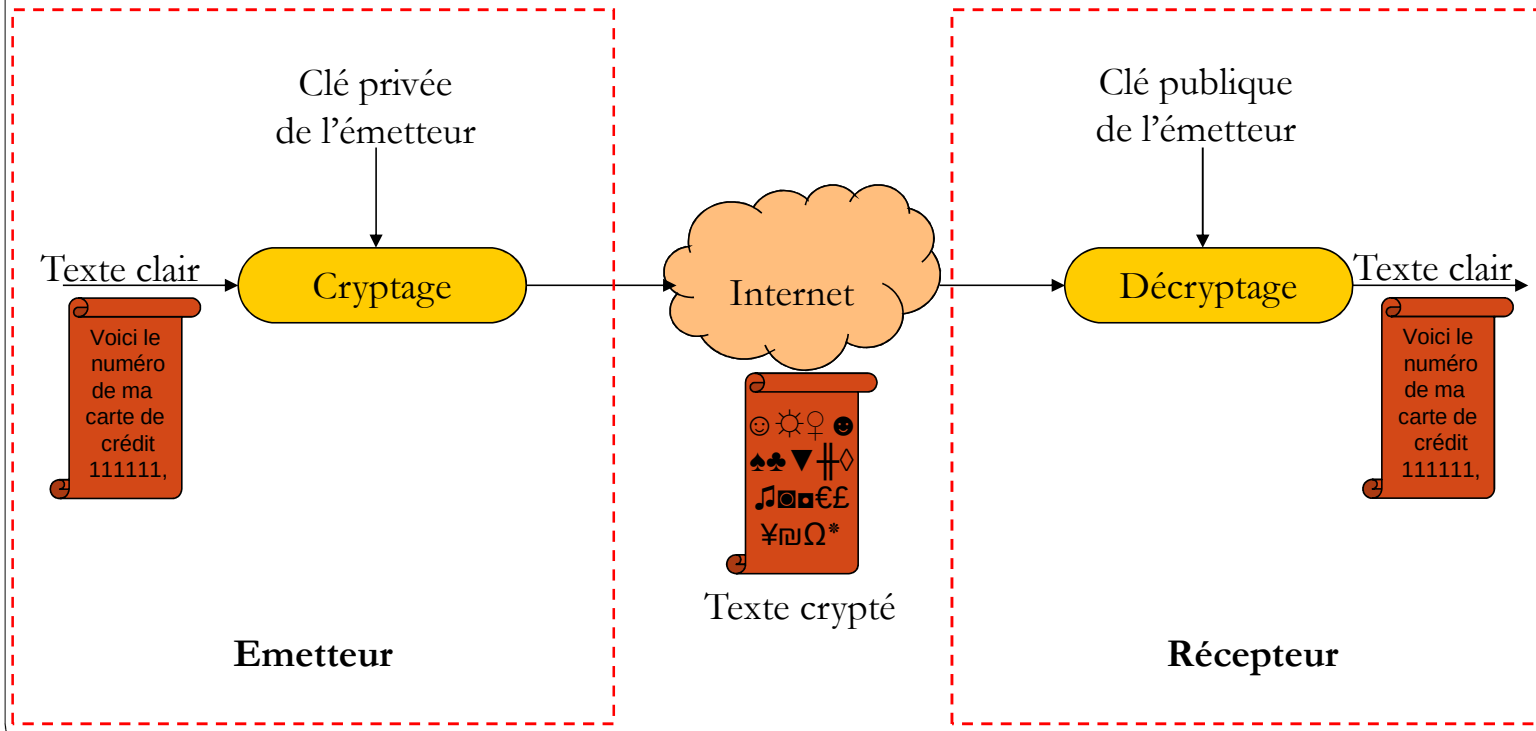
Cryptographie asymétrique: scénarios d'utilisation

Scénario: confidentialité



Cryptographie asymétrique: scénarios d'utilisation

Scénario: authenticité de l'émetteur et non répudiation d'envoi



Cryptographie asymétrique : exemples

- **RSA**
  - Développé par Rivest, Shamir & Adleman à MIT en 1977, publié en 1978.
  - Le plus connu et le plus utilisé comme algorithme de cryptage asymétrique : utilisé pour le cryptage et la signature électronique.
  - Utilise des entiers très larges 1024+ bits
  - La sécurité repose sur le coût de factorisation des entiers larges.
- **Diffie-Hellman**
  - Algorithme utilisé pour l'échange et la distribution des clés symétriques.

Cryptage asymétrique: Avantages et inconvénients

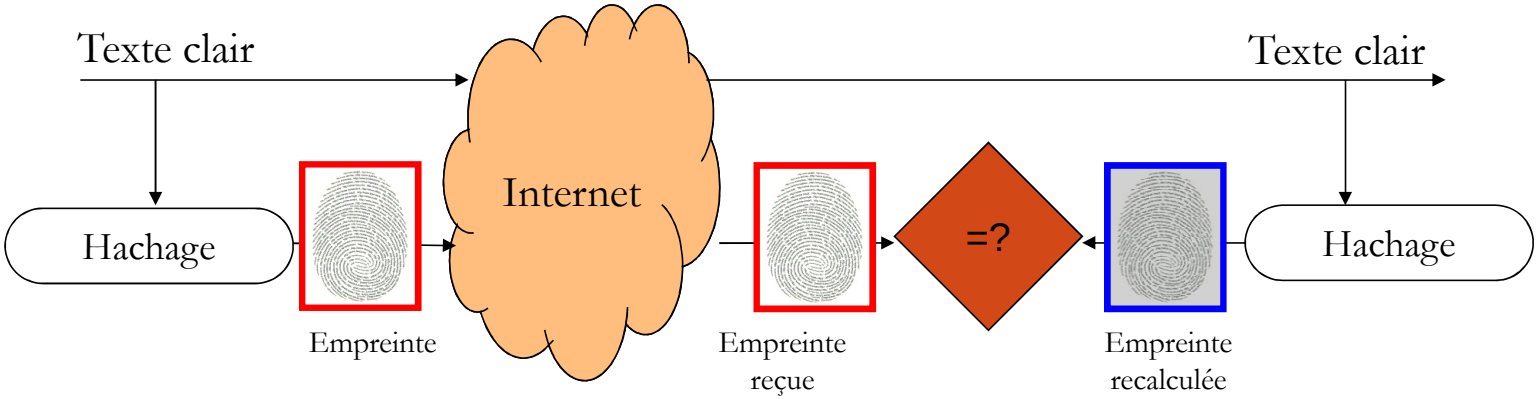
- Avantages
  - Pas besoin d'établir un canal sûr pour la transmission de la clé.
  - Plusieurs fonctions de sécurité: confidentialité, authentification, et non-répudiation
- Inconvénient
  - Généralement dix fois plus lent que le cryptage symétrique.
  - Problème d'implémentation sur les équipements disposants de faible puissance de calcul (ex: cartes bancaire, stations mobiles, etc.)
    - Clés longues
    - Complexité algorithmique de la méthode (ex: réalisation des opérations modulo n)

➔ Solution: Utilisation du cryptage asymétrique pour l'échange des clés secrètes de session d'un algorithme symétrique à clés privées.

Fonction de hachage

- Entrée: message M avec contenu et taille arbitraire.
- Sortie: message de taille fixe h=H(M).
- La fonction de hachage permet d'extraire une empreinte qui caractérise les données.
  - Une empreinte a toujours une taille fixe indépendamment de la taille des données.
- Irréversible:
  - Etant donnée h, il est difficile de trouver x tel que: h = H(x)
  - Complexité de l'ordre de 2^n, n est le nombre de bits du *digest*.
- Calcul facile et rapide (plus rapide que le cryptage symétrique).
- Exemples:
  - MD5, SHA, …
  - Taille du *digest*: 128-160-… bits

Fonctions de Hachage: Principes



1)  **=**

Empreinte reçue — Empreinte recalculée

Le texte reçu est intègre

2)  **≠**

Empreinte reçue — Empreinte recalculée

Le texte reçu est altéré

Propriétés d'une fonction de hachage

- Irréversible
  - Soit « y » le résultat de hachage, il est pratiquement infaisable de trouver « x » tel que h(x)=y.
- Résistance forte à la collision:
  - Soit « x » et « y=h(x) », il est pratiquement infaisable de trouver «x'≠x » tel que h(x')=h(x).
  - Il est pratiquement infaisable de trouver deux valeurs distinctes « x' » et « x » tel que h(x')=h(x).

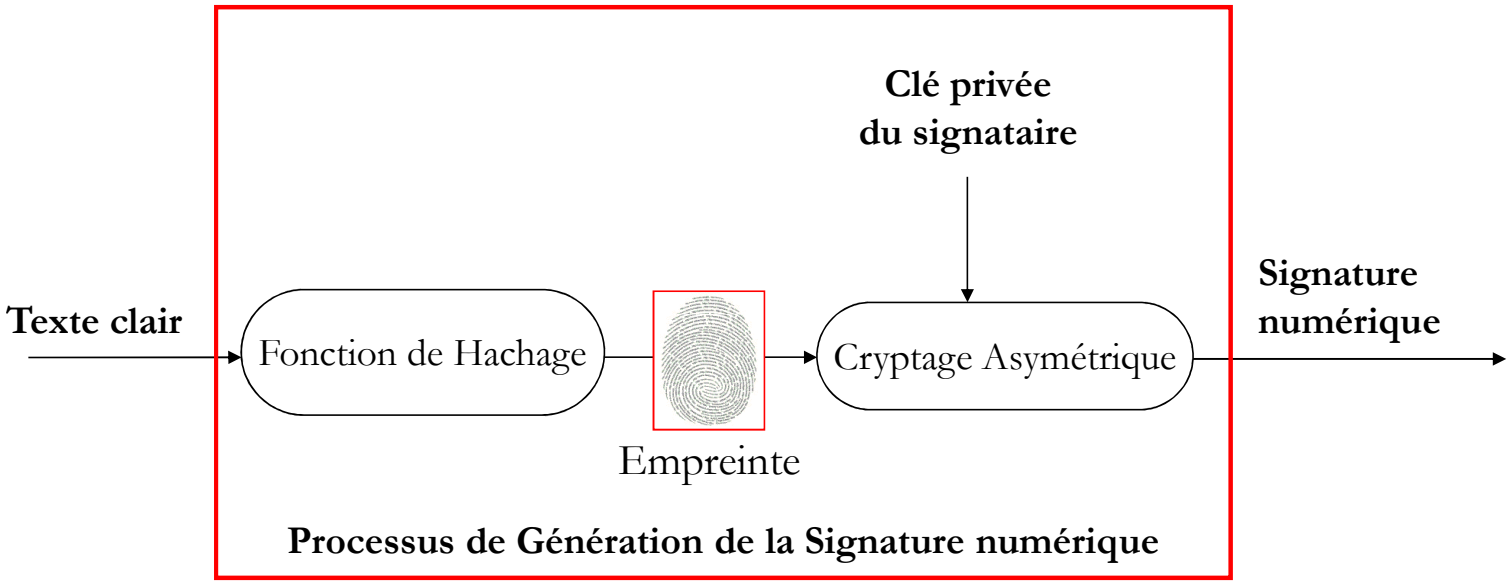
Fonctions de Hachage: Exemples

- **MD5** : *Message Digest 5*
  - Développé en 1991
  - Génère une empreinte de taille 128 bits en traitant les données d'entrée par blocs de 512 bits.
- **SHA-1** : *Secure Hash algorithm*
  - Génère une empreinte de taille 160 bits.
  - Plus fort que MD5.

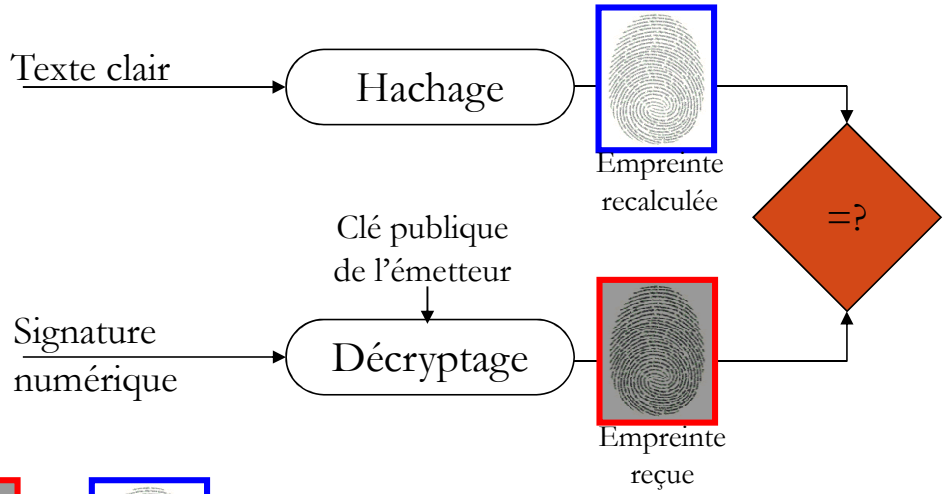
Signature numérique

- Principe de fonctionnement
  - Le *Hash* (résultat de la fonction de hachage) d'un message est crypté avec la clé privée de l'émetteur.
  - La clé publique est utilisée pour la vérification de la signature
- Soit:
  - M: message à signer, H: fonction de hachage
  - Kpr, Kpu: paire de clés privée/publique de l'émetteur.
  - E / D: fonction de cryptage / Décryptage en utilisant Kpu / Kpr.
- En recevant $(M, E_{Kpr}(H(M)))$, le récepteur vérifie si: $H(M)=D_{Kpu}(E_{Kpr}(H(M)))$

Signature numérique: Génération



Signature numérique: Vérification



1)  =

Empreinte reçue — Empreinte recalculée

La signature reçue est correcte

2)  ≠ 

Empreinte reçue — Empreinte recalculée

La signature reçue est incorrecte

Signature numérique VS Signature Manuscrite

- Les deux signatures (numérique & manuscrite) assurent:
  - Authentification du signataire
  - Non répudiation.
- La signature numérique, seule, assure l'intégrité des données.