

# Sécurité

**Sécurité des Systèmes d'Information  
Concepts, Organisation, Outils et Tendance**

J. Saraydaryan

CPE - Lyon



# Cryptologie et Applications

Sécurité des Systèmes d'information  
Concepts, Organisation, outils et Tendance

J. Saraydaryan

CPE - Lyon



## I Introduction et définitions

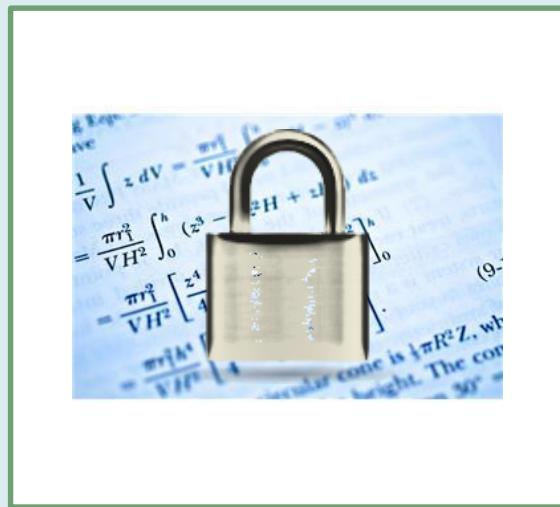
## II Chiffrement Symétrique

## III Chiffrement Asymétrique

## IV Fonction à sens unique

## V PKI

## VI Sécurité de l'Internet



## Introduction et définition

- 
- Historique
  - Définitions et concepts
  - Type de chiffrement
  - Méthodes de chiffrement

- **La cryptologie**

- **Cryptographie**

*Science permettant de créer des systèmes de chiffrement*

- **Système de Chiffrement- Définition**

*Opération de chiffrement qui transforme un texte en clair en un texte chiffré, appelé **cryptogramme**, au moyen d'une clé (qu'on dénomme la **clé de chiffrement**)*

- **Cryptanalyse - Définition**

*Science complémentaire qui consiste à déterminer certaines propriétés d'un système cryptographique dans le but de reconstituer le texte en clair, souvent en l'absence des paramètres qui sont nécessaires pour le déchiffrement*



- Cryptologie – les origines

Secret: **111**

Système de chiffrement : +

Clé: **58**

A



cryptogramme: **169**

B



**111 + 58**

**169 - 58**

- Cryptologie – les origines

Chiffrement Hébraïque: Atbash

→ Décalage de l'alphabet

Ancien testament ou la tanakh



<b>En clair</b>	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
<b>Chiffré</b>	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A

Cryptogramme= ROLEVXIBKGL

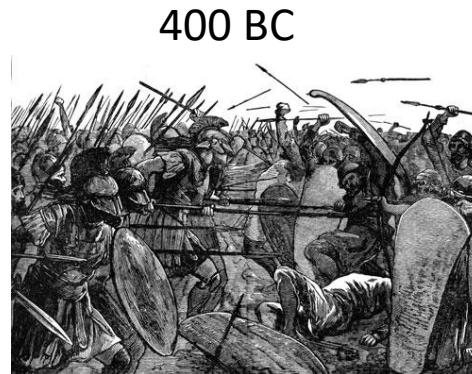
Résultat?

Chiffrement par  
Substitution monoalphabetique

- Cryptologie – les origines

Chiffrement Spartiate:

→ Utilisation d'un rondin de bois pour déchiffrer



## • Cryptologie – les origines

Chiffrement de césar:

2 AC

→ Remplacer chaque lettre de l'alphabet par celle située trois places plus loin dans l'ordre alphabétique



En Clair	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Chiffré	D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

S E C U R I T Y F O R D U M M I E S  
V H F X U L W B I R U G X P P L H V

- Cryptologie – les origines

Chiffrement de Blaise de Vigenere pour Henri VIII:

1500 AC

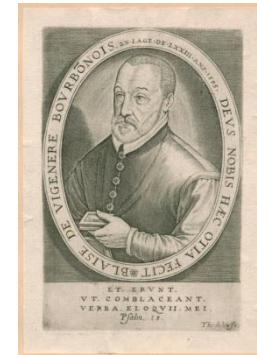
→ Basé sur chiffrement de césar , utilisation de clé et de décalage de 27 positions dans l'alphabet

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Message=NOUS...

Clé=ETESTLA

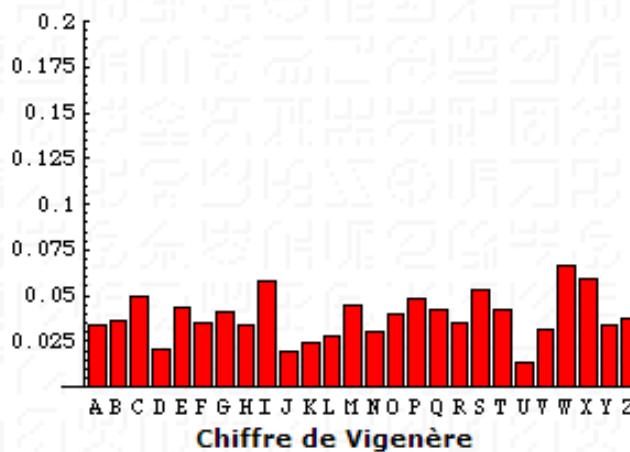
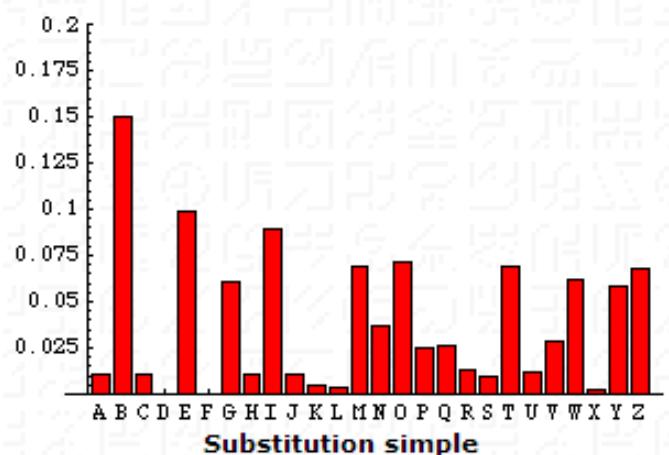
Chiffrer=RHYKG PSSFQ WLAAW LIMED



- Cryptologie – les origines

Chiffrement de Blaise de Vigenere pour Henri VIII:

1500 AC

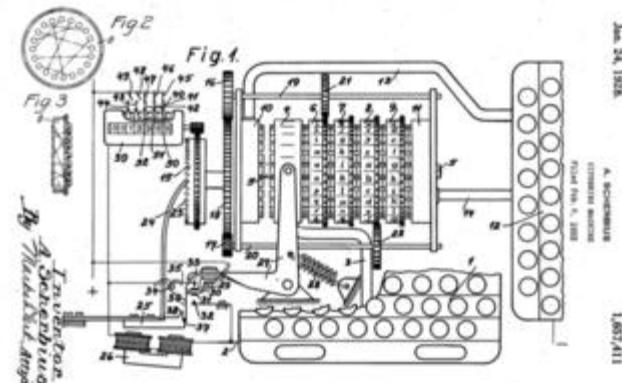
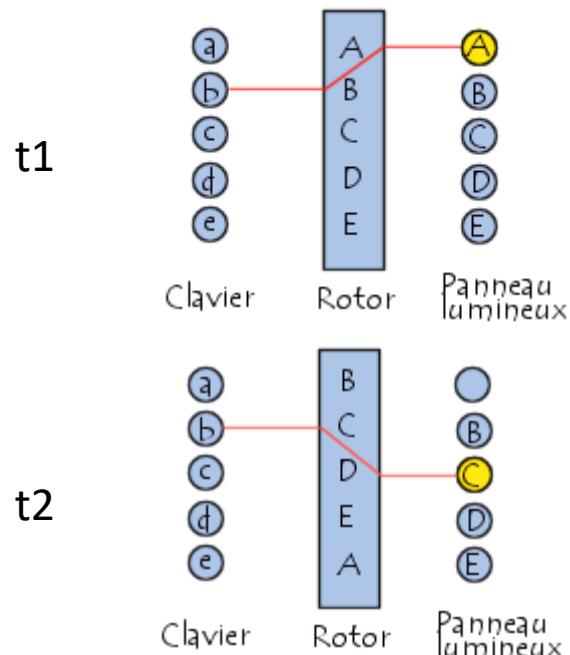


- Cryptologie – les origines

The enigma machine

1939

Exemple avec 1 ROTOR



## Introduction et définition

- 
- Historique
  - Définitions et concepts
  - Type de chiffrement
  - Méthodes de chiffrement

## • La cryptologie: les concepts

### □ L'algorithme

*Ensemble des règles décrivant comment un message est chiffré et déchiffré.*

- La plupart des algorithmes de chiffrement ne sont pas secrets.
- La partie secrète de la plupart des algorithmes de chiffrement est la clé.

### □ La clé- Définition

*Clé ou cryptovariable peut être vue comme une valeur comprenant une grande séquence de bits aléatoires.*

- Plus l'espace des possibles de la clé est grande
  - Plus les valeurs des clés ont un caractère aléatoire
- plus la difficulté est grande pour un attaquant de trouver le secret



## • Cryptologie – les concepts

# Algorithme secret

## VS Algorithme public

## Principe de Kerckhoff (1883)

Plus un algorithme des testés, utilisés, plus le nombre de vulnérabilité découvert sera grand

## • La cryptologie: les concepts

### □ Puissance d'un algorithme de chiffrement

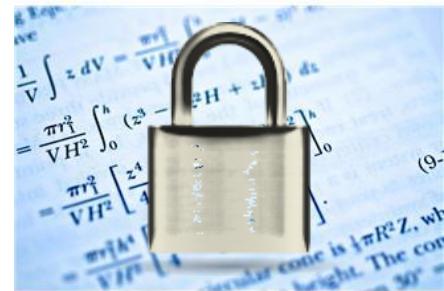
#### □ Dépend de:

- La méthode de chiffrement
- La taille de la clé
- Les vecteurs d'initialisation
- La faculté de tous ces éléments à travailler ensemble

#### □ Est liée à

- À La puissance
- Aux ressources

→ Nécessaires pour casser le système de chiffrement



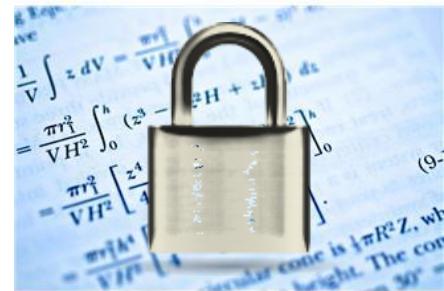
- **La cryptologie: les concepts**

- ❑ **One-Time Pad: la pierre philosophale**

- ❑ Chiffrement parfait, considéré comme incassable
    - ❑ Gilbert Vernam 1917 (chiffrement vernam)
    - ❑ Algorithme de chiffrement XOR (ou exclusif)

- ❑ Pourquoi incassable?

- La clé (pad) ne doit être utilisée qu'une seule fois
    - La clé (pad) doit être aussi longue que le message
    - La clé (pad) doit être distribuée de façon sécurisée avec le destinataire



- Cryptologie – One-time pad

Les dodos n'ont jamais froid, la fin du monde ne passera pas !

```
010111000111010100
101110101010000101
010111011010011010
011001001
```

A	B	R
0	0	0
0	1	1
1	0	1
1	1	0



```
110010001100111110
010001001111100010
001100111110010001
001111010
```

Chiffrement



```
110101001011101010
111111100101110110
100110100110010010
101110001
```



Les dodos n'ont jamais froid, la fin du monde ne passera pas !

```
010111000111010100
101110101010000101
010111011010011010
011001001
```



Déchiffrement

```
110010001100111110
010001001111100010
001100111110010001
001111010
```



- Cryptologie – One-time pad



- La cryptologie: les concepts

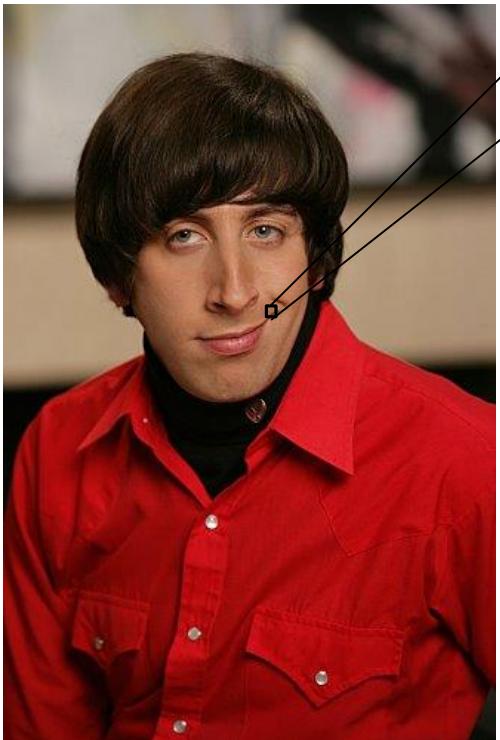
- Stéganographie

*Dissimuler un message dans un autre message*

- Démocrite, ancien roi de Sparte 485 BC
      - « il prit une tablette double, en gratta la cire, puis écrivit sur le bois même les projets de Xerxès ; ensuite il recouvrit de cire son message : ainsi le porteur d'une tablette vierge ne risquait pas d'ennuis. »



## • Cryptologie – Stéganographie



10101010 10101010

Information codé sur 16 bits  
Poids fort à gauche

10101010 1011011

Remplacer l'info de  
poids faible par  
message secret

11101000 1010011

10000010 1001111

01111010 1010001

Reconstitution du  
message secret

- La cryptologie: Pourquoi ?

- Assurer les services de sécurité suivants:

- Confidentialité
  - Intégrité
  - Authentification
  - Autorisation
  - Non répudiation



- **La cryptologie: Pourquoi ?**

- Assurer les services de sécurité suivants:

- Confidentialité

*Empêcher toutes divulgations d'information à des personnes, programmes ou équipements non autorisés*

- Intégrité

*Assurer que les informations stockées, transmises et reçues n'ont pas été modifiées par une entité non autorisée. Toutes modifications d'information entraîne un viol d'intégrité et doit être détecté.*



- **La cryptologie: Pourquoi ?**

- **Assurer les services de sécurité suivants:**

- **Authentification**

*Vérifier l'authenticité de l'identité d'une entité (what you know, what you have, what you are).*



- **Autorisation**

*Attribution de droits, autorisation en accord avec la politique de sécurité en vigueur.*

- **Non répudiation**

*Imputabilité d'un message, action , activité sur le système d'information.*

## Introduction et définition

- 
- Historique
  - Définitions et concepts
  - Type de chiffrement
  - Méthodes de chiffrement

## • La cryptologie: Types de chiffrement

### □ Substitution

*La substitution remplace des bits, des caractères ou des blocs de caractères avec d'autres bits, caractères ou blocs de caractères*

- Effet d'une substitution = **confusion**

### □ Transposition

*La transposition ne remplace pas les informations d'un message, mais déplace les informations (bits, caractères, blocs de caractères) du message original dans ce dernier*

- Effet d'une substitution = **diffusion**

### □ Transposition et transposition simples sont sensibles à l'analyse fréquentielle

### □ Les techniques actuelles utilisent à la fois la substitution et la transposition



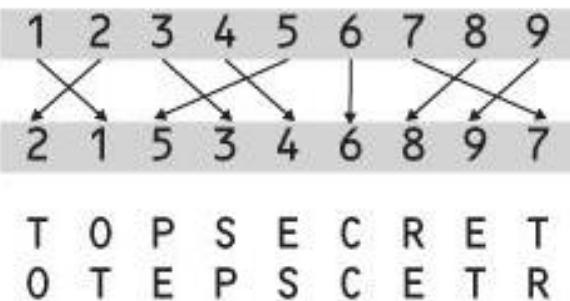
- La cryptologie: Types de chiffrement

## Substitution Cipher



GRAY FOX HAS ARRIVED  
UKQN YGB IQL QKKOCTR

## Transposition Cipher



- **La cryptologie: Chiffrement par blocs ou par flux**

- **Chiffrement par bloc**

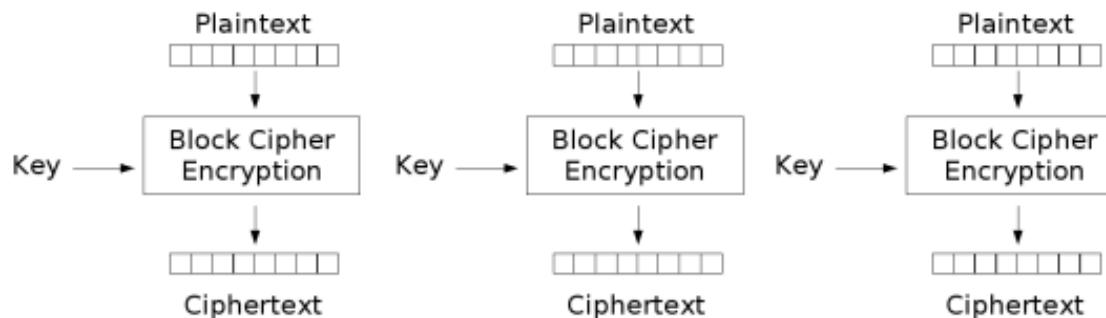
*Le chiffrement par bloc, utilisé pour le chiffrement et le déchiffrement, divise le message en blocs de bits puis chiffre / déchiffre ces blocs les uns après les autres*

- **Chiffrement par flux**

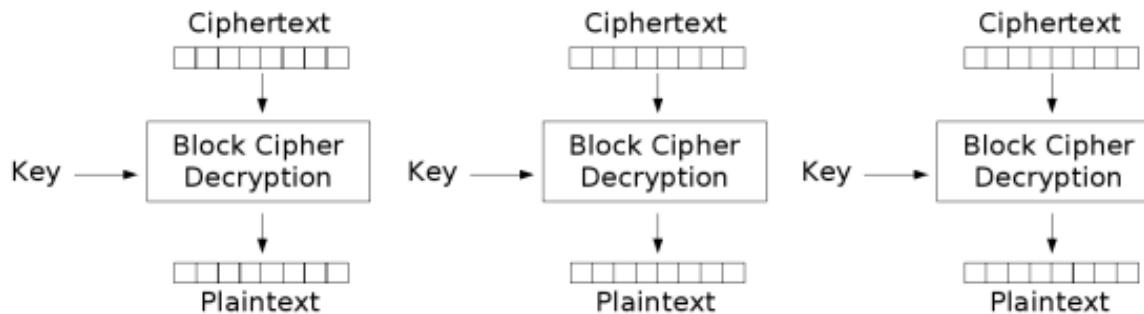
*Le chiffrement traite le message comme un flux et chaque bit du message original est chiffré (fonction mathématique)*



- La cryptologie: Chiffrement par blocs



Electronic Codebook (ECB) mode encryption



Electronic Codebook (ECB) mode decryption

## • La cryptologie: Chiffrement par blocs

### □ Chiffrement par bloc

- Chaque bloc est chiffré indépendamment
- Notation  $C=E(P,K)$
- Pour un ensemble de message  $P_0, P_1, P_m$

Chiffrement

$$C_0 = E(P_0, K)$$

$$C_1 = E(P_1, K)$$

$$C_2 = E(P_2, K)$$

Déchiffrement

$$P_0 = D(C_0, K)$$

$$P_1 = D(C_1, K)$$

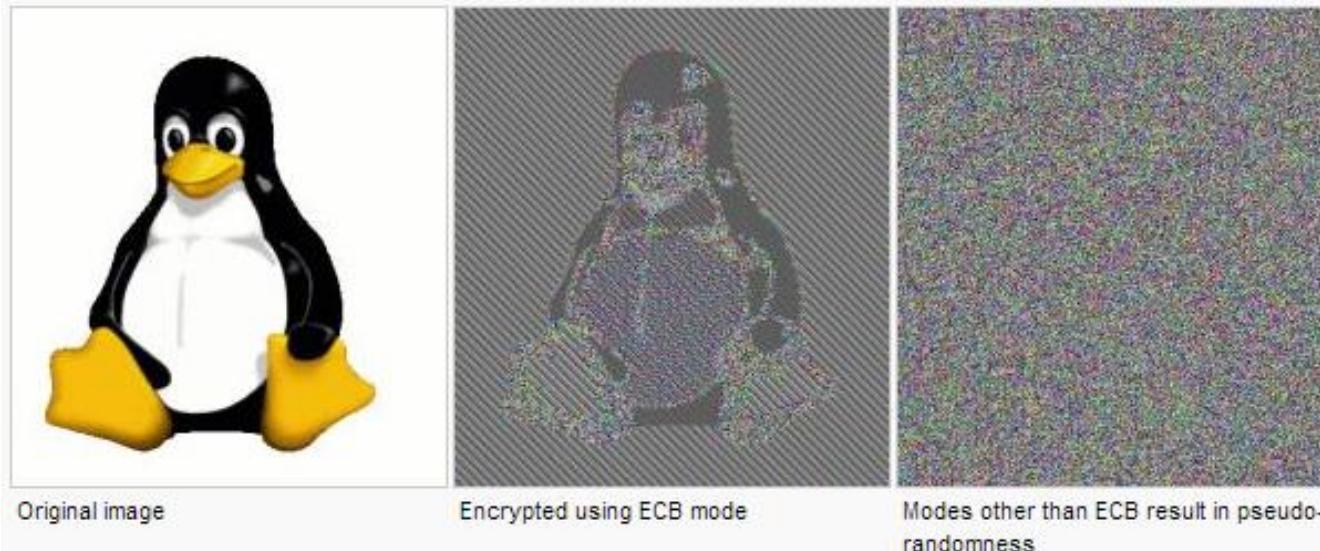
$$P_2 = D(C_2, K)$$



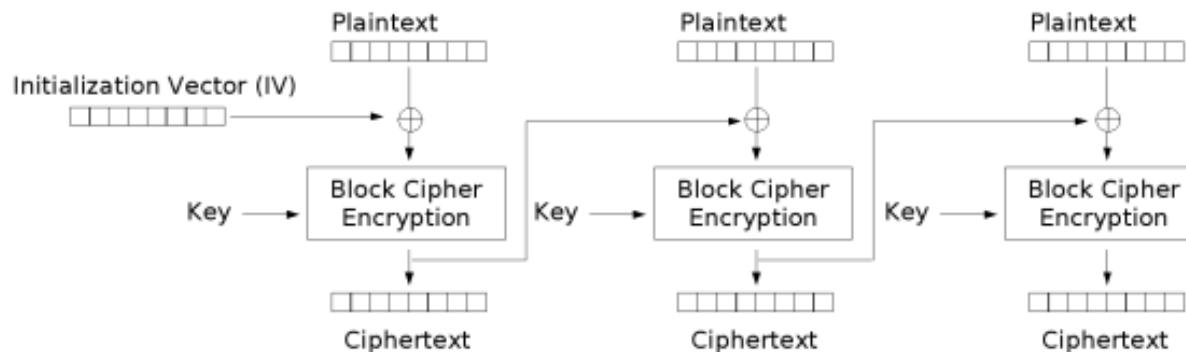
Mêmes blocs de messages sont chiffrés de la même façon

→ Divulgation d'information perte de confidentialité

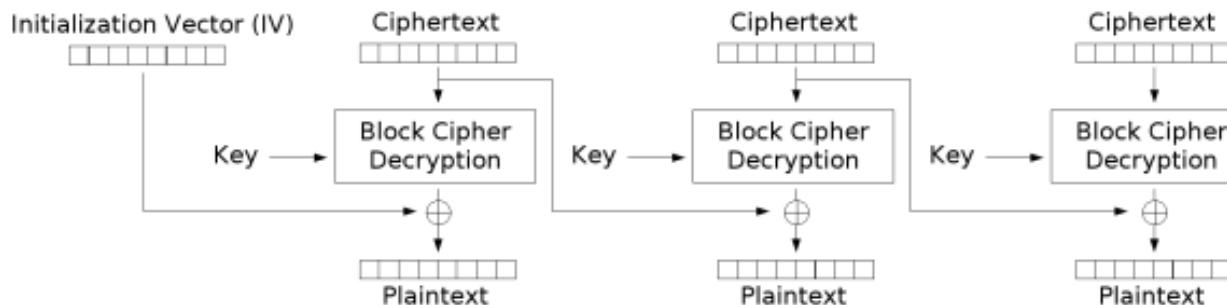
- La cryptologie: Chiffrement par blocs



- La cryptologie: Chiffrement par blocs



Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

## • La cryptologie: Chiffrement par blocs

### □ Chiffrement par bloc

- Blocs sont chainés entre eux
- Utilisation d'un vecteur d'initialisation (VI) pour initialiser
- VI aléatoire mais pas nécessairement secret

Chiffrement

$$C_0 = E(IV \oplus P_0, K)$$

$$C_1 = E(C_0 \oplus P_1, K)$$

$$C_2 = E(C_1 \oplus P_2, K)$$

Déchiffrement

$$P_0 = VI \oplus D(C_0, K)$$

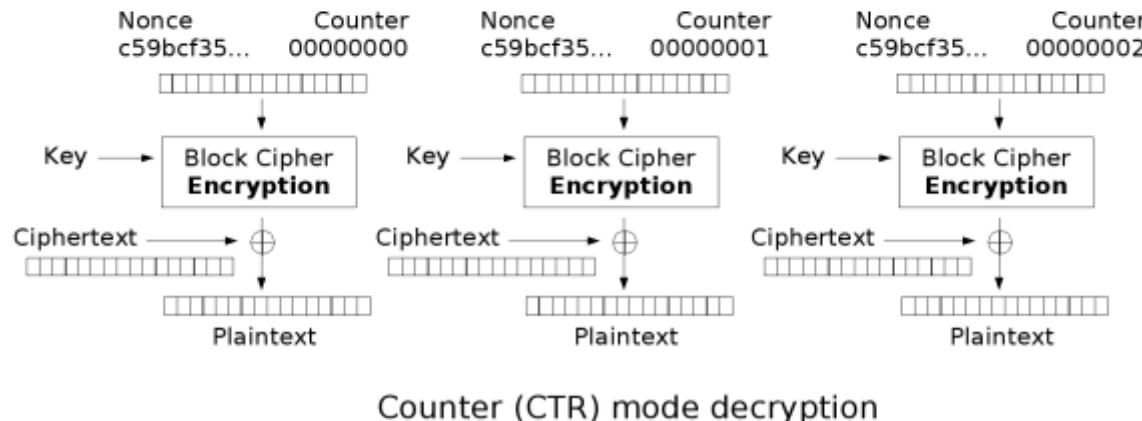
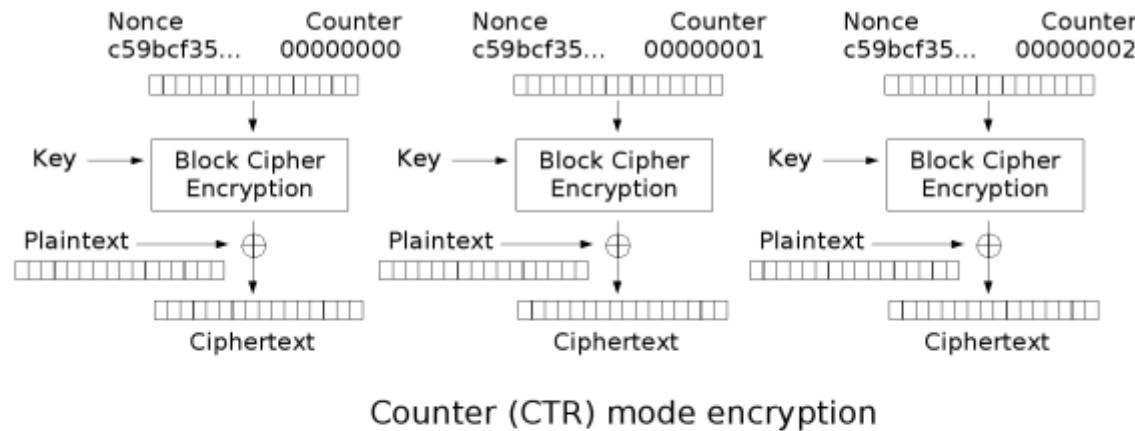
$$P_1 = C_0 \oplus D(C_1, K)$$

$$P_2 = C_1 \oplus D(C_2, K)$$



- Chiffrement séquentiel → lenteur
- Découpage du message en multiple de la taille des blocs chiffrés

- La cryptologie: Chiffrement par blocs



## • La cryptologie: Chiffrement par blocs

### □ Chiffrement par bloc

- Utilise le chiffrement par bloc comme un chiffrement par flux
- Peut être utilisé pour des accès aléatoires

Chiffrement

$$C_0 = P_0 \oplus E(VI, K)$$

$$C_1 = P_1 \oplus E(VI+1, K)$$

$$C_2 = P_2 \oplus E(VI+2, K)$$

Déchiffrement

$$P_0 = C_0 \oplus E(VI, K)$$

$$P_1 = C_1 \oplus E(VI+1, K)$$

$$P_2 = C_2 \oplus E(VI+2, K)$$



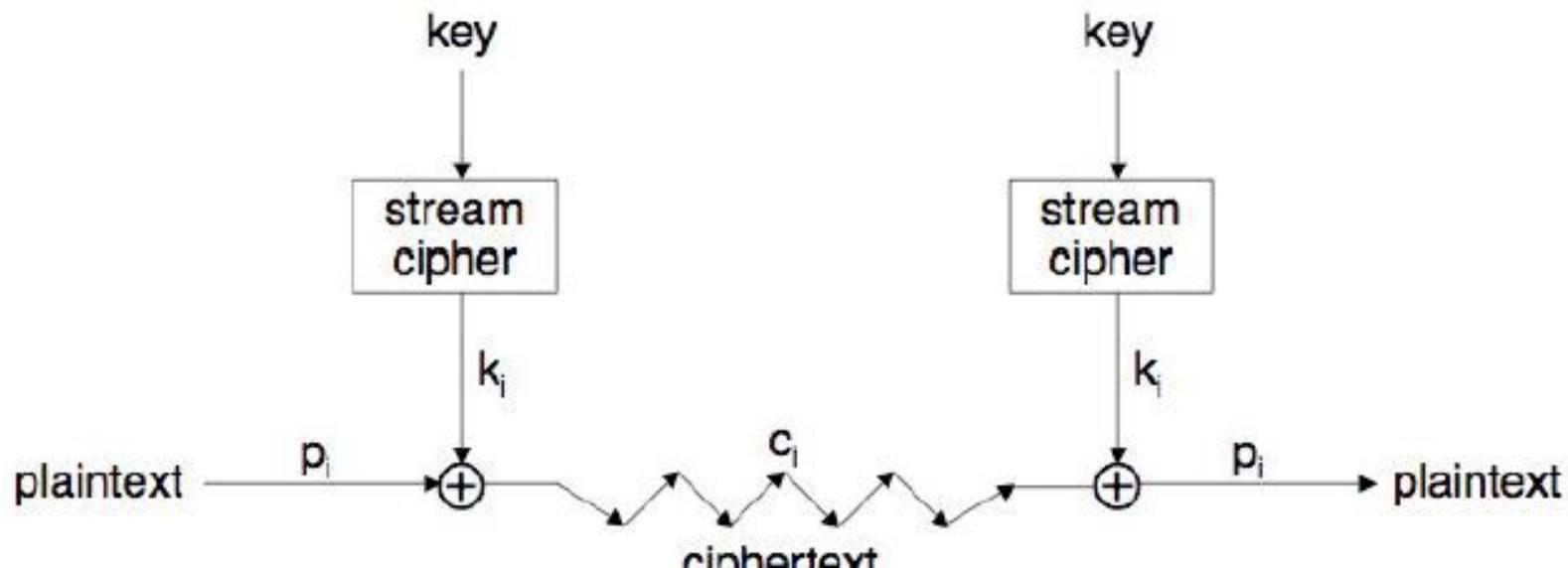
- Chiffrement en parallèle possible

## • La cryptologie: Chiffrement par Flux

- ❑ Généralisation de l'idée du one-time pad
- ❑ Initialisé avec une clé courte
- ❑ Clé est transformée en un keystream
- ❑ XOR pour le chiffrement et le déchiffrement



- La cryptologie: Chiffrement par Flux



## • La cryptologie: Chiffrement par Flux

- ❑ Décalage de registre
  - ❑ Chiffrement par flux largement basé sur le décalage de registre
  - ❑ Contient une boucle de rétroaction (feedback)
  - ❑ Utilisation de fonction de rétroaction linéaire ou non (Linear Feedback Shift register)



- La cryptologie: Chiffrement par Flux: Exemple RC4

**Clef secrète K**, composée de k mots de n bits,  $K[0], \dots, K[k-1]$ .

T tableau temporaire

S tableau de valeurs

$|K|$  taille du vecteur K

**Initialisation.**

Pour i de 0 à 255,

$$S[i] \leftarrow i$$

$$T[i] = K[i \bmod (|K|)]$$

$j=0$

Pour i = 0 à 255 faire

$$j \leftarrow (j + S[i] + T[i]) \bmod 256$$

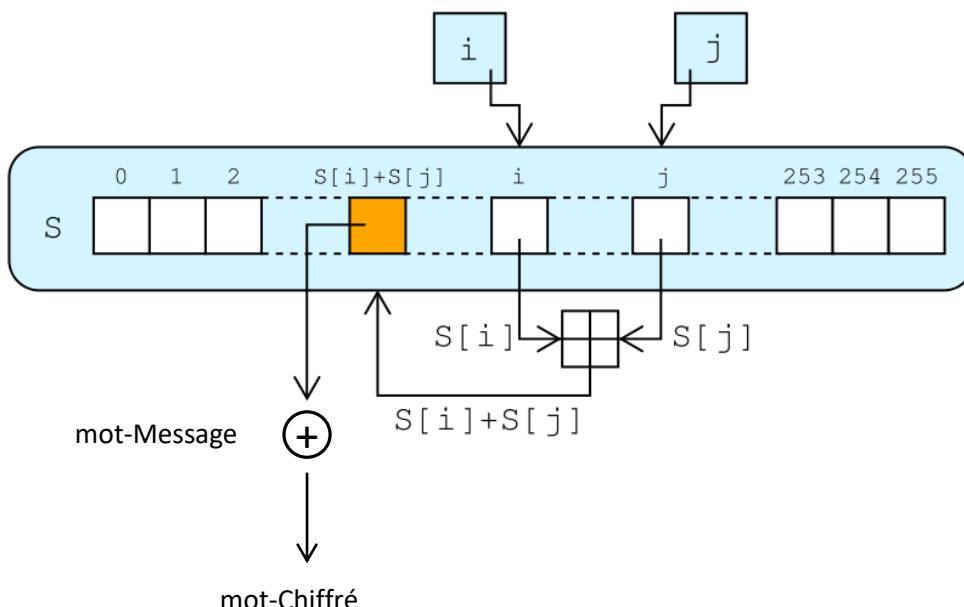
échanger  $S[i]$  et  $S[j]$

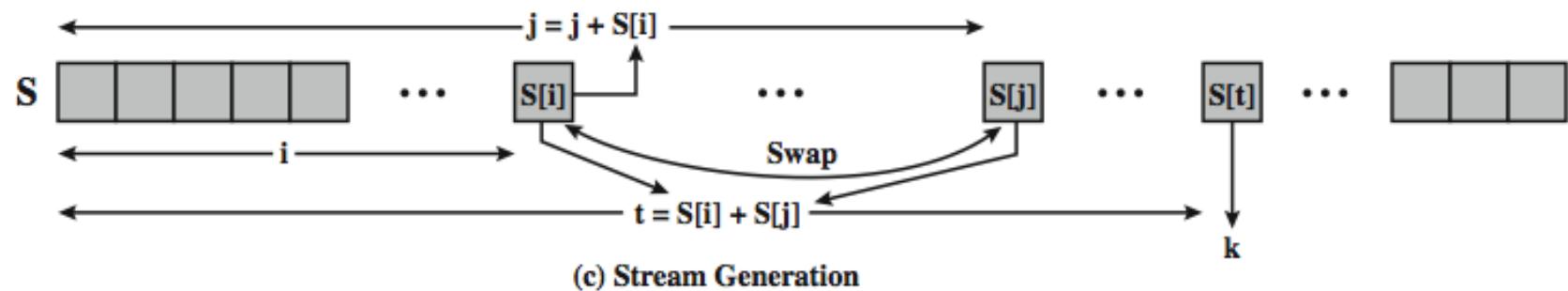
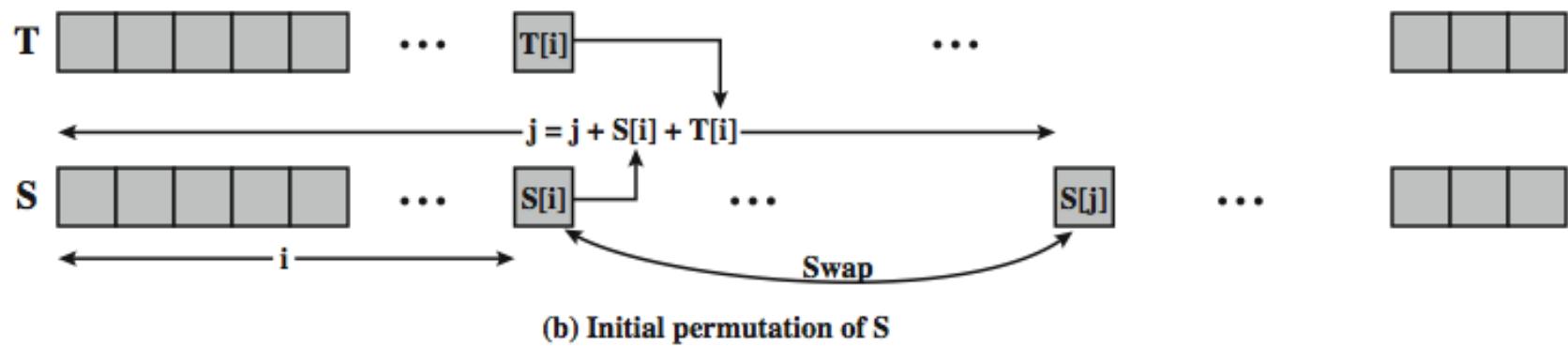
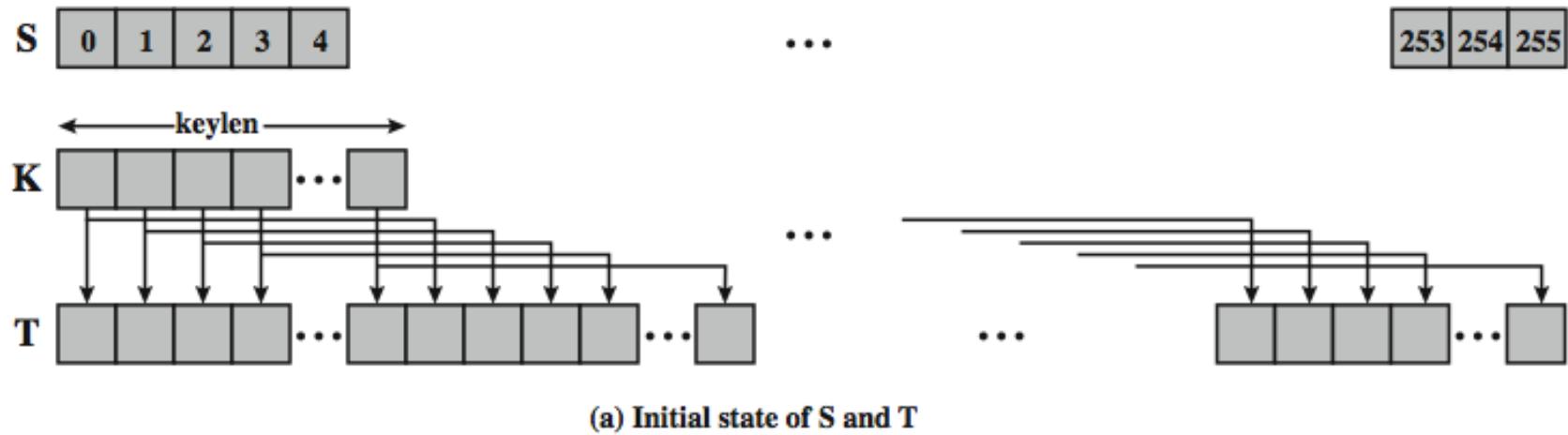
**Génération de la suite chiffrante.**

$$i = j = 0$$

• Répéter

- $i \leftarrow (i+1) \bmod 256$
- $j \leftarrow (j+S[i]) \bmod 256$
- échanger  $S[i]$  et  $S[j]$ .
- Retourner  $S[S[i] + S[j]]$  (sous clé)





## • La cryptologie: Chiffrement par Flux

### ❑ Avantages

- ❑ Très rapide
- ❑ Adapté aux applications temps réelles

### ❑ Inconvénients

- ❑ Propagation d'erreurs (problème de synchronisation)
- ❑ Sécurité difficile à atteindre (pas de preuve)



## Introduction et définition

- 
- Historique
  - Définitions et concepts
  - Type de chiffrement
  - Méthodes de chiffrement

- **La cryptologie: Méthodes de chiffrement**

- Symétrique

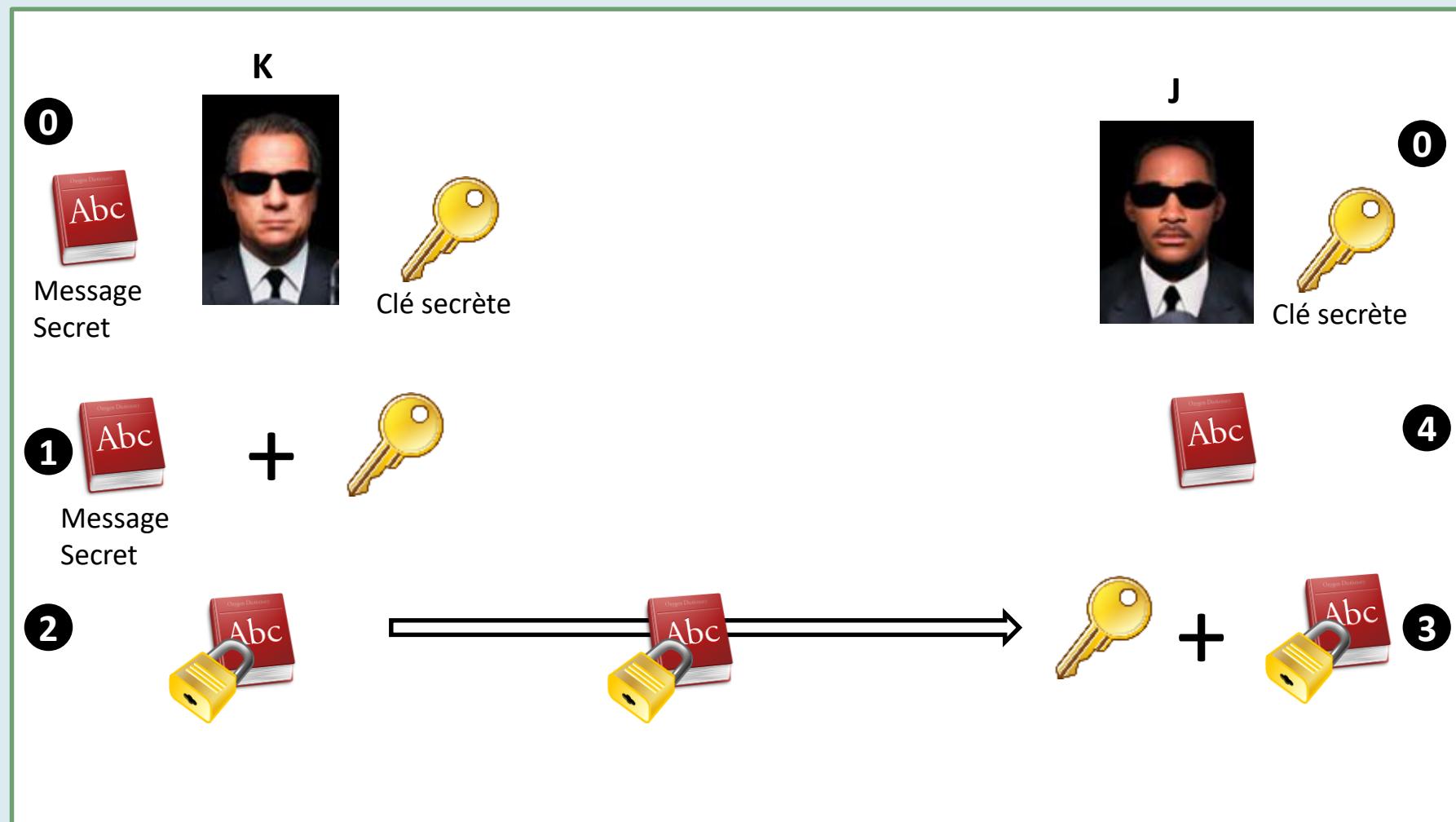
- Secret partagé (clé symétrique)

- Asymétrique

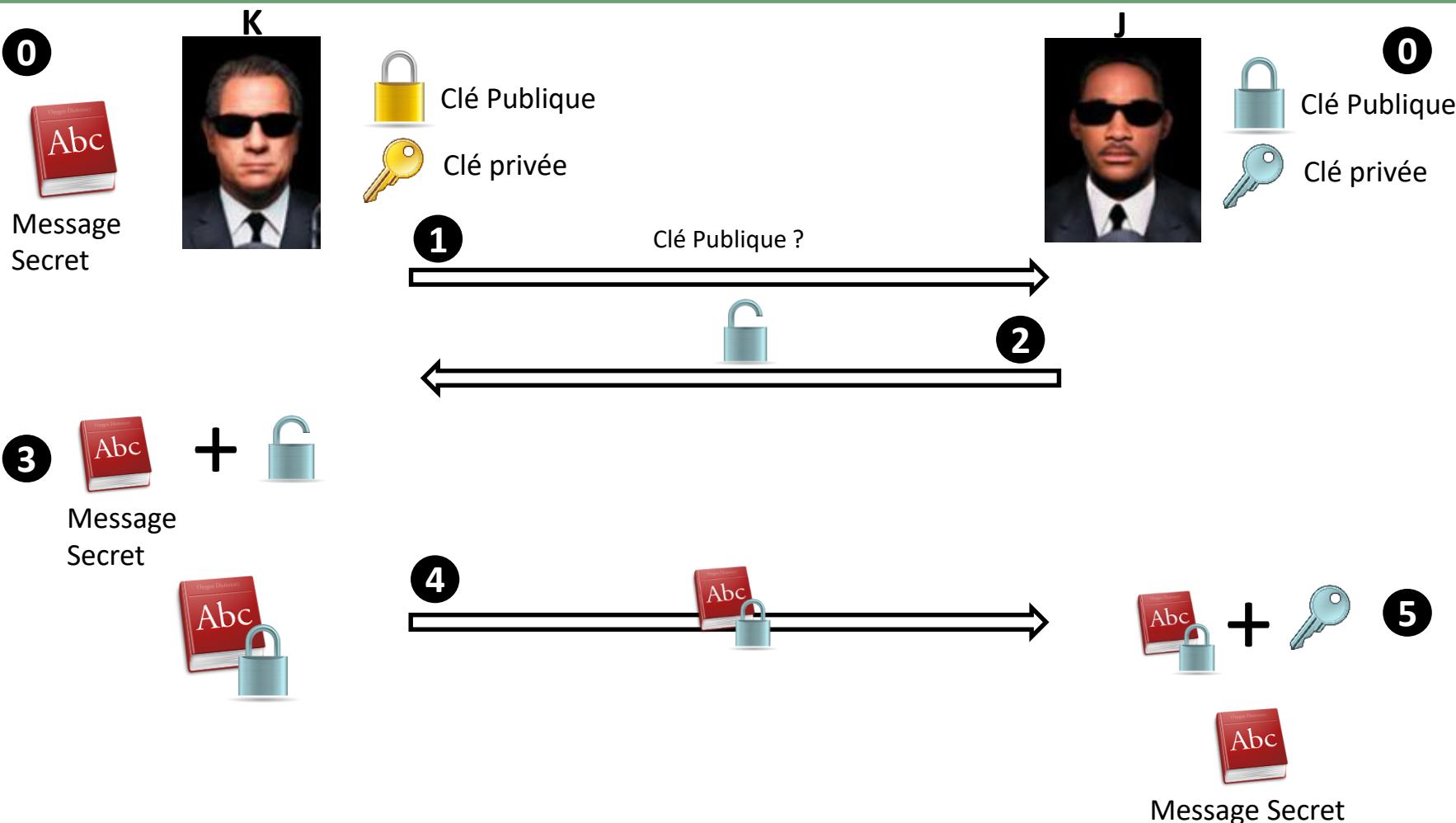
- utilisation de clé publique et clé privée



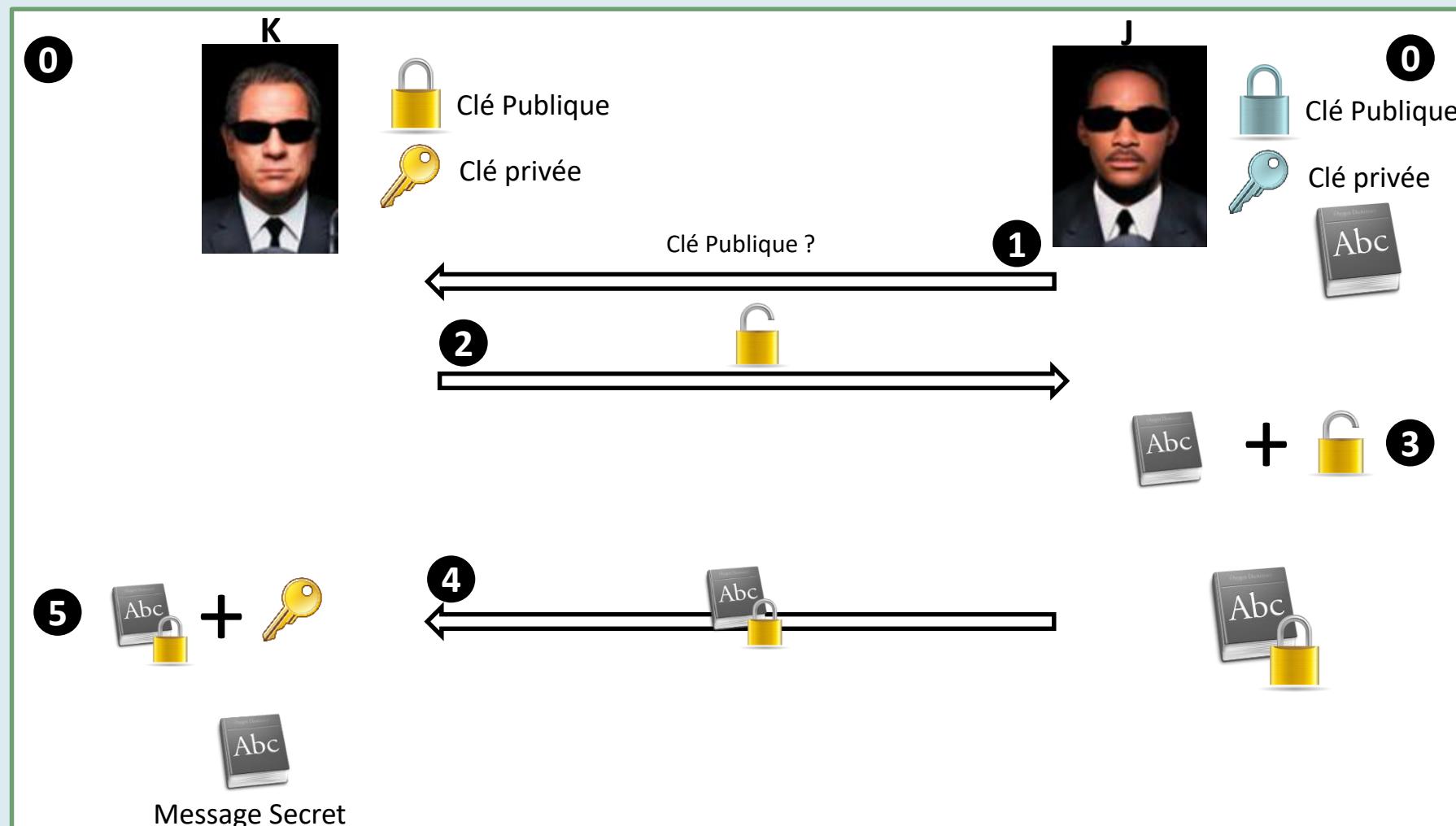
- La cryptologie: Méthodes de chiffrement symétrique



- La cryptologie: Méthodes de chiffrement asymétrique



- La cryptologie: Méthodes de chiffrement asymétrique



- La cryptologie: Méthodes de chiffrement asymétrique A vous de Jouer!

K



J



F



J a besoin de récupérer des informations de K pour les transmettre à F

## • La cryptologie: Méthodes de chiffrement

### □ Symétrique

#### - Avantages

- Plus rapide que les chiffrements asymétriques
- Difficile à casser si grande taille de clé

#### - Inconvénients

- Demande un mécanisme permettant de délivrer les clés
- Chaque pair d'utilisateur à besoin d'une clé unique, problème de management des clés
- Garantit la confidentialité mais pas l'authenticité et la non réputation



## • La cryptologie: Méthodes de chiffrement

### □ Asymétrique

#### - Avantages

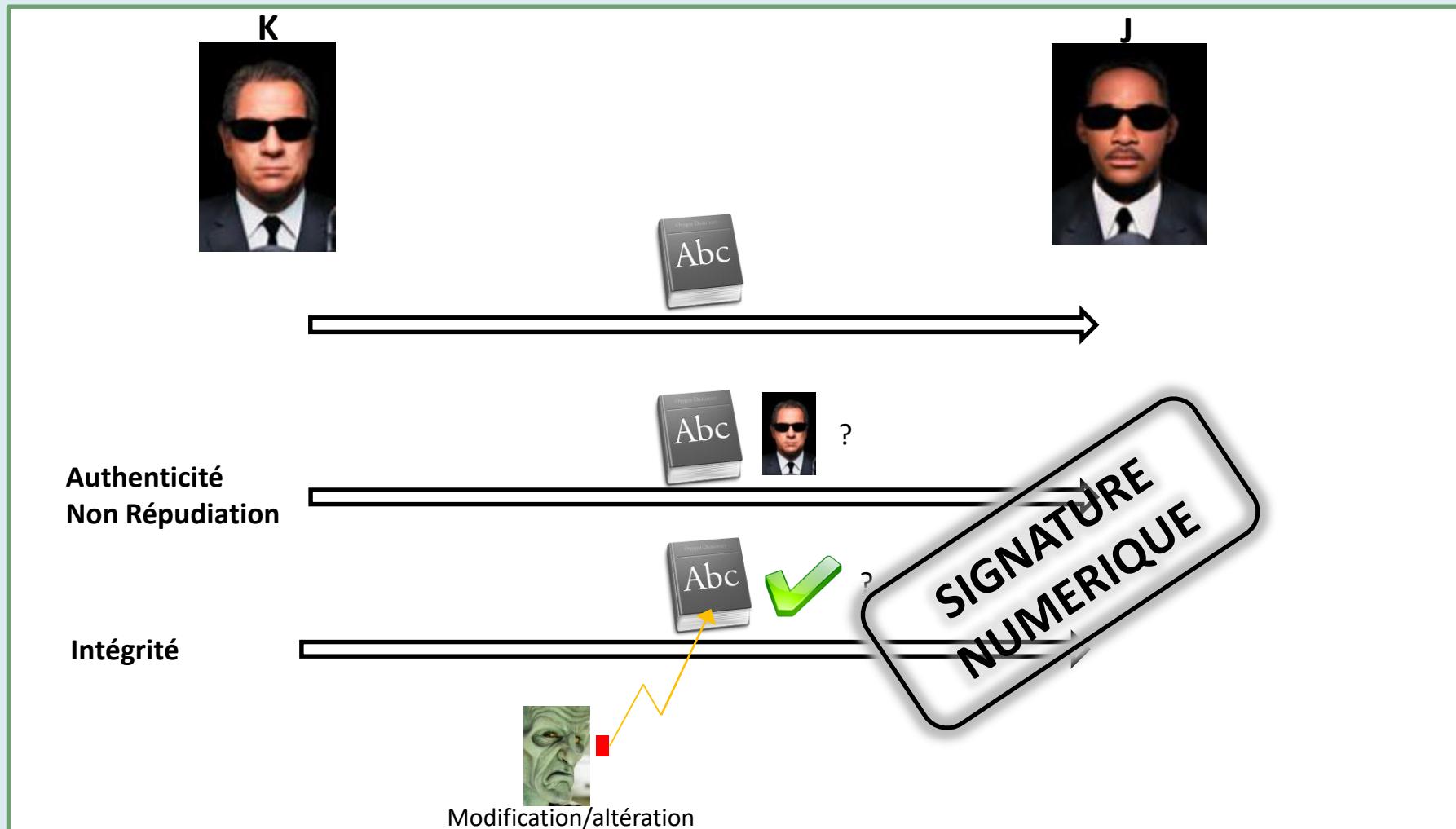
- Distribution des clés plus facile
- Meilleur passage à l'échelle
- Garantit la confidentialité mais aussi l'authenticité et la non répudiation

#### - Inconvénients

- Bien plus lent que le chiffrement symétrique
- Demande beaucoup de ressource (calcul mathématique complexe)



- La cryptologie: Méthodes de chiffrement asymétrique



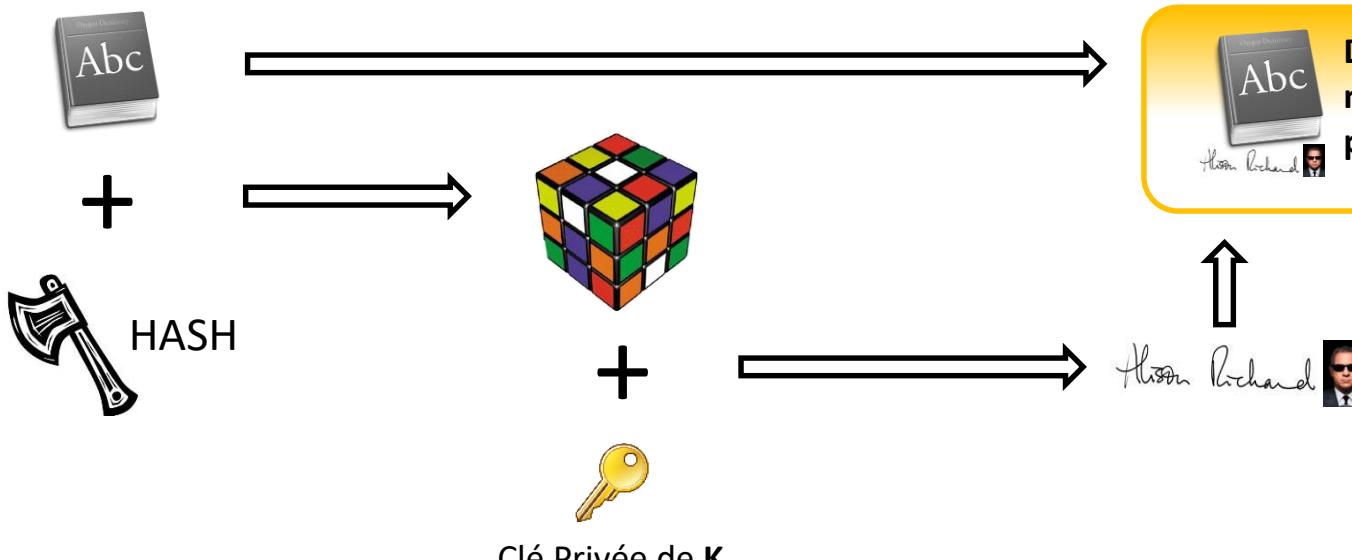
- La cryptologie: Méthodes de chiffrement asymétrique



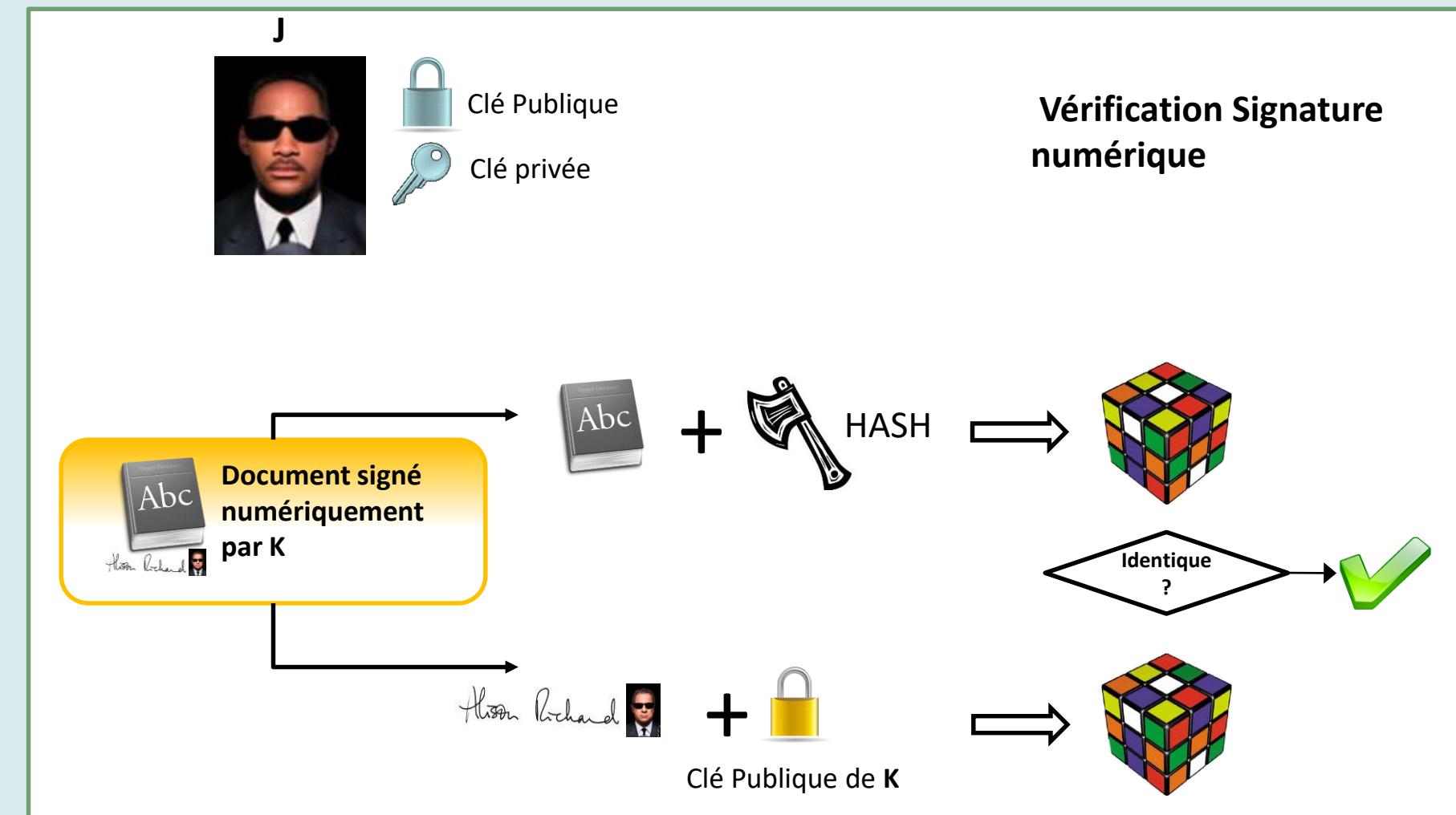
Clé Publique

Clé privée

Signature numérique



- La cryptologie: Méthodes de chiffrement asymétrique



- La cryptologie: Méthodes de chiffrement asymétrique



Je chiffre avec une clé **publique**:

Seule les personnes possédant la clé privée associée peuvent lire le message

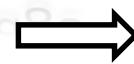


Confidentialité



Je chiffre avec une clé **privée**:

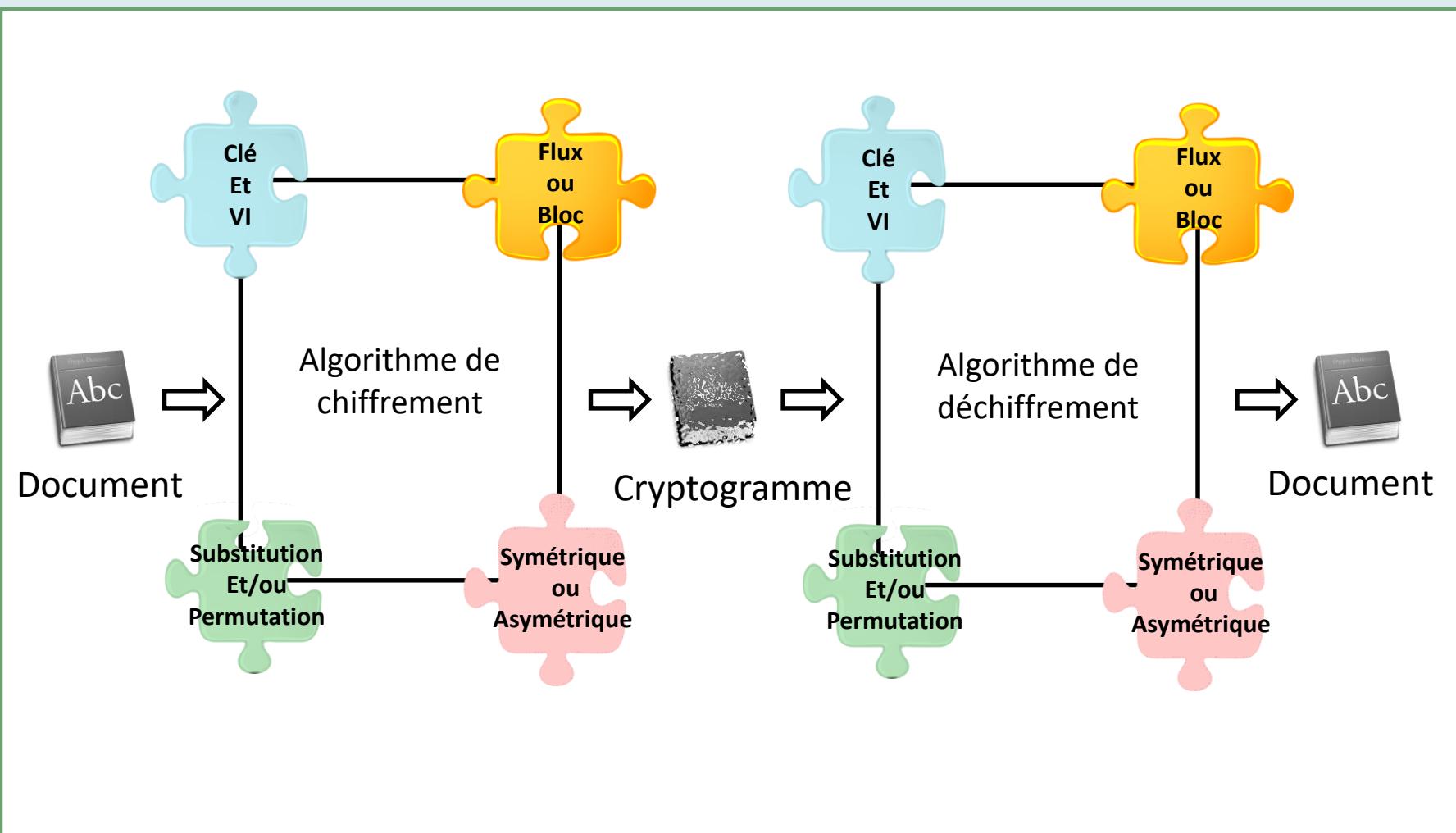
Toutes les personnes possédant la clé publique peuvent lire le message



Authenticité,  
non répudiation

(Signature numérique)

- **Introduction et définition: Bilan**



# Chiffrement Symétrique

- 
- Bilan
  - DES / 3 DES
  - AES

## • Chiffrement symétrique

- Le plus couramment utilisé
- Principale avantage liée à la rapidité et la complexité liée à la taille de la clé
- Utilisation du chiffrement asymétrique pour la distribution de clé (voir partie Chiffrement Hybride)
- Exemples d'algorithmes de chiffrement
  - Data Encryption Standard (DES)
  - 3DES(triple DES)
  - Blowfish
  - Twofish
  - IDEA (Internation Data Encryption Algorithme)
  - RC4,RC5,RC6
  - AES
  - SAFER
  - Serpent



# Chiffrement Symétrique

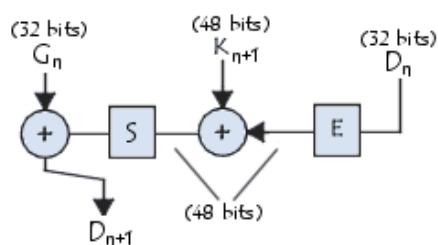
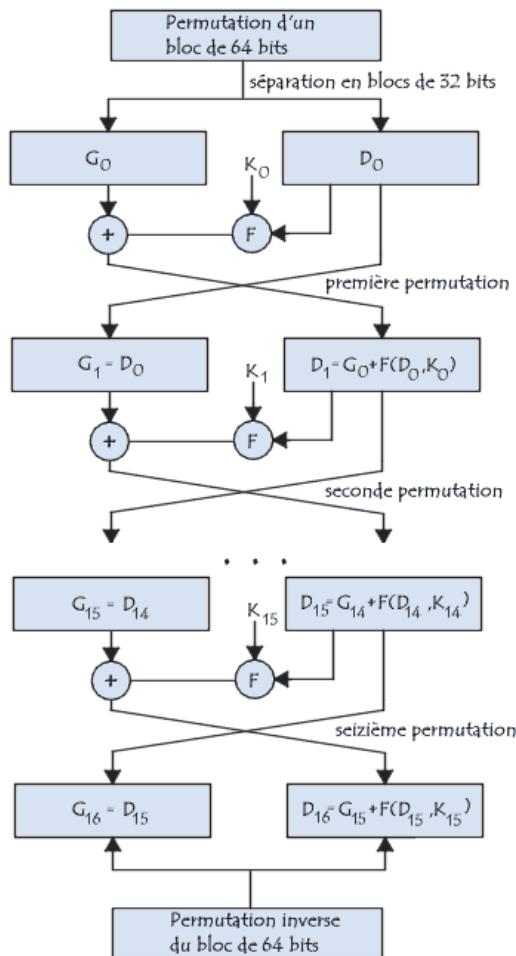
- 
- Bilan
  - DES / 3 DES
  - AES

## • DES-Data Encryption Standard

- ❑ IBM 1977
- ❑ Chiffrement symétrique
- ❑ Chiffrement par blocs (64 bits)
- ❑ Utilisation d'une clé de 64 bits (56 vrai clé 8 parité)
- ❑ Substitution et permutation
  
- ❑ Algorithme
  1. Fractionnement du texte en blocs de 64 bits (8 octets) ;
  2. Permutation initiale des blocs ;
  3. Découpage des blocs en deux parties: gauche et droite, nommées G et D ;
  4. Etapes de permutation et de substitution répétées 16 fois (appelées rondes) ;
  5. Recollement des parties gauche et droite puis permutation initiale inverse

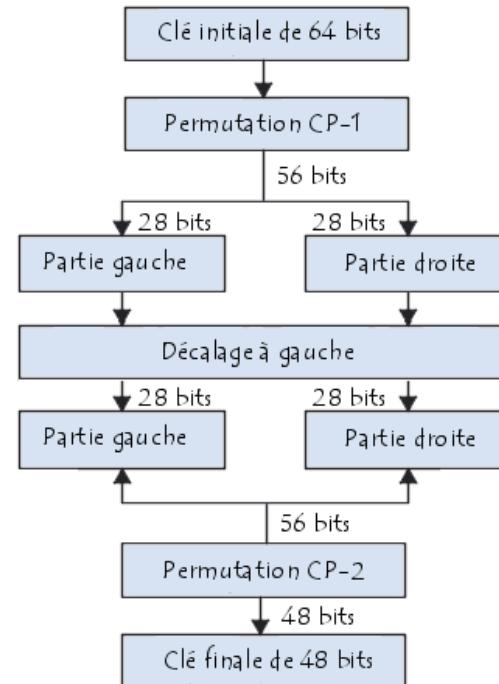


- DES-Data Encryption Standard

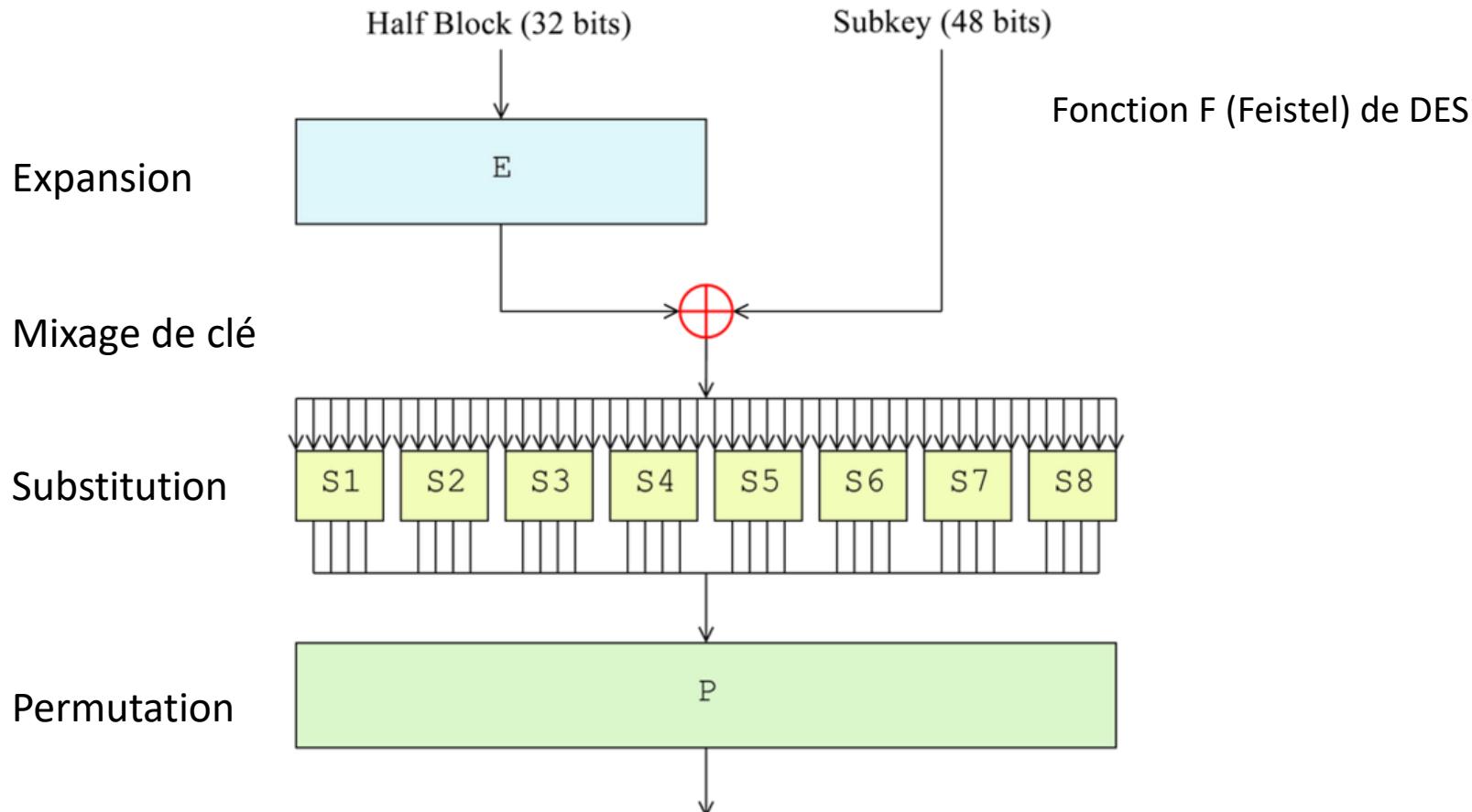


Une ronde

## Génération de clé

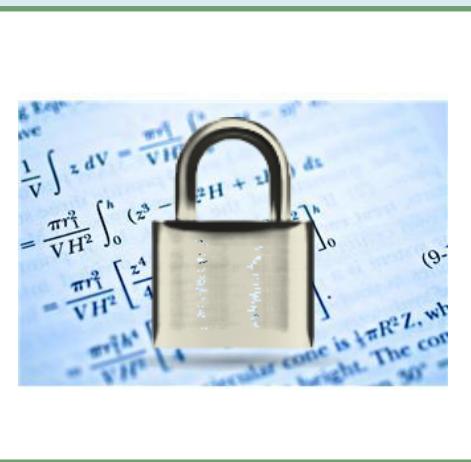


- DES-Data Encryption Standard

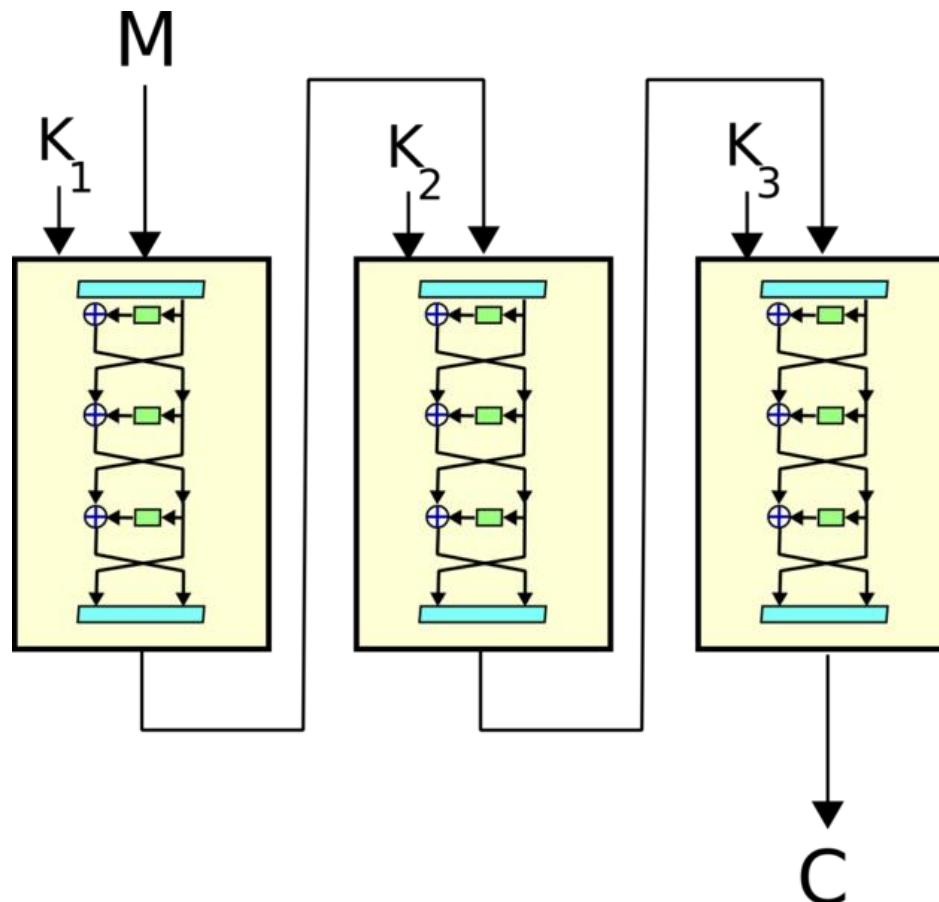


## • 3 DES- Triple Data Encryption Standard

- ASN 1998
- Chiffrement symétrique
- Chiffrement par blocs (64 bits)
- Utilisation d'une clé de 168, 112 ou 56 bits
- Substitution et permutation
- 48 rondes équivalentes DES



- 3 DES- Triple Data Encryption Standard



# Chiffrement Symétrique

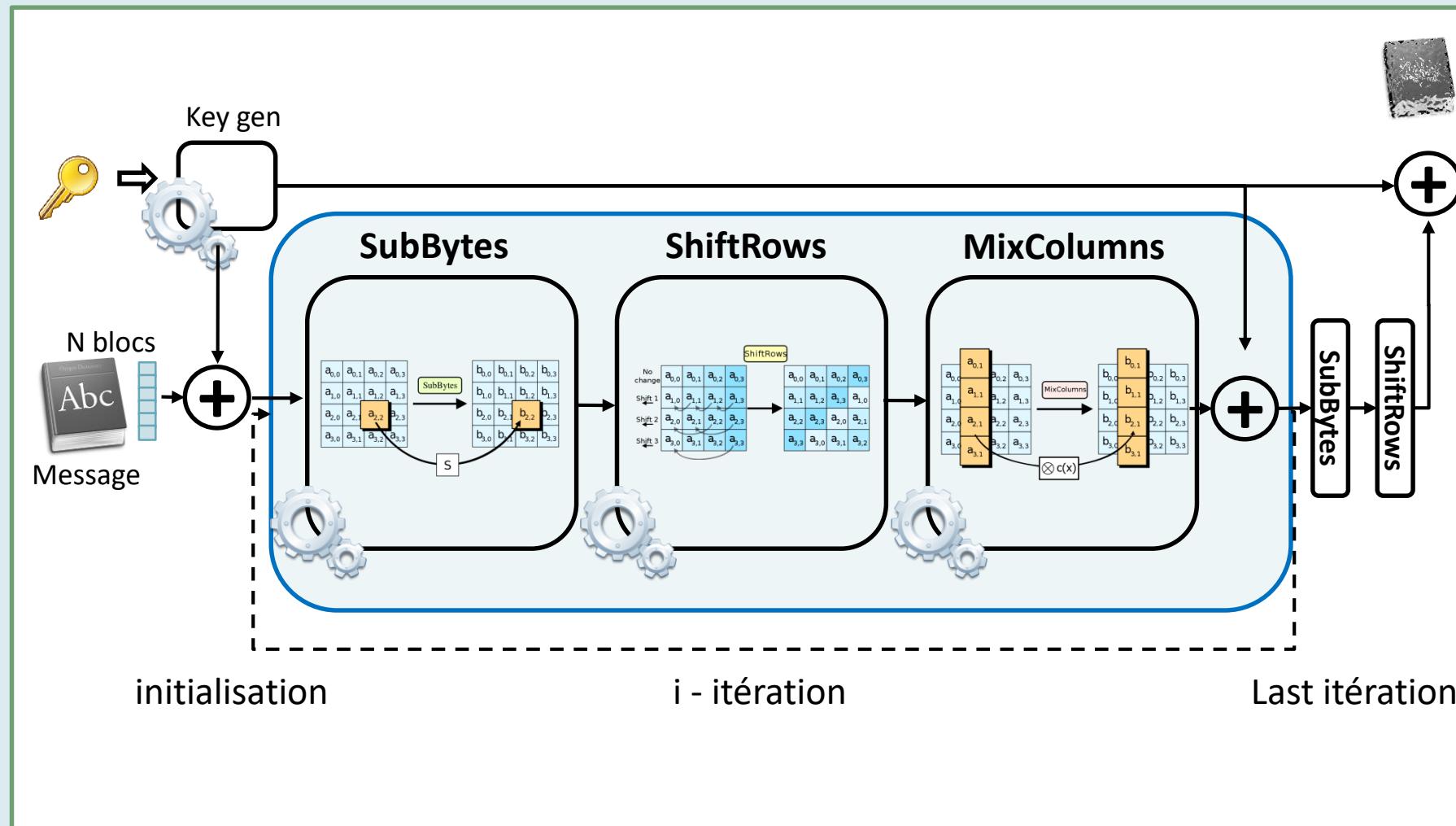
- 
- Bilan
  - DES / 3 DES
  - AES

## • AES- Advanced Encryption Standard

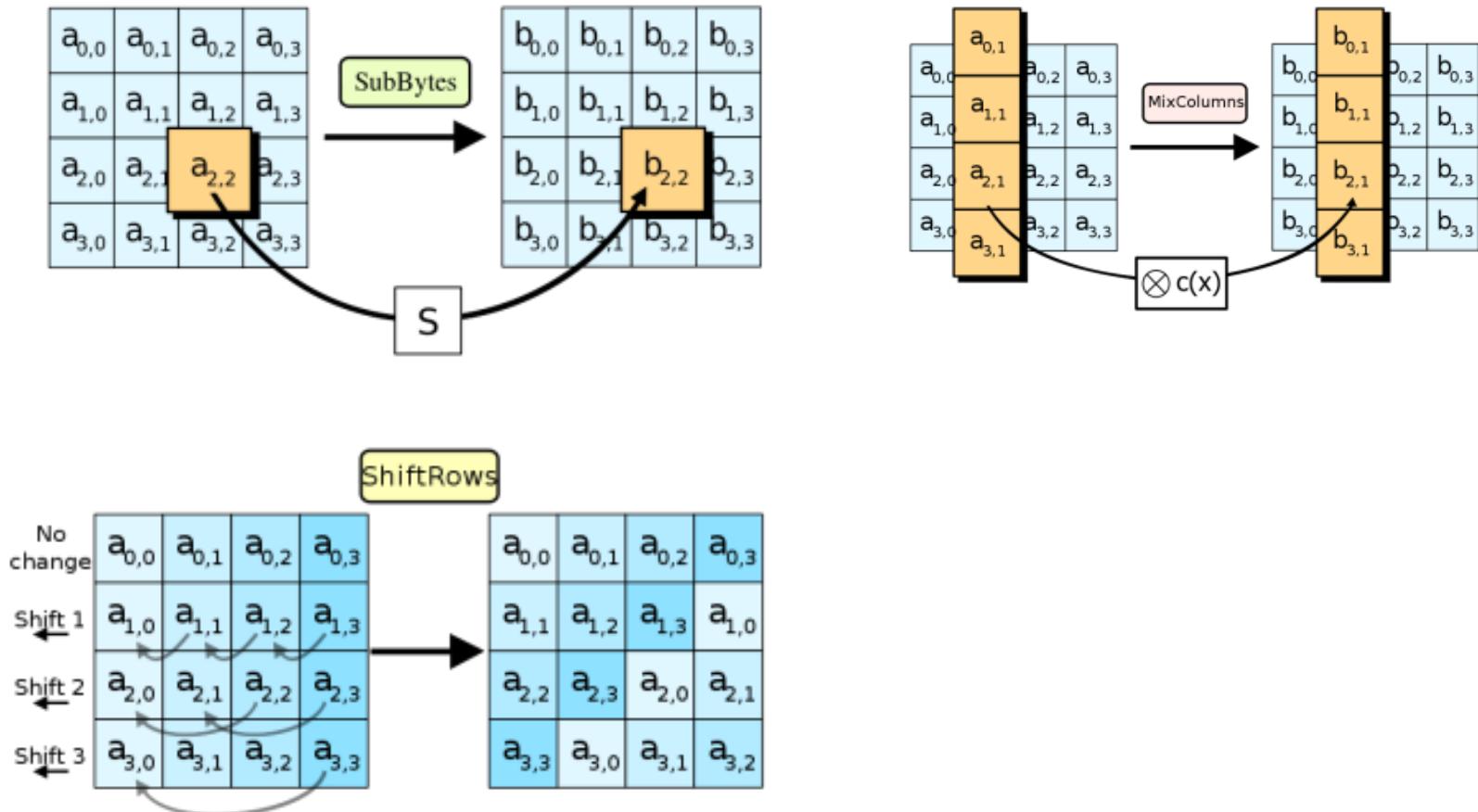
- ❑ AES ou Rijndael 2000 approuvé par la NSA
  - standard Chiffrement US
- ❑ Chiffrement symétrique
- ❑ Chiffrement par blocs (128 bits)
- ❑ Utilisation d'une clé de 128, 192 ou 256 bits
- ❑ Substitution et permutation
- ❑ 10,12 ou 14 rondes selon la taille de la clé



- AES- Advanced Encryption Standard

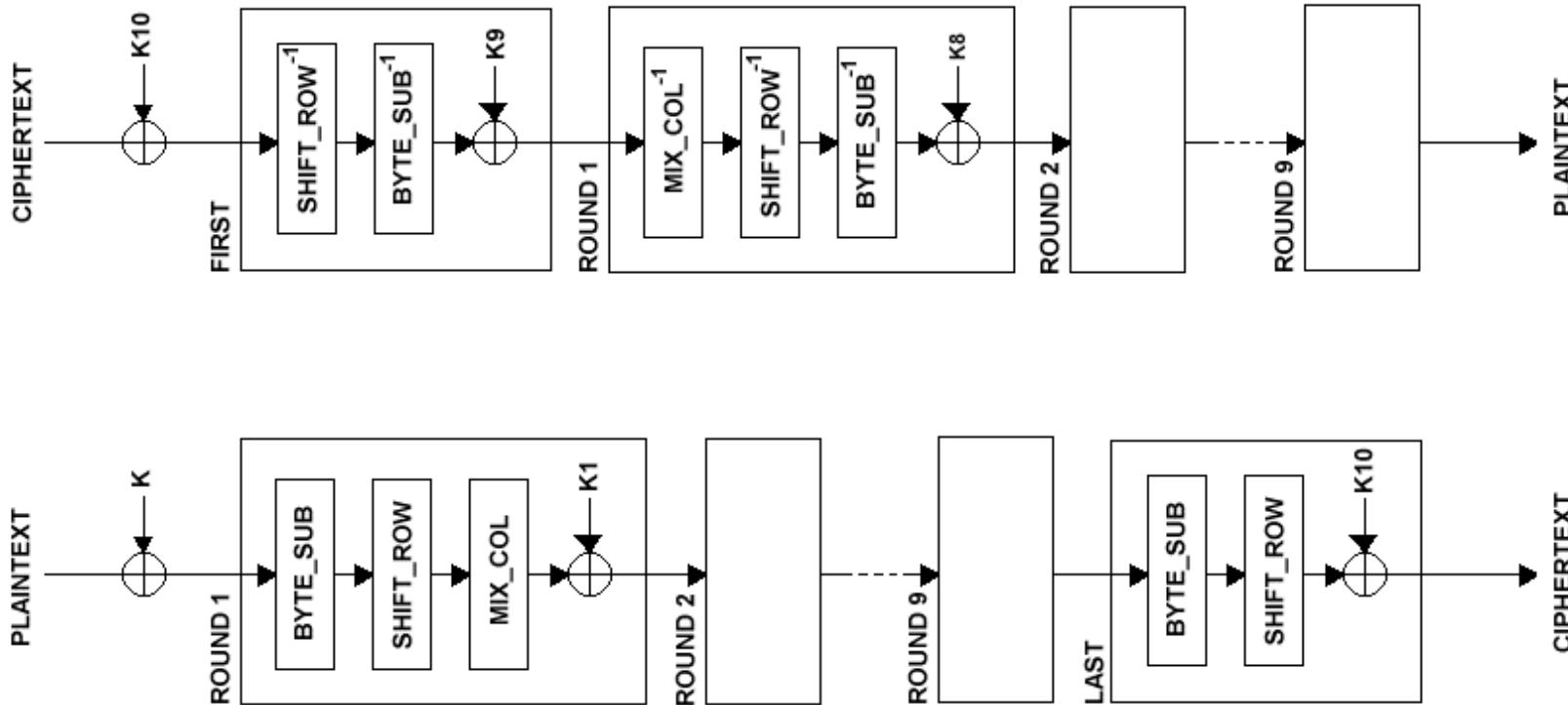


- AES- Advanced Encryption Standard



[http://en.wikipedia.org/wiki/Advanced\\_Encryption\\_Standard](http://en.wikipedia.org/wiki/Advanced_Encryption_Standard)

- AES- Advanced Encryption Standard



## • AES- Advanced Encryption Standard

### □ Bilan

- Difficile à casser (bruteforce).
- Simplicité des calculs → rapidité de traitement
- Besoin en ressource et en mémoire faible
- flexibilité d'implémentation (taille des blocs et des clés)
- Hardware et software
- Simplicité : le design de l'AES est relativement simple



- Recommandation Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI)

- Chiffrement symétrique

- Algorithme: AES
    - Taille des clés min: 128bits
    - Chiffrement par bloc: 128bits
    - Algo Chiffrement par flot: aucune attaque ne nécessitant moins de  $2^{218}$  opérations ne doit être connue
    - **Plutôt Bloc que Flot**



[https://www.ssi.gouv.fr/uploads/2015/01/RGS\\_v-2-0\\_B1.pdf](https://www.ssi.gouv.fr/uploads/2015/01/RGS_v-2-0_B1.pdf)

# Chiffrement Asymétrique

- 
- Propriétés
  - Diffie-Hellman
  - RSA
  - Courbe elliptique
  - Bilan

## • Chiffrement asymétrique

- ❑ Plus lent que le chiffrement symétrique
- ❑ Consommateur de ressource
- ❑ Permet un passage à l'échelle
- ❑ Distribution de clé
- ❑ Utiliser pour la distribution de clé de session
- ❑ Exemples d'algorithmes de chiffrement
  - Diffie-Hellman
  - Rivest, Shamir, Adleman (**RSA**)
  - Courbe Elliptique
  - El Gamal
  - Digital Signature Algorithm (DSA)
  - Knapsak



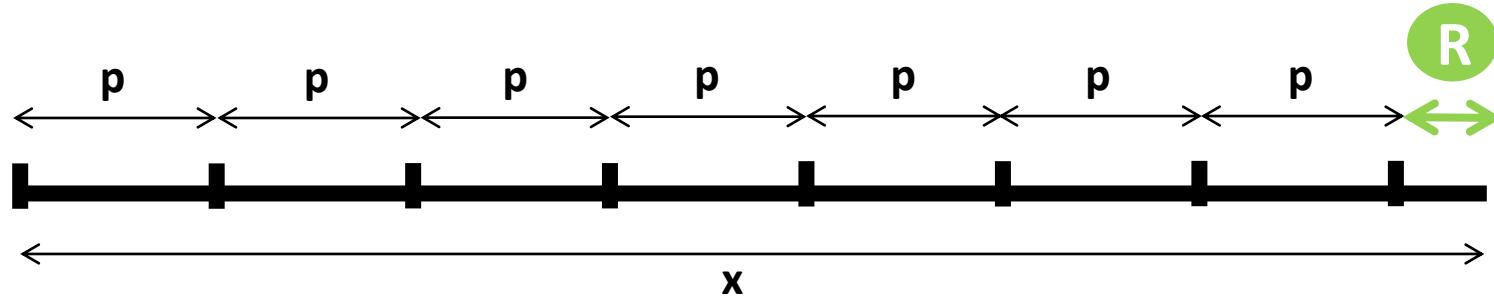
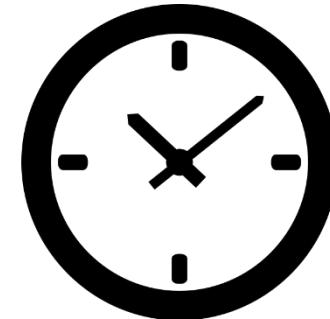
# Chiffrement Asymétrique

- 
- Propriétés
  - Diffie-Hellman
  - RSA
  - Courbe elliptique
  - Bilan

- Objectif

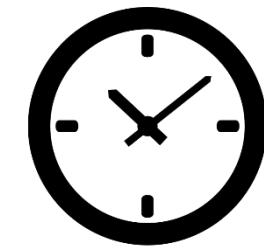
→ trouver une fonction qui est rapide et facile dans un sens et lente et complexe dans l'autre

$$x \bmod p \equiv R$$

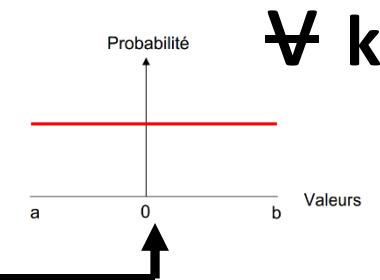
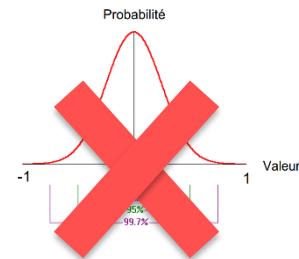


$$x \bmod p \equiv R$$

$$46 \bmod 12 \equiv 10$$



$$3^k \bmod 17$$



**3 Est un générateur**  
**17 Est le module**

$$3^{15} \bmod 17 \equiv R \rightarrow \text{EASY !}$$

---

$$\text{HARD !} \leftarrow 3^k \bmod 17 \equiv 6$$

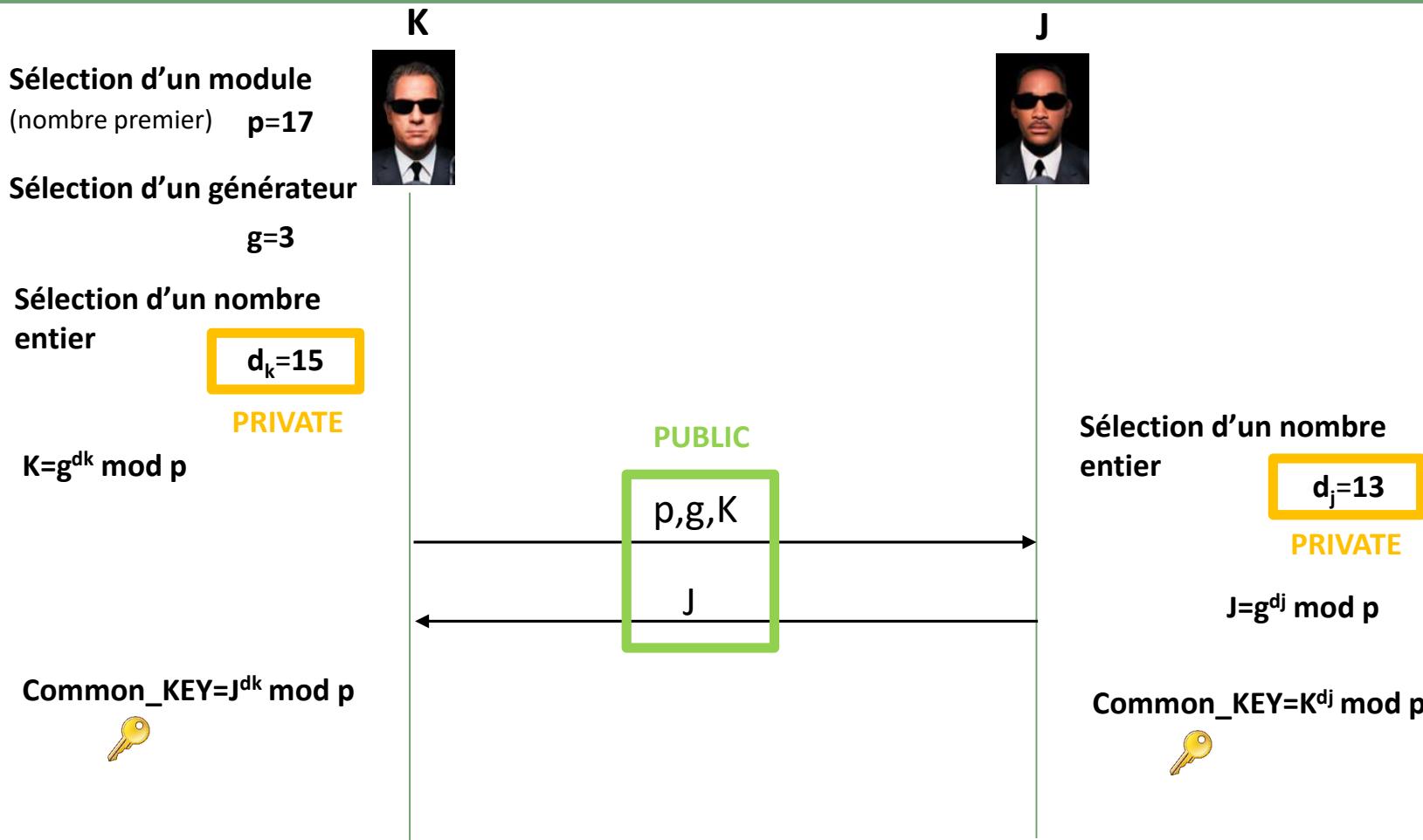
Si module = nombre premier très grand

$$3^{15} \bmod 17 \equiv R \rightarrow \text{EASY !}$$

$$\text{HARD !} \leftarrow 3^k \bmod 17 \equiv 6$$

## Problème des logarithmes discrets

- Chiffrement asymétrique: Diffie-Hellman



- Chiffrement asymétrique: Diffie-Hellman

K



$$d_k = 15$$

$$K = g^{dk} \bmod p$$

$$\text{Common\_KEY} = J^{dk} \bmod p$$

$$(g^{d_j} \bmod p)^{d_k} \bmod p$$

$$(g^{d_j \times d_k} \bmod p) \bmod p$$

$$g^{d_j \times d_k} \bmod p$$

J



$$d_j = 13$$

$$J = g^{dj} \bmod p$$

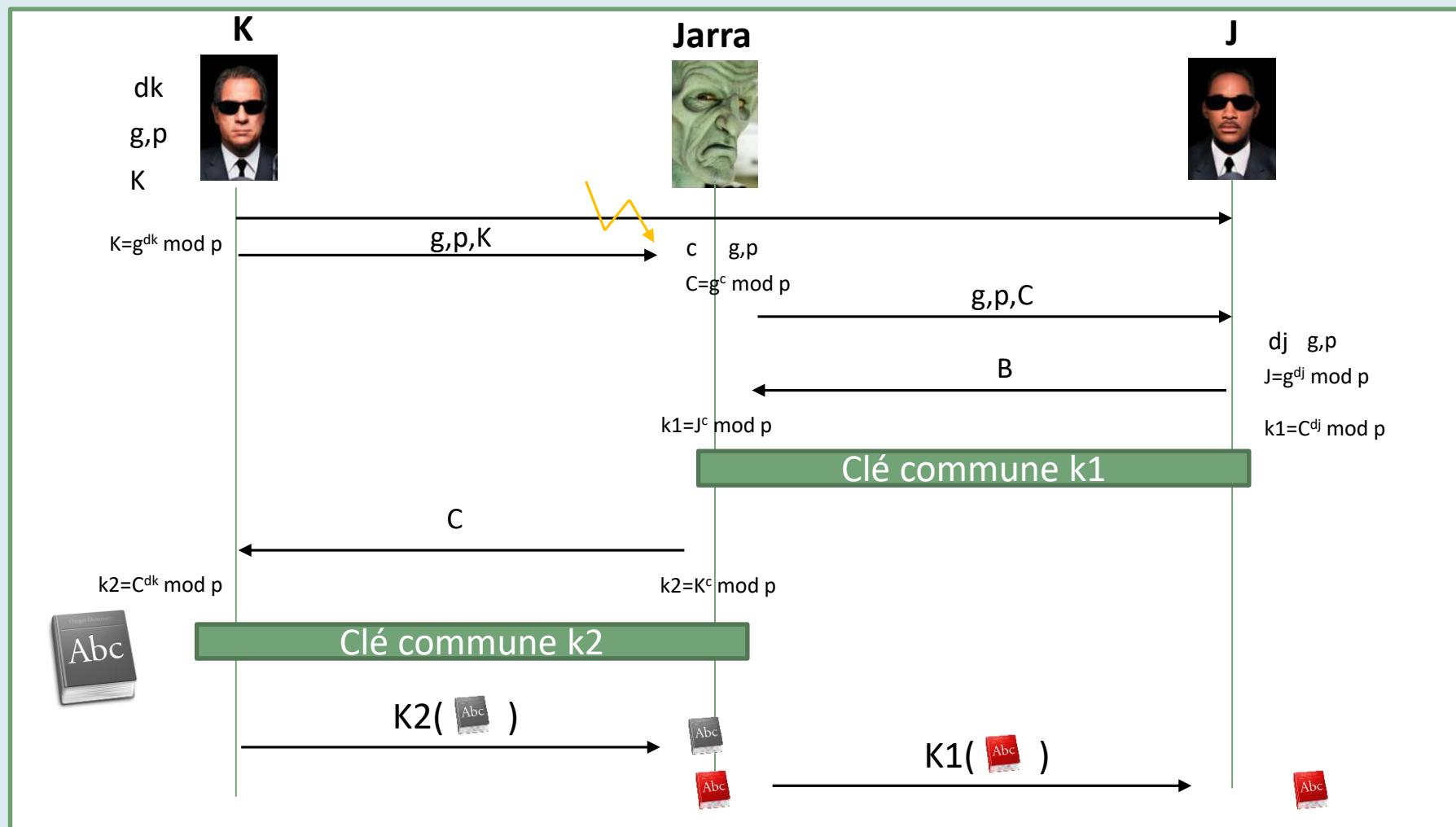
$$\text{Common\_KEY} = K^{dj} \bmod p$$

$$(g^{d_k} \bmod p)^{d_j} \bmod p$$

$$(g^{d_k \times d_j} \bmod p) \bmod p$$

$$g^{d_k \times d_j} \bmod p$$

- Chiffrement asymétrique: Diffie-Hellman man in the middle



- **La cryptologie: Méthodes de chiffrement asymétrique A vous de Jouer!**

K



J



J a besoin de récupérer des informations de K , utiliser Diffie-Hellman pour les faire communiquer

## • Chiffrement asymétrique: Diffie-Hellman

- ❑ Vulnérable à l'attaque man in the middle
- ❑ Force de l'algorithme repose sur la difficulté du problème de logarithme discret retrouver  $g^a, g^b$  à partir de  $g^{ab}$  très complexe
- ❑ Nécessiter de vérifier l'identité de son interlocuteur avant de prendre la clé publique



# Chiffrement Asymétrique

- 
- Propriétés
  - Diffie-Hellman
  - RSA
  - Courbe elliptique
  - Bilan

- Chiffrement asymétrique: RSA

La fonction Phi ou indicateur d'Euler

$\phi(n)$

Propriétés:  $\forall n \in \mathbb{N}^*$

$$\phi(n) = vcard(\{m \in \mathbb{N}^* \mid m \leq n, m \text{ premier avec } n\})$$

$$\phi(A \times B) = \phi(A) \times \phi(B)$$

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array}$$

$$\phi(8) = 4$$

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}$$

$$\phi(7) = 6$$

- Chiffrement asymétrique: RSA

La fonction Phi ou indicateur d'Euler

$$\phi(n)$$

Calculer  $\phi(n)$  est **difficile sauf pour les nombres premiers**

$$\phi(Prime) = Prime - 1$$

1

2

3

4

5

6

7

$$\phi(7) = 6$$

$\forall P1, P2$  nombre premier

$$\phi(P1 \times P2) = \phi(P1) \times \phi(P2)$$

$$\phi(P1 \times P2) = (P1 - 1) \times (P2 - 1)$$

- Chiffrement asymétrique: RSA

La fonction Phi ou indicateur d'Euler

$$\phi(n)$$

Calculer  $\phi(n)$  est **difficile sauf pour les nombres premiers**

$$\phi(P_1 \times P_2) = \phi(P_1) \times \phi(P_2) = R \rightarrow \text{EASY !}$$

HARD ! 

$$\phi(n) = \phi(P_1) \times \phi(P_2) = R$$

- Chiffrement asymétrique: RSA

La fonction Phi ou indicateur d'Euler

$$\phi(n)$$

**Comment utiliser  $\phi(n)$   
avec l'exponentiation  
modulaire  $m^e \text{mod } n$  ?**

- Chiffrement asymétrique: RSA

## Théorème d'Euler

$$m^{\phi(n)} \equiv 1 \pmod{n}$$

Avec m et n sans facteur commun

$$1^k = 1$$

$$m^{k \times \phi(n)} \equiv 1^k \pmod{n}$$

$$1 \times m = m$$

$$m \times m^{k \times \phi(n)} \equiv m \times 1^k \pmod{n}$$

$$m^{k \times \phi(n) + 1} \equiv m \pmod{n}$$

Hypothèse

$$m^{e \times d} \equiv m \pmod{n}$$

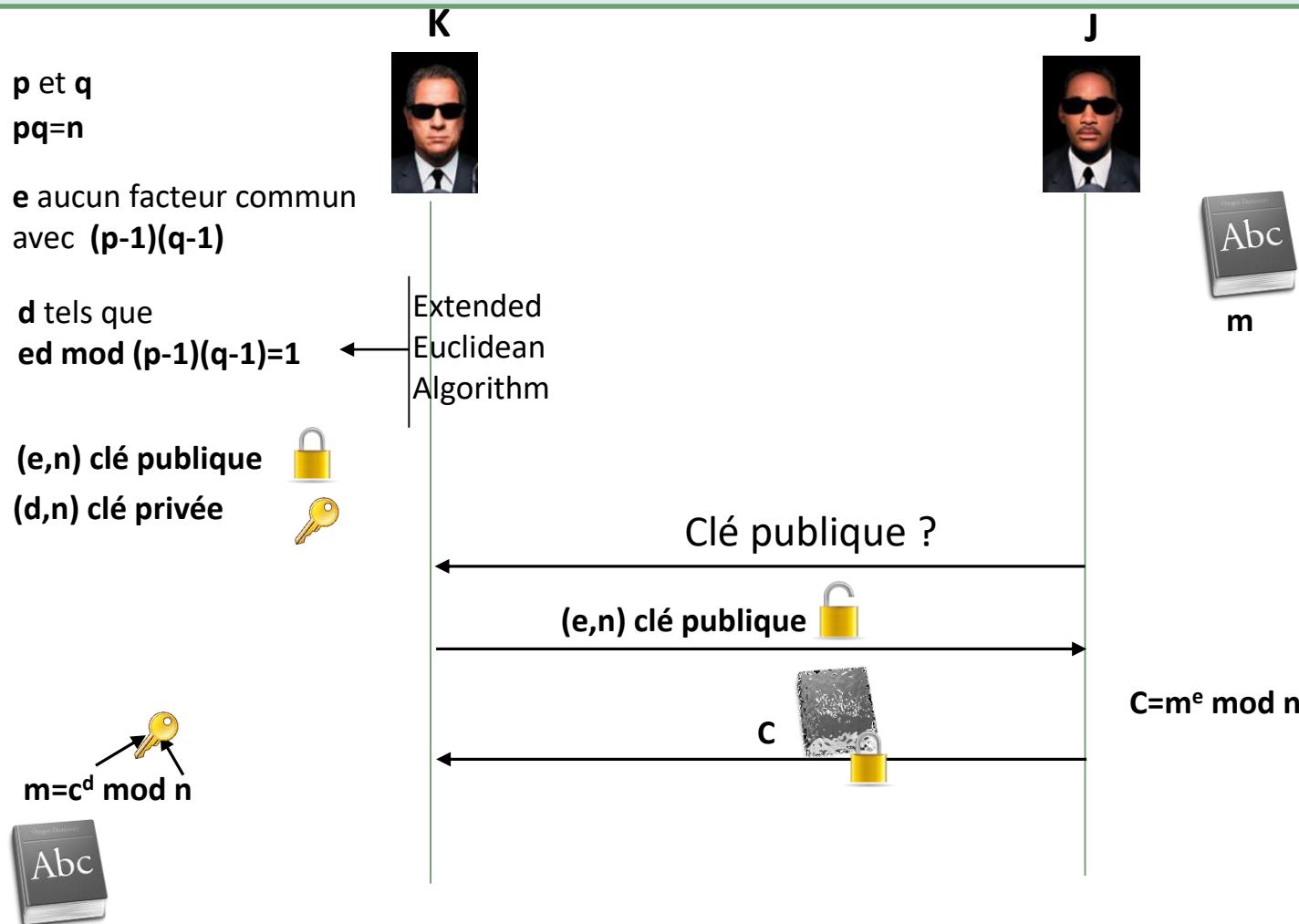
PUBLIC

$$e \times d = k \times \phi(n) + 1$$

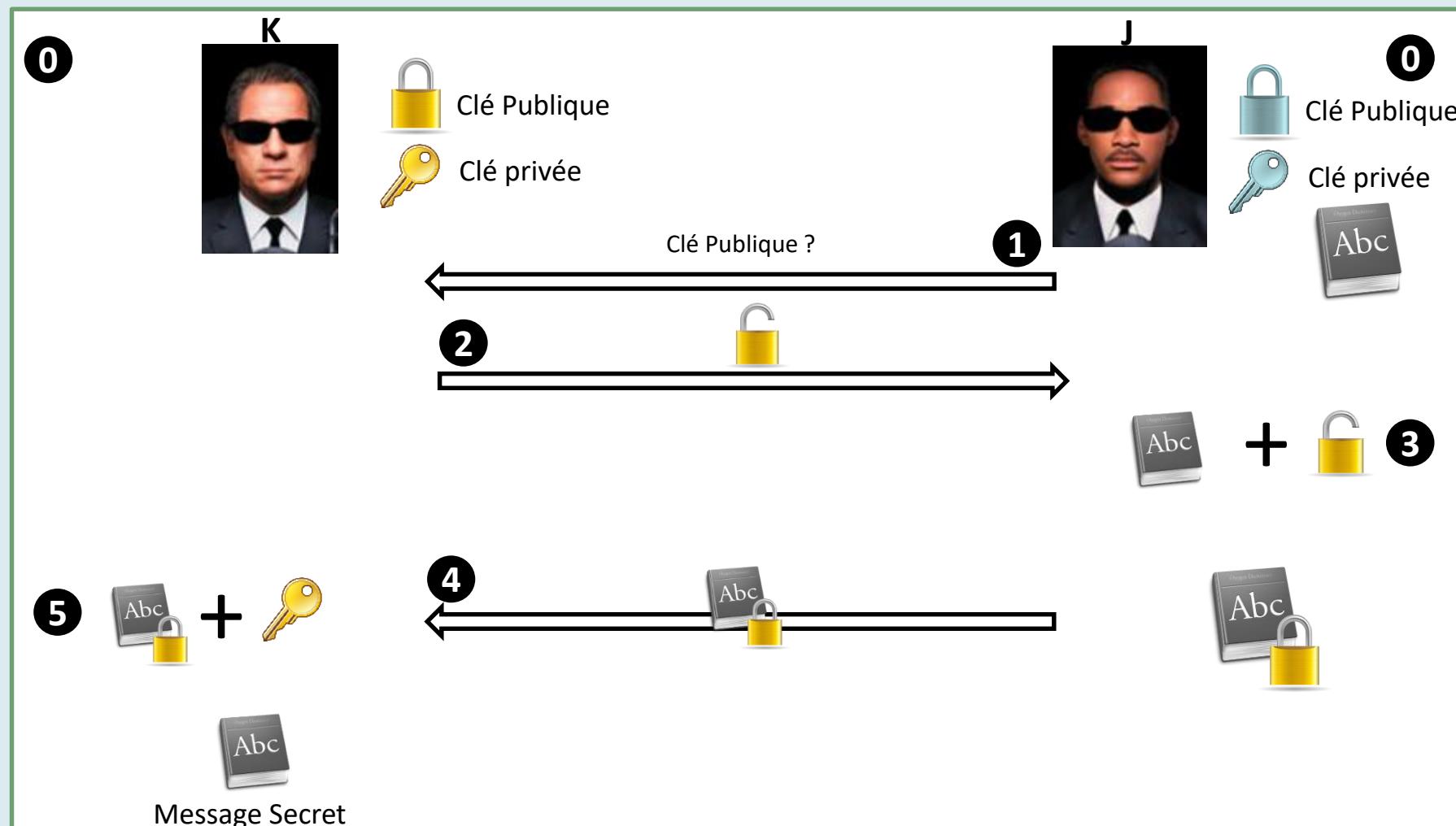
$$d = \frac{k \times \phi(n) + 1}{e}$$

PRIVATE

- Chiffrement asymétrique: RSA



- La cryptologie: Méthodes de chiffrement asymétrique



- La cryptologie: Méthodes de chiffrement asymétrique A vous de Jouer!

K



J



J a besoin de récupérer des informations de K , utiliser RSA pour les faire communiquer

## • Chiffrement asymétrique: RSA

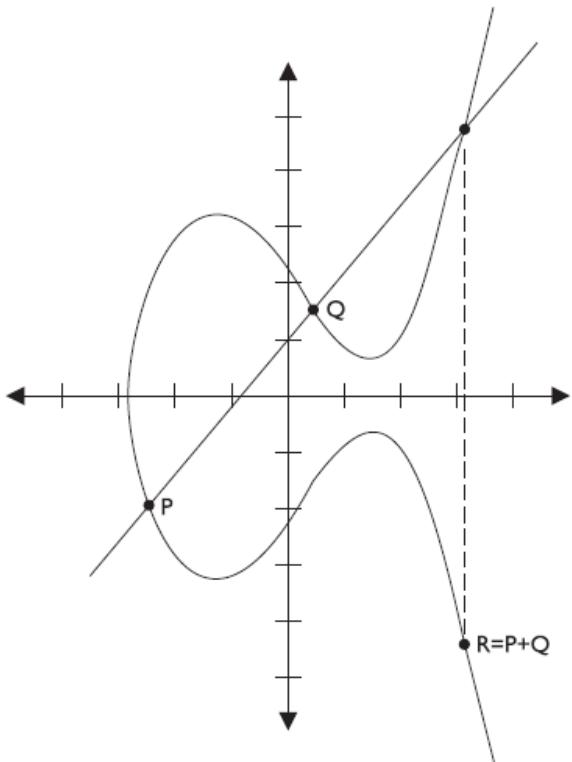
- ❑ Rivest, Shamir, Adleman
- ❑ Sélection des paramètres:
  - ❑ p et q choisis au hasard de façon à ce que p-q pas trop petit
  - ❑ p et q nombres premiers forts
    - ❑ p-1 possède un grand facteur premier
    - ❑ p+1 possède un grand facteur premier
- ❑ Peut être utilisé pour la signature numérique
- ❑ Force de l'algorithme repose sur la difficulté à factoriser n (calculer p et q)



# Chiffrement Asymétrique

- 
- Propriétés
  - Diffie-Hellman
  - RSA
  - Courbe elliptique
  - Bilan

- La cryptologie: ECC Elliptic Curve Cryptosystem



$$y^2 = x^3 + ax + b$$

- La cryptologie: ECC Elliptic Curve Cryptosystem

Choix d'une courbe elliptique  $E(a,b,K)$



Choix d'un point  $P$  sur la courbe



Sélection d'un entier  $k_a$

$E(a,b,K), P$  Information publique

$(k_a * P)$  clé publique



$(k_a)$  clé privée



$(k_a k_b)P$  clé commune



Clé publique ?

$E(a,b,K), P \quad (k_a * P)$



Clé commune

Sélection d'un entier  $k_b$

$(k_a k_b)P$  clé commune



## • La cryptologie: ECC Elliptic Curve Cryptosystem

- ❑ Calcul d'une clé commune (semblable Diffie-Hellman)
- ❑ Complexité mathématique plus élevée que RSA pour cryptanalyse
- ❑ Taille de clé plus petite permettant d'assurer une sécurité équivalente à RSA (200 bits ECC contre 1024 bits pour RSA)
- ❑ Complexité des calculs peu élevée pour le calcul de la clé commune
- ❑ Beaucoup de brevet sur les courbes elliptiques dans la cryptographie (couteux)
- ❑ Théorie des courbes elliptiques encore récentes (trappes potentielles)



- Recommandation Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI)

- Chiffrement asymétrique

- Algorithme:

- RSAES-OAEP,...

- Taille des clés min:

- 2048 bits (2030)

- puis 3072 bits

- Propriétés:

- sous-groupes dont l'ordre

- est multiple d'un nombre

- premier d'au moins 256

- bits (pour RSA)



[https://www.ssi.gouv.fr/uploads/2015/01/RGS\\_v-2-0\\_B1.pdf](https://www.ssi.gouv.fr/uploads/2015/01/RGS_v-2-0_B1.pdf)

- **Quel futur pour les algorithmes de chiffrement ?**

- Comment casser un RSA ? Trouver p et q de  $n=p.q$

- Trouver un **a** tel que  $a < N$  et relativement premier à **N**

**PGDC(a,N)=1**

Très long !

- Trouver **r** tel que **r** est la période de **a mod N**

- Vérifier que **r** est pair et  $a^{r/2} + 1 \not\equiv 0 \pmod{N}$

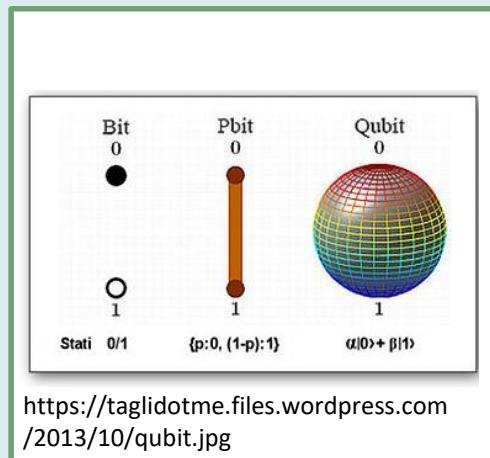
$$a^r \equiv 1 \pmod{N}$$

$$a^r - 1 \equiv 0 \pmod{N} \rightarrow a^r - 1 \equiv k.N \rightarrow (a^{\frac{r}{2}} - 1) \cdot (a^{\frac{r}{2}} + 1) \equiv k.p.q$$

- Résoudre

$$\text{PGCD}\left((a^{\frac{r}{2}} - 1), p\right)$$

$$\text{PGCD}\left((a^{\frac{r}{2}} + 1), q\right)$$

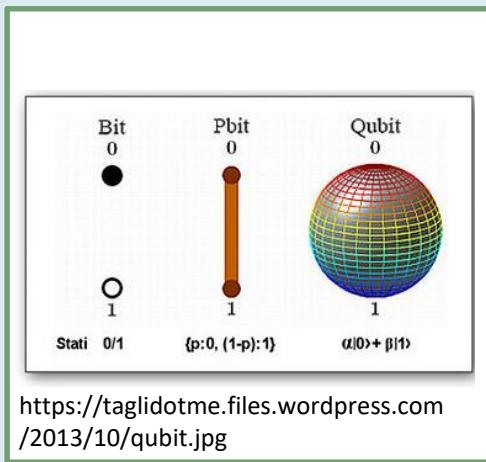


- Quel futur pour les algorithmes de chiffrement ?  
**Facile avec un ordinateur quantique**

- Trouver  $r$  tel que  $r$  est la période de  $a \text{ mod } N$



<https://www.youtube.com/watch?v=wUwZZal5u0c>

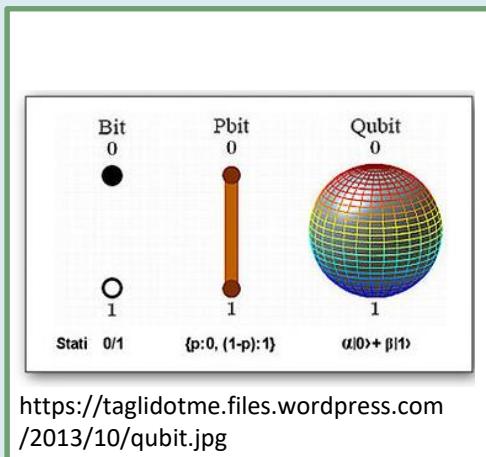


- Quel futur pour les algorithmes de chiffrement ?

- Une nouvelle alternative : **Quantum Key Distribution**

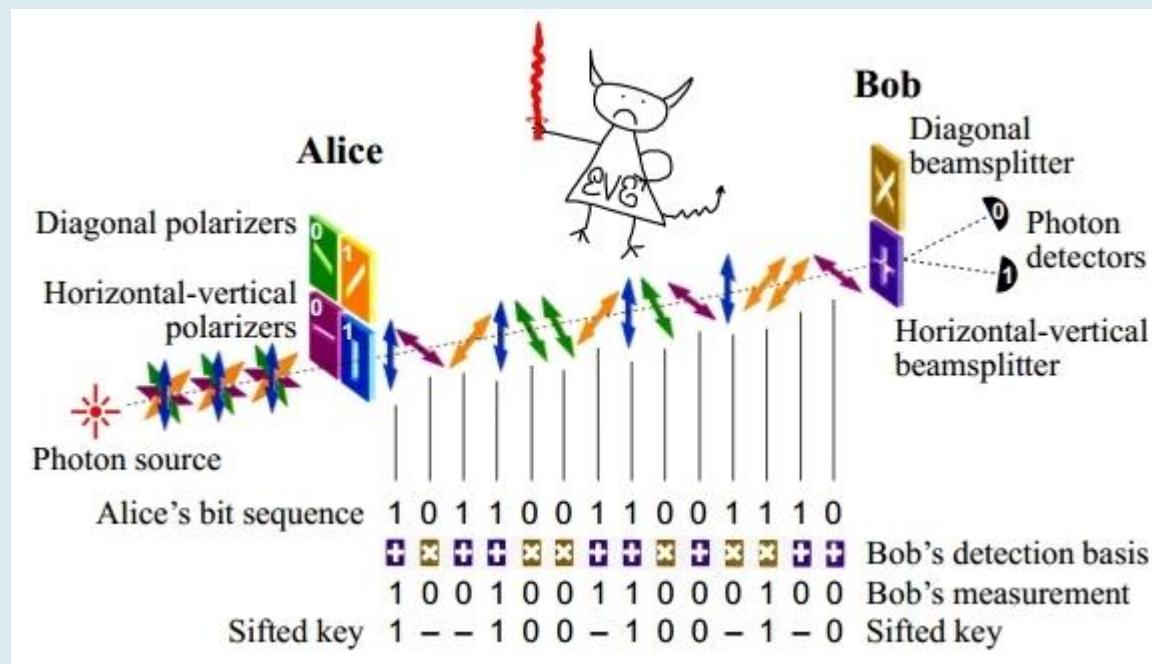


[http://qubitekk.com/wp-content/uploads/2015/12/QKD\\_product\\_small.jpg](http://qubitekk.com/wp-content/uploads/2015/12/QKD_product_small.jpg)



- Quel futur pour les algorithmes de chiffrement ?

- Une nouvelle alternative : **Quantum Key Distribution**



# Chiffrement Asymétrique

- 
- Propriétés
  - Diffie-Hellman
  - RSA
  - Courbe elliptique
  - Bilan

## • La cryptologie: Bilan symétrique asymétrique

- ❑ Utilisation de système hybride
- ❑ Utilisation de la puissance des algorithmes asymétriques pour l'échange de clé
- ❑ Utilisation du chiffrement symétrique rapide pour chiffrer les contenus



## Fonctions de Hachage

- 
- Propriétés
  - MD5
  - SHA

## • La cryptologie: Fonctions de Hachage

### Fonction de hachage ou One Way Hash

*Fonction capable à partir un élément de taille variable de fournir une valeur de taille fixe appelée empreinte ou hash.*

### Utilisation de fonction à sens unique

*Fonction facile à calculer dans un sens mais très difficile à inverser*

### Propriétés

#### Calcul rapide

#### Eviter les collisions (2 données différentes représentées par une même empreinte)

#### Possibilité d'avoir une empreinte plus grande que données initiale (protection des mots de passe)

#### Volonté qu'un seul changement de bits entraîne un changement important dans l'empreinte résultante



## • La cryptologie: Fonctions de Hachage

- ❑ Propriétés nécessaires pour la cryptographie
  - ❑ Très difficile de trouver un message à partir de son empreinte
  - ❑ Très difficile à partir d'un message et de son empreinte de générer un message différent possédant la même empreinte
  - ❑ Très difficile de trouver 2 messages aléatoires possédant la même empreinte
- ❑ Notion de salting (salage)

*Ajout d'une chaîne pseudo-aléatoire au message avant le hash*

e.g. password + MD5(login) → SHA (password + MD5(login))

→ évite les attaques par table de hash.



## • La cryptologie: Fonctions de Hachage

- Exemple de fonction de Hachage
  - HMAC
  - CBC-MAC
  - MD5
  - SHA



## Fonctions de Hachage

- 
- Propriétés
  - MD5
  - SHA

- Les Fonctions de Hachage: MD5

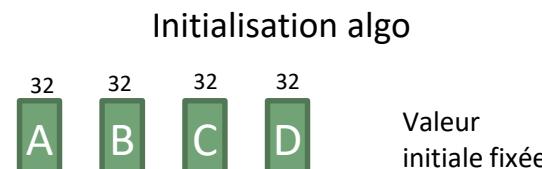
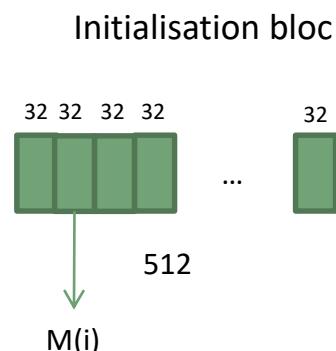
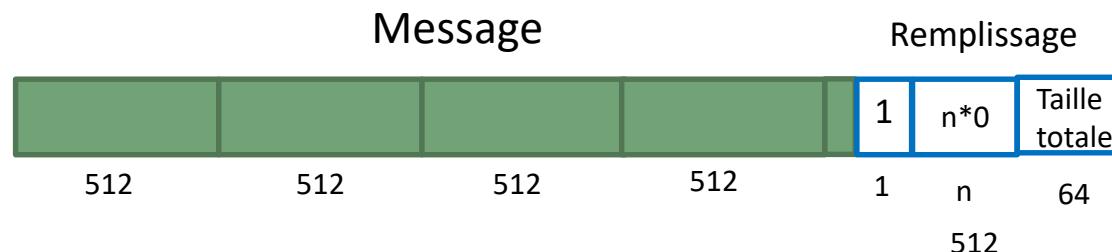
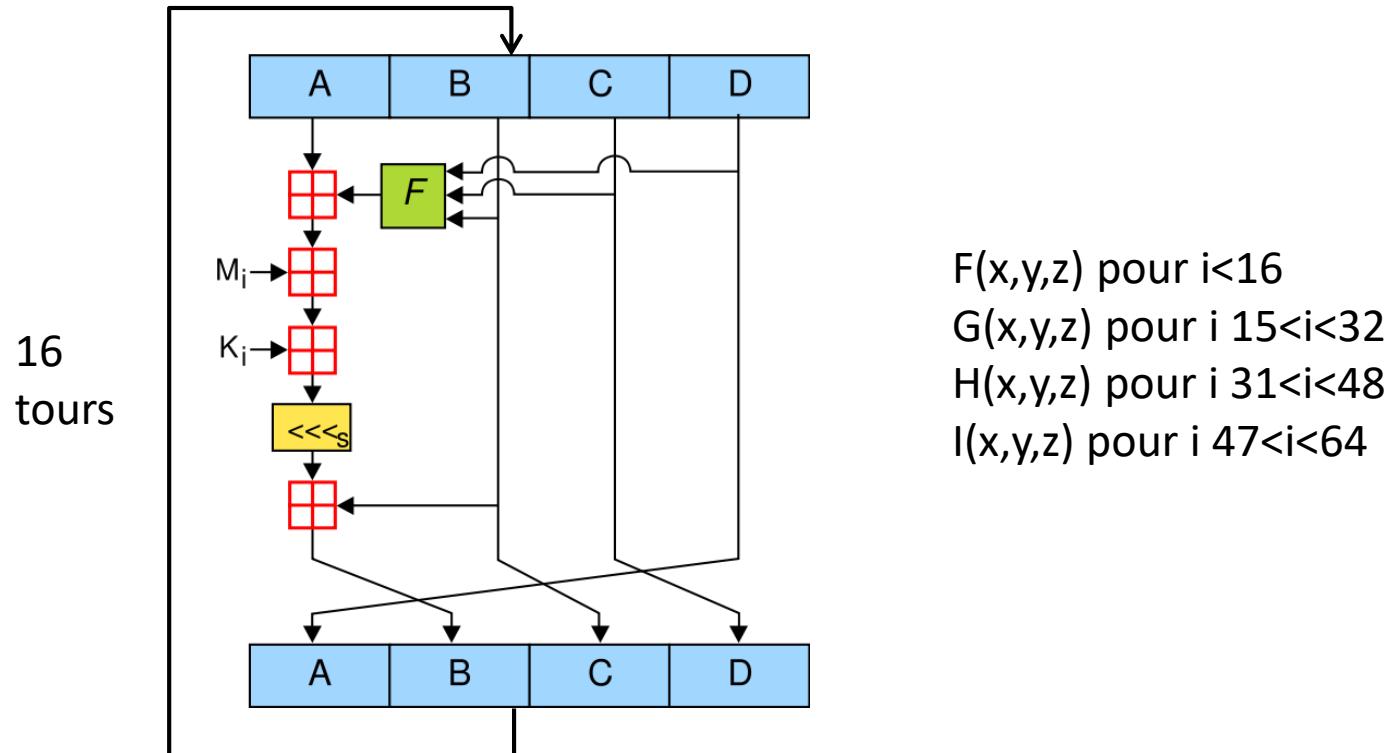


Tableau de valeur fixée

K[i] / i in [0,63]

$$\begin{aligned}
 F(x,y,z) &= (x \text{ AND } y) \text{ OR } (\text{not}(x) \text{ AND } z) \\
 G(x,y,z) &= (x \text{ AND } z) \text{ OR } (y \text{ AND } \text{not}(z)) \\
 H(x,y,z) &= x \text{ XOR } y \text{ XOR } z \\
 I(x,y,z) &= x \text{ XOR } (x \text{ AND } \text{not}(z))
 \end{aligned}$$

- Les Fonctions de Hachage: MD5



MD5(" The quick brown fox jumps over the lazy dog ")

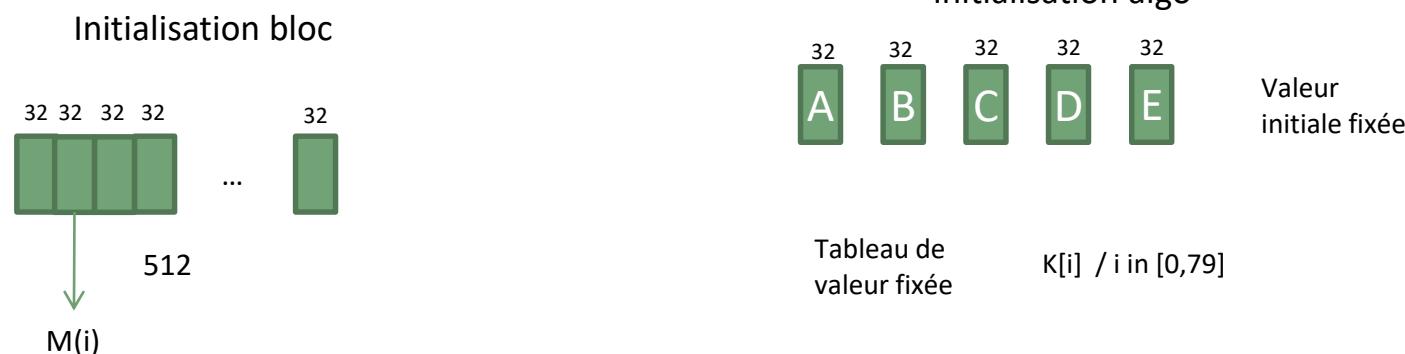
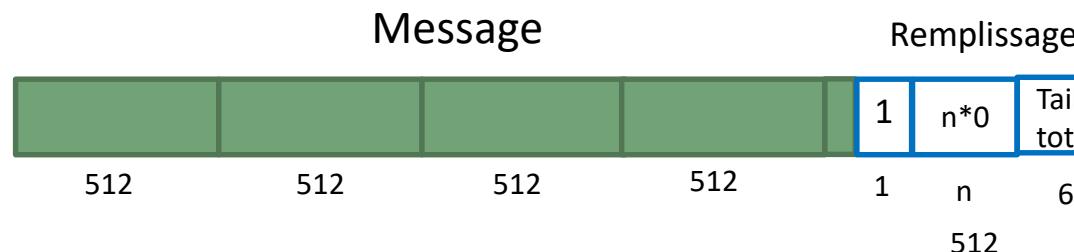
9e107d9d372bb6826bd81d3542a419d6

## • Les Fonctions de Hachage: MD5

- ❑ Message Digest 5
- ❑ Ronald Rivest 1991
- ❑ 1996 faille grave de collision
- ❑ 2004 découvert des collisions complètes → SHA 256



- Les Fonctions de Hachage: SHA

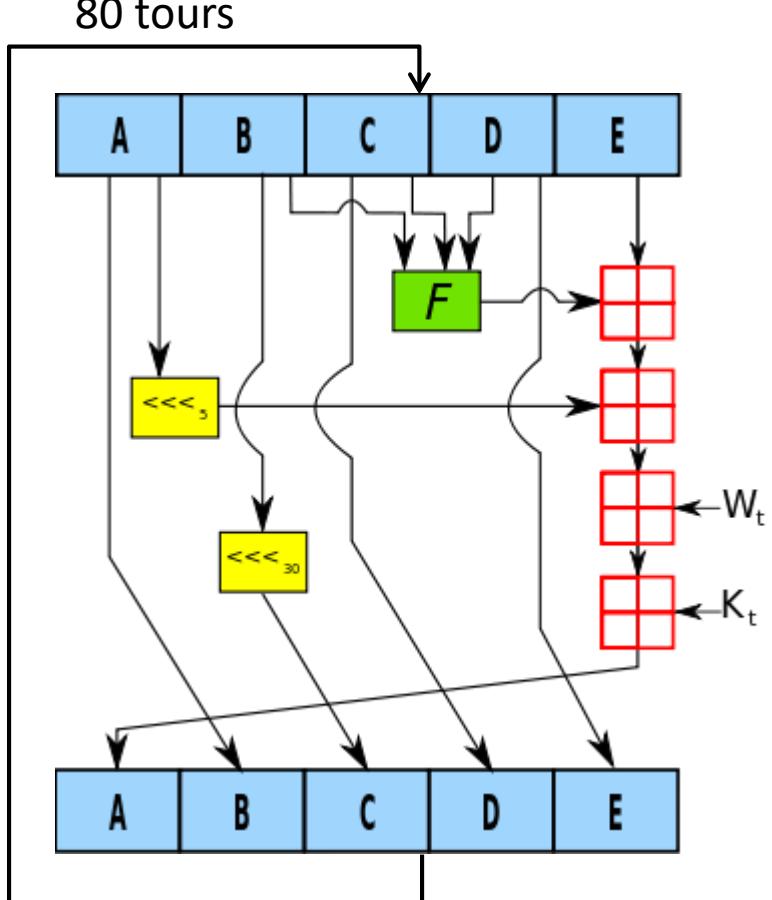


$$F(x,y,z) = (x \text{ AND } y) \text{ OR } (\text{not}(x) \text{ AND } z)$$

$$G(x,y,z) = x \text{ XOR } y \text{ XOR } z$$

$$H(x,y,z) = (x \text{ AND } z) \text{ OR } (x \text{ AND } z) \text{ OR } (y \text{ AND } z)$$

- Les Fonctions de Hachage: SHA



$F(x,y,z)$  pour  $i < 20$   
 $G(x,y,z)$  pour  $i \geq 21 \text{ et } i < 40$   
 $H(x,y,z)$  pour  $i \geq 39 \text{ et } i < 60$   
 $G(x,y,z)$  pour  $i \geq 59 \text{ et } i < 80$

## • Les Fonctions de Hachage: SHA

- ❑ Secure Hash Algorithm
- ❑ NSA SHA0(1993), SHA1(1995)
- ❑ 2005 faille de sécurité découverte, pas d'attaques réelles  
(SHA3 à venir)



Algorithm and variant		Output size (bits)	Internal state size (bits)	Block size (bits)	Max message size (bits)	Word size (bits)	Rounds	Operations	Collisions found?
SHA-0									Yes
SHA-1		160	160	512	$2^{64} - 1$	32	80	add, and, or, xor, rotate, mod	Theoretical attack ( $2^{51}$ )
SHA-2	SHA-256/224	256/224	256	512	$2^{64} - 1$	32	64	add, and, or, xor, rotate, mod, shift	No
	SHA-512/384	512/384	512	1024	$2^{128} - 1$	64	80		

<http://en.wikipedia.org/wiki/SHA-1>

## • Les Fonctions de Hachage: Bilan

- ❑ MD5 et SHA1 encore très utilisés
- ❑ Préférable d'utiliser SHA256
- ❑ Permettent d'assurer l'intégrité d'un document
- ❑ Utilisé pour la signature numérique conjointement avec le chiffrement asymétrique
- ❑ Utilisé pour protéger du contenu stocké
  - ❑ Sous linux traditionnellement MD5
  - ❑ Possible de préciser la méthode

```
password sufficient pam_unix.so min=4 sha256
```



## Bilan Eléments Chiffrement

---

## • Bilan Elément de chiffrement

Algorithm Type	Encryption	Digital Signature	Hashing Function	Key Distribution
<b>Asymmetric Key Algorithms</b>				
RSA	X	X		
ECC	X	X		X
Diffie-Hellman				X
El Gamal	X	X		X
DSA		X		
LUC	X	X		X
Knapsack	X	X		X
<b>Symmetric Key Algorithms</b>				
DES	X			
3DES	X			
Blowfish	X			
IDEA	X			
RC4	X			
SAFER	X			
<b>Hashing Algorithms</b>				
Ronald Rivest family of hashing functions: MD2, MD4, and MD5			X	
SHA			X	
HAVAL (variable-length hash values using a one-way function design)			X	

# PKI : Public Key Infrastructure

- 
- Besoin et définition
  - Architecture
  - Bilan

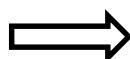
- PKI – Public Key Infrastructure: Besoin



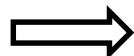
Comme faire confiance à son interlocuteur ?



Comment s'assurer que son interlocuteur  
est bien là personne qu'elle prétend être ?



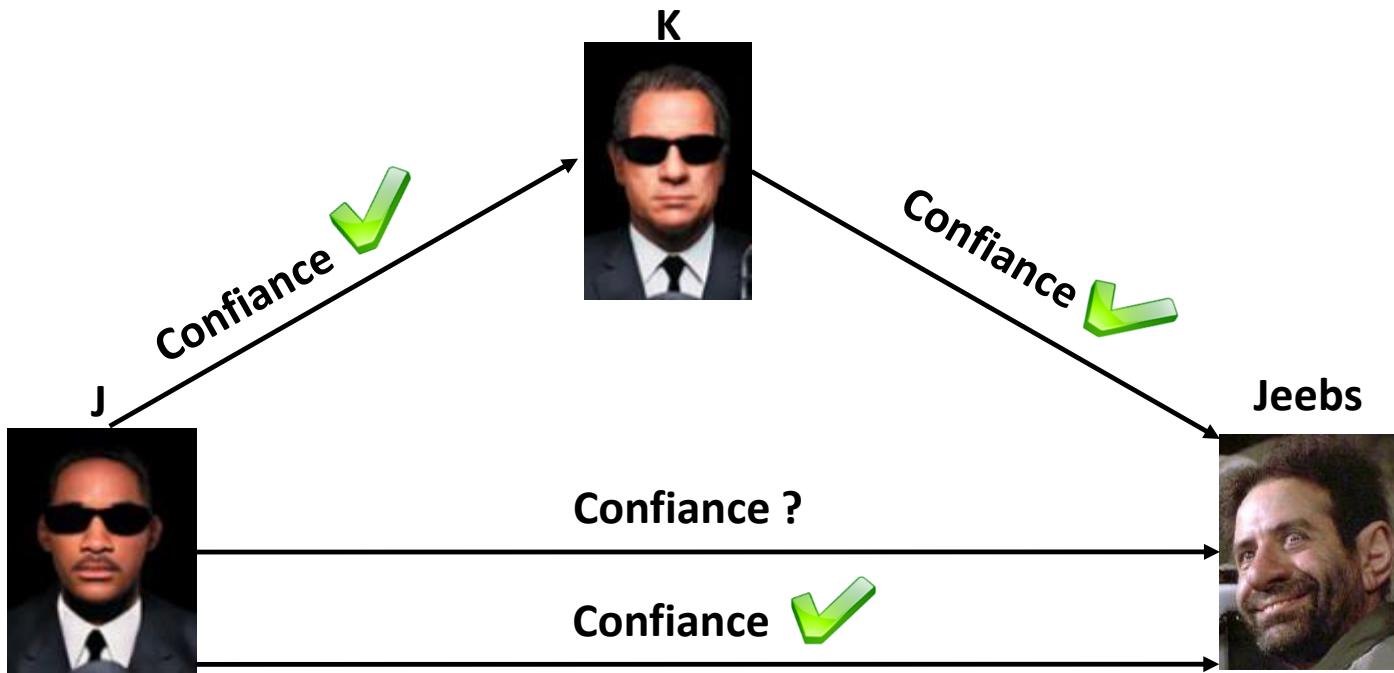
Utilisation d'un Tiers de  
confiance



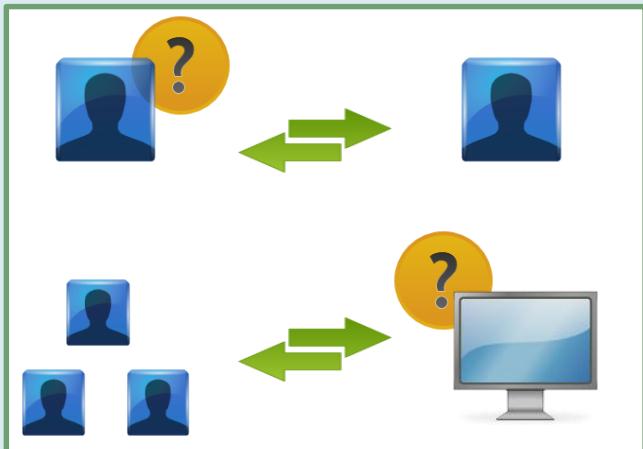
Utilisation de certificats

- PKI – Public Key Infrastructure: Besoin

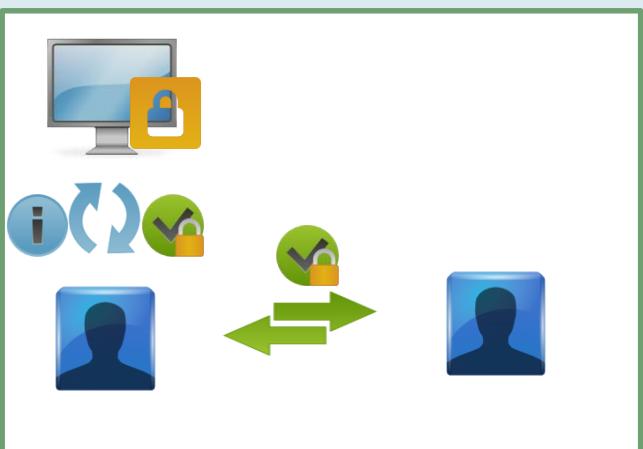
## Tiers de confiance



## • PKI – Public Key Infrastructure: Besoin



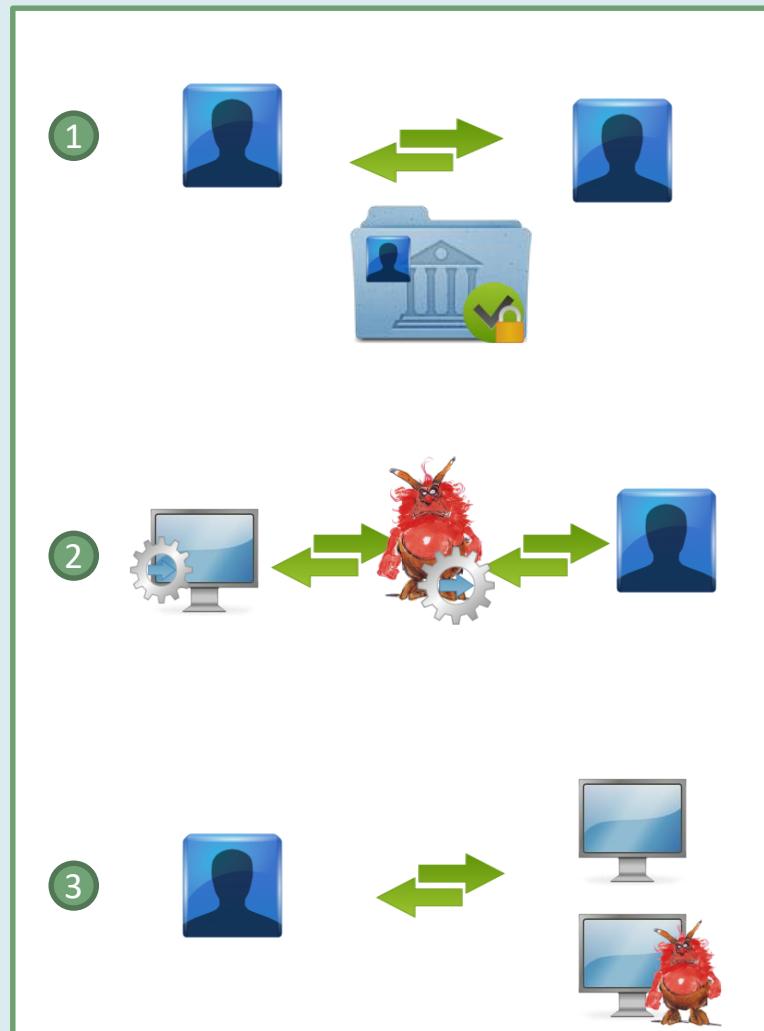
- ❑ Comment assurer son identité vis-à-vis d'un tiers ?
- ❑ Comment assurer que les entités sont bien celles qu'elles prétendent être ?
- ❑ Une « autorité de confiance » signe avec sa clé privée un document contenant
  - L'identité d'une entité possédant un couple de clé
  - La clé publique
  - Des informations décrivant l'usage de cette clé
  - ...
- Résultat = Certificats
- Autorité de confiance = Autorité de certification



## • PKI – Public Key Infrastructure: Besoin

### Objectif d'un certificat

- 1 Prouve l'identité d'une personne au même titre qu'une carte d'identité, dans le cadre fixé par l'autorité de certification qui l'a validé.
- 2 Pour une application il assure que celle-ci n'a pas été détournée de ses fonctions.
- 3 Pour un site il offre la garantie lors d'un accès vers celui-ci que l'on est bien sur le site auquel on veut accéder.



## • PKI – Public Key Infrastructure: Utilisation

### ❑ Qui utilise les certificats

- IPSec
- SSL
- S/MIME (PGP)
- Signature de code de package (Java, Javascript, ActiveX,...)
- Signature de formulaire,...



### ❑ Format de type de certificats

- X509 PKIX (UIT, 1988, RFC 5280)
- PKCS (rsa)
- PGP (Phil Zimmermann, 1991, GnuPG)
- SPKI/SDSI ([IETF](#), 1996, RFC 2692, RFC 2693)

## • PKI – Public Key Infrastructure: Les acteurs

- Les utilisateurs (homme, machine, service)

*Entités utilisant les certificats afin de vérifier l'identité d'autres utilisateurs mais aussi afin de connaître la clé public des ces derniers*



- L'autorité de certification – Certification Authority (CA)

*Entité de confiance délivrant et révoquant des certificats (certificats à clé publique)*

- L'autorité d'enregistrement – Registration Authority (RA)

*Entité en qui le CA a confiance pour vérifier l'identité de l'utilisateur*

## • PKI – Public Key Infrastructure: Les acteurs

- ❑ Certificat

*Object représentant l'identité d'un utilisateur et contenant la clé publique de ce dernier*



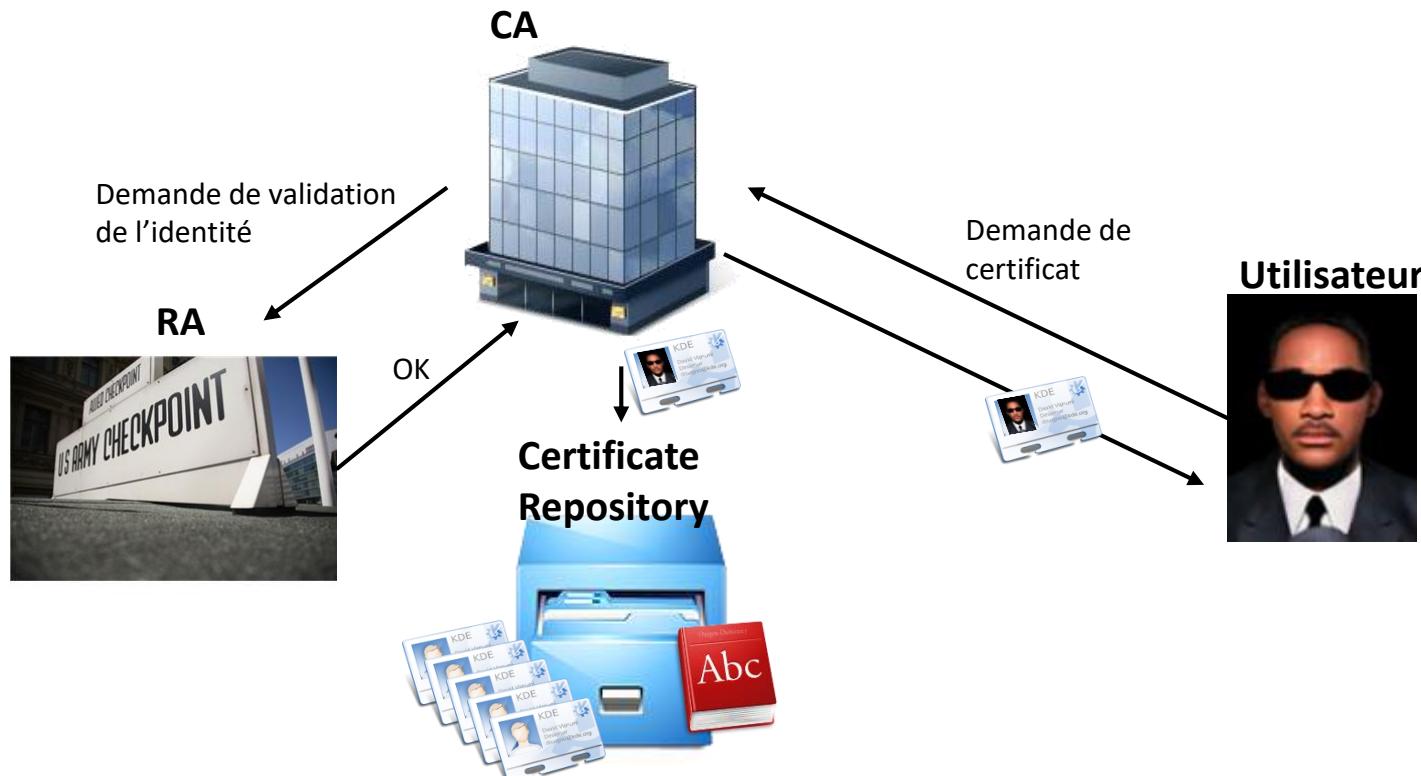
- ❑ Annuaire de certificats -Certificate Repository

*Object regroupant l'ensemble des certificats et des listes de révocation et les rend publique*

- ❑ Liste de révocation des certificats – Certificate Revocation List (CRL)

*Object regroupant l'ensemble des certificats révoqués*

- PKI – Public Key Infrastructure: Les acteurs



## • PKI – Public Key Infrastructure: Contenu d'un certificat

- Numéro de série
- Identité du porteur (owner)
- Identité du certifier émetteur (issuer)
- Période de validité (début-fin)
- Classe de certificat
- Clé public du porteur (+algo utilisé, longueur des clés,...)
- Signature (+algo utilisé, longueur des clés,...), auto-signé ou non



- PKI – Public Key Infrastructure: Contenu d'un certificat

Numéro de série  
Nom du porteur



Durée de validité  
Signature



Nom du certifier émetteur

- + Classe
- + Clé publique du porteur

- PKI – Public Key Infrastructure: Contenu d'un certificat**

Numéro de série

Data:

Version: 3 (0x2)

Serial Number: 1 (0x1)

Nom du certifier émetteur

Signature Algorithm: sha1WithRSAEncryption

Issuer: C=FR, ST=Rhone\_Alpes, L=Villeurbanne, O=INSA-LYON,  
OU=Dept Telecom, CN=CA/emailAddress=mitsuco26@hotmail.com

Durée de validité

Validity

Not Before: Jun 9 08:43:11 2011 GMT

Not After : May 9 08:43:11 2013 GMT

Nom du porteur

Subject:

C=FR, ST=Rhone\_Alpes, L=Villeurbanne, O=INSA-LYON,  
OU=Dept Telecom, CN=serveur radius/emailAddress=mitsuco26@hotmail.com

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public Key: (1024 bit)

Modulus (1024 bit):

00:b8:d1:ce:aa:e7:36:07:7f:46:5d:15:8d:24:25:

a7:2b:08:7d:5d:2c:78:21:94:8d:f0:c3:99:dd:d9:

18:8d:7d:89:5c:7a:43:b8:a5:4c:2c:69:db:49:4b:

e1:ea:9f:83:59:53:6b:f6:da:9e:5a:d3:ac:46:2f:

33:21:50:ac:f3:cc:c2:27:6e:e2:f2:d4:50:4d:fb:

f1:15:4f:3e:60:9b:07:6a:6c:65:17:bd:7c:c2:f7:

a1:d5:25:2f:23:35:39:d1:1f:ff:66:4e:ff:d6:7b:

04:50:e0:12:6e:71:7e:f3:bf:01:3a:d2:29:4a:bd:

7d:e1:89:9c:bf:1e:4a:60:99

Exponent: 65537 (0x10001)

Clé publique du porteur

- **PKI – Public Key Infrastructure: Contenu d'un certificat**

Classe

X509v3 extensions:

X509v3 Basic Constraints: critical

CA:FALSE

X509v3 Authority Key Identifier:

keyid:30:5B:05:AA:6E:D3:AE:2D:CD:45:25:05:0A:1F:A0:68:62:E5:67:7

X509v3 Subject Key Identifier:

54:52:EF:F4:94:39:18:5E:0A:9D:51:5C:AD:01:39:35:78:39:6F:35

X509v3 Key Usage:

Digital Signature, Key Encipherment

X509v3 Extended Key Usage:

TLS Web Server Authentication, TLS Web Client

Authentication

Netscape Cert Type: SSL Server

Netscape Comment: Certificat delivre par Dept Telecom

Signature

Signature Algorithm: sha1WithRSAEncryption

14:d2:ca:7d:66:5e:73:50:e3:28:14:30:cc:8c:ce:29:a8:d0:  
2c:fc:bd:ed:55:8c:60:43:c4:dc:1b:c9:6c:ef:59:ae:a8:54:  
e7:fa:e0:16:3b:2e:27:80:97:3c:f2:35:82:eb:4d:b3:33:ee:  
19:78:7e:f2:51:be:75:5f:78:32:23:65:9e:7f:f8:65:41:90:  
9c:41:6e:5d:5a:8c:94:52:06:e8:5c:b5:c1:d2:35:8d:90:37:  
1d:50:1e:7e:91:2b:67:b0:bf:c3:94:8e:0a:f5:54:3d:57:7b:

## • PKI – Public Key Infrastructure: Génération de la paire de clé

### Par le client

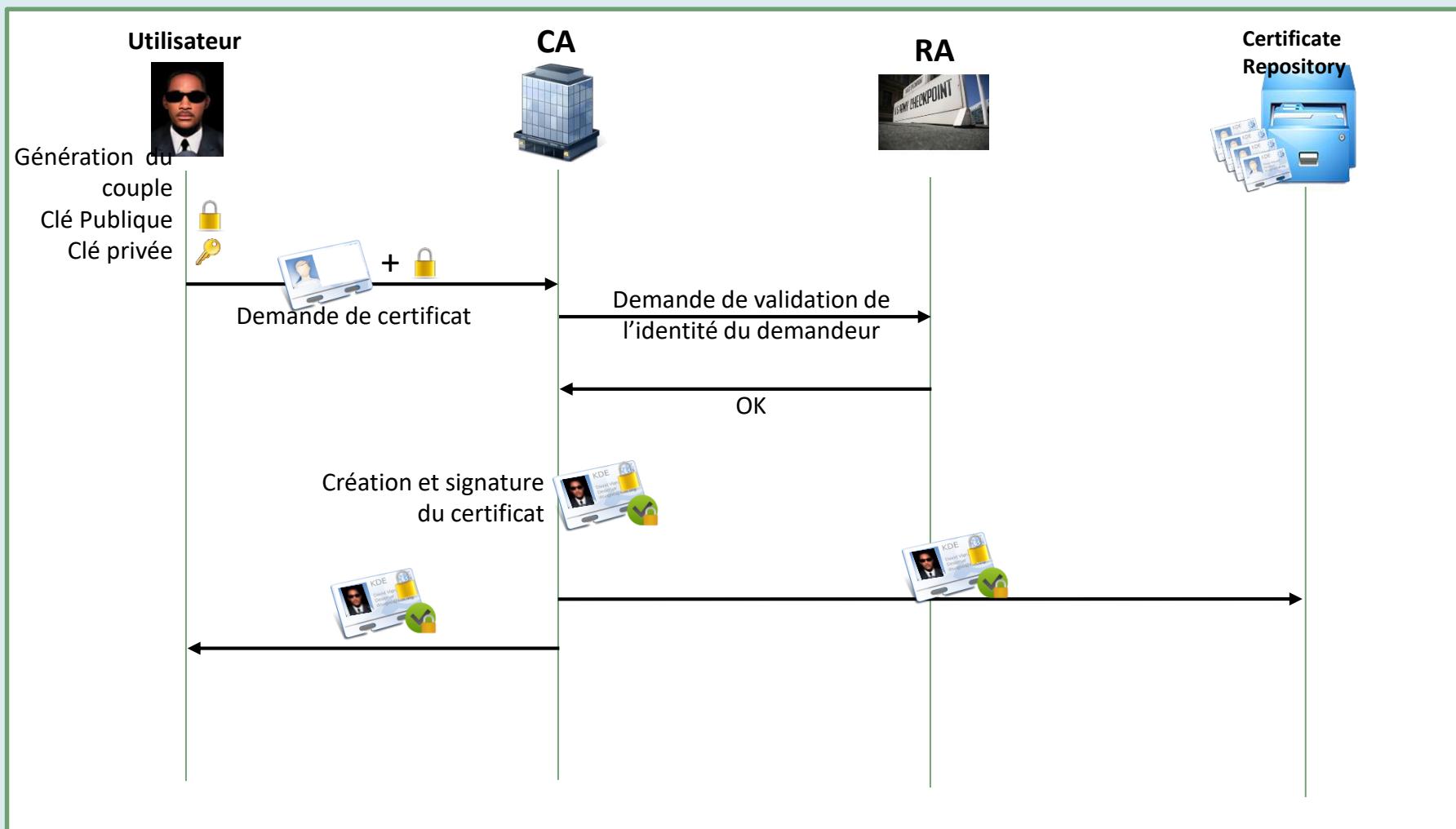
- Pas de communication de clé privée
- CA ne connaît pas la clé (perte de clé? Départ?)



### Par le CA

- Génération de clé plus sur (complexité, nombre aléatoire)
- Archivage de la clé privée
- Historique des paires de clés
- Doit transmettre de façon sécurité la clé privée

- PKI – Public Key Infrastructure: Cycle de vie



- **PKI – Public Key Infrastructure: Certification des certificats A vous de jouer**

1. Comment le CA fait –il pour certifier le certificat ?



2. Comment L'utilisateur peut –il être sûr de communiquer avec le CA ?



3. Comment L'utilisateur est sûr que son certificat n'a pas été modifié et provient bien du CA ?



- PKI – Public Key Infrastructure: Chaine de certification

Chaine de certification

Root CA



Est certifié par

CA<sub>3</sub>



Issuer Root  
CA

Est certifié par

CA<sub>2</sub>



Issuer CA<sub>3</sub>

Est certifié par

CA<sub>1</sub>



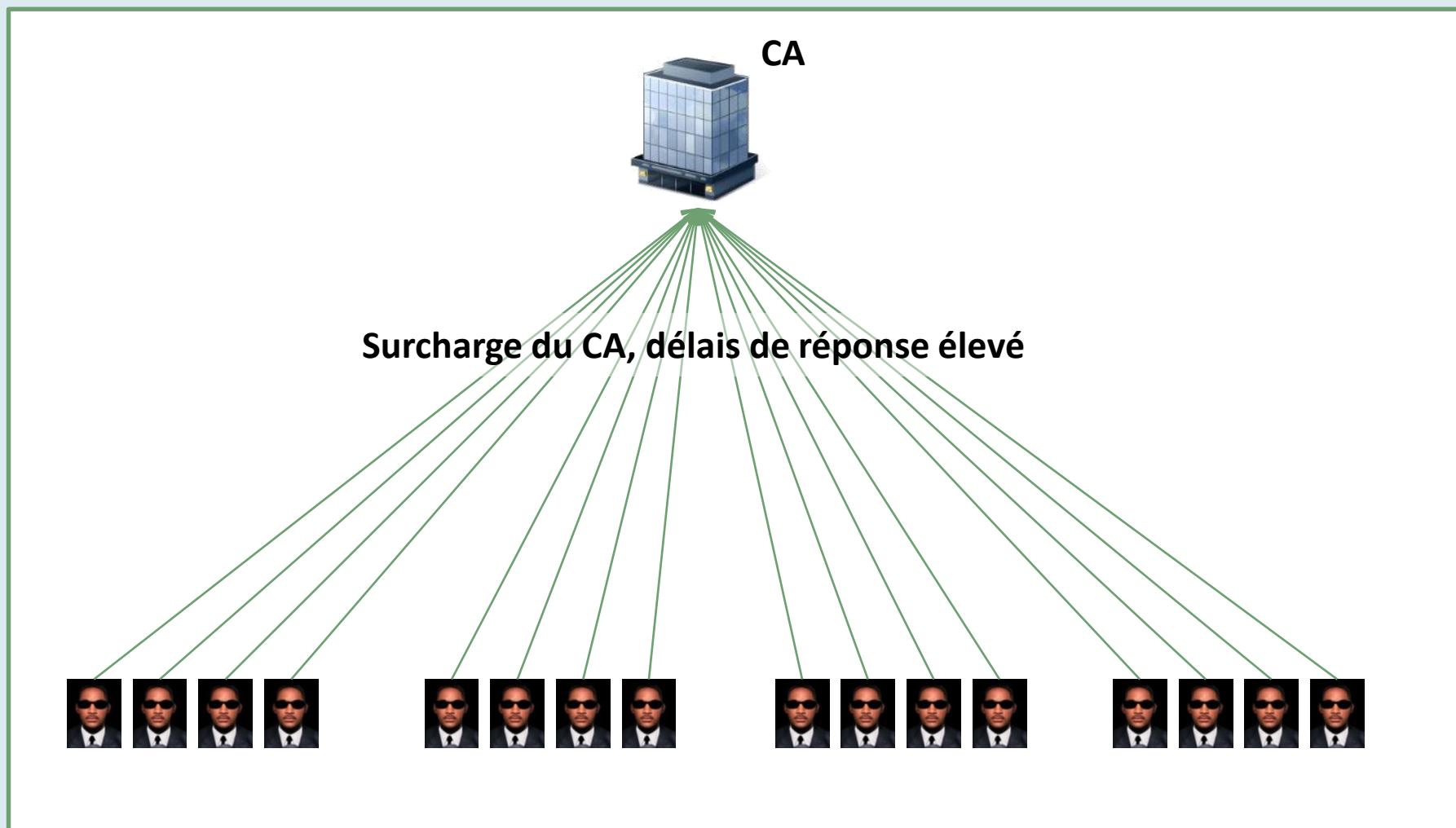
Issuer CA<sub>2</sub>

Est certifié par

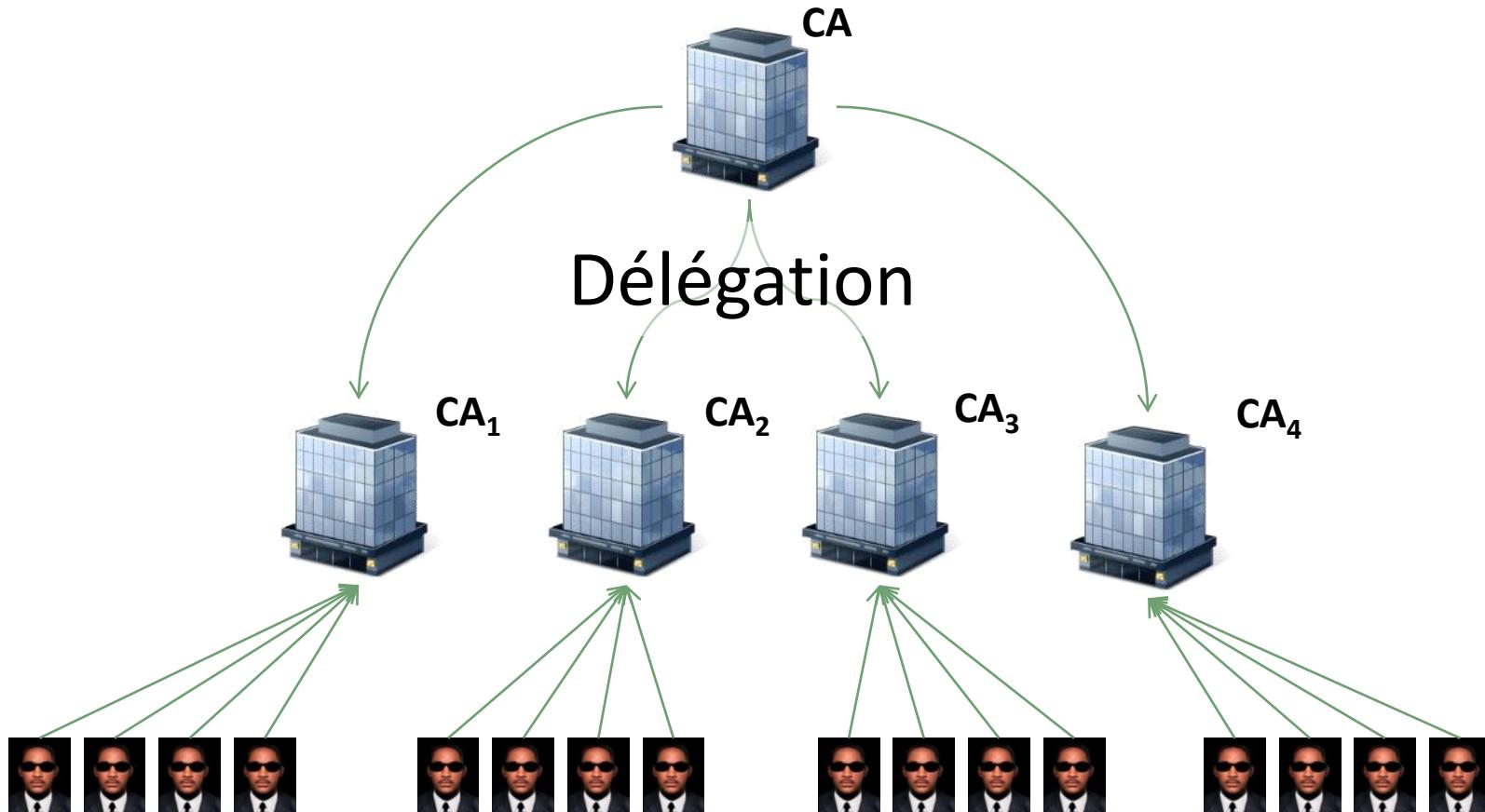


Issuer CA<sub>1</sub>

- PKI – Public Key Infrastructure: Délégation



- PKI – Public Key Infrastructure: Délégation



## • PKI – Public Key Infrastructure: Certification Authority

### Organisme publique

- Pas gratuit
- Verisign, Thawte, Entrust, Baltimore
- Certinomis (la poste, chambre du commerce,...)
- Certplus (Verisign, Matra, France Telecom, Gemplus)  
→ Reconnaissance externe, internationale



### Locale privée

- Gestion de sa propre autorité de certification
- Périmètre de reconnaissance limitée
- Flexibilité de gestion
- Openssl, OpenCa, IDX-PKI , iPlanet Certificate Manager server

## • PKI – Standard X509

### Norme

- ITU-T X.509 ou ISO/IEC 9594-8
- V3 actuelle (v1 1988, v2 1993,v3 1996)
- [RFC 2693 - SPKI Certificate Theory](#)  
→2.2 The X.500 Plan and X.509
- Utilisation de X500 pour le format de nommage issuer,subject



→ Description du format du certificat

## • PKI – Certificat X509

Version (v1=0, v2=1, v3=3)	
Serial number	
Signature algorithm ID	
Issuer Name	
Validity period (Start and expiry dates/times)	
Subject Name	
Subject public key info (Algorithm ID and public key value)	v2
Issuer unique ID	v2
Subject unique ID	v3
Extensions (Type, Critical/Non-crit. Field value)	

Data:

Version: 3 (0x2)

Serial Number: 1 (0x1)

Signature Algorithm: sha1WithRSAEncryption

Issuer: C=FR, ST=Rhone\_Alpes, L=Villeurbanne, O=INSA-LYON,  
OU=Dept Telecom,  
CN=CA/emailAddress=mitsuco26@hotmail.com

Validity

Not Before: Jun 9 08:43:11 2011 GMT

Not After : May 9 08:43:11 2013 GMT

Subject: C=FR, ST=Rhone\_Alpes, L=Villeurbanne, O=INSA-LYON,  
OU=Dept Telecom, CN=serveur  
radius/emailAddress=mitsuco26@hotmail.com

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public Key: (1024 bit)

Modulus (1024 bit):

00:b8:d1:ce:aa:e7:36:07:7f:46:5d:15:8d:24:25:  
a7:2b:08:7d:5d:2c:78:21:94:8d:f0:c3:99:dd:d9:  
18:8d:7d:89:5c:7a:43:b8:a5:4c:2c:69:db:49:4b:  
e1:ea:9f:83:59:53:6b:6f:da:9e:5a:d3:ac:46:2f:  
33:21:50:ac:f3:cc:c2:27:6e:e2:f2:d4:50:4d:fb:  
f1:15:4f:3e:60:9b:07:6a:6c:65:17:bd:7c:c2:f7:  
a1:d5:25:2f:23:35:39:d1:1f:ff:66:4e:ff:d6:7b:  
04:50:e0:12:6e:71:7e:f3:bf:01:3a:d2:29:4a:bd:  
7d:e1:89:9c:bf:1e:4a:60:99  
Exponent: 65537 (0x10001)

Signature
Algorithm ID and Signature value

Contenu

Exemple (1/2)

## • PKI – Certificat X509

Version (v1=0, v2=1, v3=3)	
Serial number	
Signature algorithm ID	
Issuer Name	
Validity period (Start and expiry dates/times)	
Subject Name	
Subject public key info (Algorithm ID and public key value)	
Issuer unique ID	v2
Subject unique ID	v2
Extensions (Type, Critical/Non-crit. Field value)	v3

Signature	
Algorithm ID and Signature value	

X509v3 extensions:  
X509v3 Basic Constraints: critical  
CA:FALSE  
X509v3 Authority Key Identifier:  
keyid:30:5B:05:AA:6E:D3:AE:2D:CD:45:25:05:0A:1F:A0:68:62:E5:67:7  
X509v3 Subject Key Identifier:  
54:52:EF:F4:94:39:18:5E:0A:9D:51:5C:AD:01:39:35:78:39:6F:35  
X509v3 Key Usage:  
Digital Signature, Key Encipherment  
X509v3 Extended Key Usage:  
TLS Web Server Authentication, TLS Web Client  
Authentication  
Netscape Cert Type: SSL Server  
Netscape Comment: Certificat delivre par Dept Telecom  
Signature Algorithm: sha1WithRSAEncryption  
14:d2:ca:7d:66:5e:73:50:e3:28:14:30:cc:8c:ce:29:a8:d0:  
2c:fc:bd:ed:55:8c:60:43:c4:dc:1b:c9:6c:ef:59:ae:a8:54:  
e7:fa:e0:16:3b:2e:27:80:97:3c:f2:35:82:eb:4d:b3:33:ee:  
19:78:7e:f2:51:be:75:5f:78:32:23:65:9e:7f:f8:65:41:90:  
9c:41:6e:5d:5a:8c:94:52:06:e8:5c:b5:c1:d2:35:8d:90:37:  
1d:50:1e:7e:91:2b:67:b0:bf:c3:94:8e:0a:f5:54:3d:57:7b:

Contenu

Exemple (2/2)

## • PKI – Liste de révocation X509

Version (v1=0, v2=1, v3=3)
Signature algorithm ID
Issuer Name
This update date/time
Next update date/time
Revoked certificate
Certificate serial number
Revoked certificate
Certificate serial number
Revocation date
CRL entry extensions
...
Revoked certificate
Certificate serial number
Revocation date
CRL entry extensions
CRL extensions

Signature  
Algorithm ID and Signature value

Contenu

Certificate Revocation List (CRL):

Version 1 (0x0)

Signature Algorithm: md5WithRSAEncryption

Issuer: /C=FRL=Paris/O=Hervé E9 Schauer Consultants  
/OU=Certificate Authority/CN=HSC CA/Email=ca@hsc.fr

Last Update: Aug 26 12:13:35 1999 GMT

Next Update: Sep 25 12:13:35 1999 GMT

Revoked Certificates:

Serial Number: 07

Revocation Date: Aug 26 12:12:31 1999 GMT

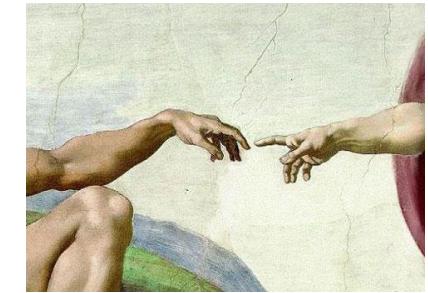
Signature Algorithm: md5WithRSAEncryption  
c4:92:09:bd:ca:9f:cd:56:bd:ef:05:85:f7:b8:01:a6:f5:69:  
...

Exemple

- **PKI – Standard Public Key Infrastructure X.509 (PKIX)**

- Norme

- IETF 1995
    - Objectif: développer une PKI pour internet basée sur les standards X.509, IPKI (Internet PKI)
    - RFC2459



- Composants

- Protocoles d'exploitations (RFC2559, RFC2585)
      - Distribution des certificats et de listes de révocation
      - Utilisation de procédures basées sur LDAP, HTTP, FTP, x500
    - Protocol de gestion (RFC 2510)
      - Communication entre les composants de la PKI
    - Règles d'usage et bonnes pratiques (RFC 2527)

<http://www.hsc.fr/ressources/presentations/pki/img14.htm>

- **PKI – Public Key Cryptography Standard (PKCS)**

- Présentation
    - Développé par RSA devenu un standard de-facto

- Eléments du standard
    - PKCS #1: RSA Encryption
    - PKCS #5: Password-Based Cryptography
    - PKCS #6: Extended-Certificate Syntax
    - PKCS #7: Cryptographic Message Syntax
    - PKCS #8: Private-Key Info Syntax
    - PKCS #9: Selected Attributes Types
    - PKCS #10: Certification Request Syntax
    - PKCS #12: Personal info Exchange Syntax
    - PKCS #13: Elliptic Curve Crypto. Standard
    - PKCS #15 Cryptographic Token Information Format Standard



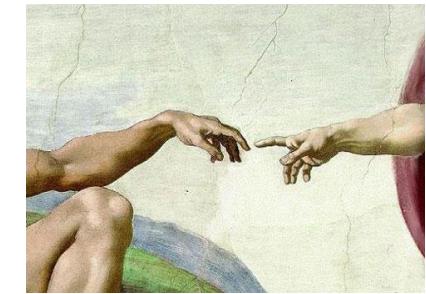
- **PKI – Format de fichiers et contenu de certificat**

<b>Certifikat</b>	X509 DER, PEM (*.cer, *.pem) PKCS#7 DER (*.p7b) PKCS#12 DER (*.p12, *.pfx)
<b>Clé privée</b>	PKCS#1 PEM (*.key, *.pem) PKCS#8 DER (*.key, *.pem) PKCS#12 DER (*.p12, *.pfx)
<b>CRL : Liste de révocation</b>	X509 DER, PEM (*.crl, *.pem) PKCS#7 DER (*.p7b)
<b>Certifikat échange P à Pair</b>	X509 DER, PEM (*.ccp, *.pem)
<b>Certifikat de requête de signature</b>	PKCS#10 DER, PEM (*.crl, *.pem)

- **PKI – Simple Public Key Infrastructure (SPKI)**

- Présentation

- IETF 1996
    - Objectif: définir une infrastructure à clés publiques et format de certificats (propre IETF, simples, adaptés aux applications web)



- Plus de notion "*un certificat lie une clé à une identité*" pour une notion plus générale "*un certificat attribut des permissions au possesseur d'une clé*"

- Contenu des certificats

- Identité du porteur (owner)
    - Identité du certifier émetteur (issuer)
    - Période de validité (début-fin)
    - Permission de délégation
    - Autorisation

## • PKI – Les autres définitions

- ❑ XKMS XML Key Management Spécification
  - PKI par échange de messages XML via SOAP
- ❑ API JAVA: Java Cryptography Extension (JCE), Java Secure Socket Extension (JSSE)
- ❑ API Java for XML:
  - ❑ JSR 104 XML Trust Service API
  - ❑ JSR 105 XML Digital Signature API
  - ❑ JSR 106 XML Digital Encryption API
- ❑ Carte à puce
  - ❑ Dédiées (GemSafe, CryptoSafe)
  - ❑ Card Applet PKI pour JavaCard
  - ❑ OCF (PKICardService)
  - ❑ CMS (Card Management Service): interface avec CA, extension à XKMS (Entrust)



- **PKI – Public Key Infrastructure: A vous de jouer**

1. A quelle problématique réponds les PKI



2. Quelles objectifs de sécurité permettent d'assurer les certificats?



- PKI – Public Key Infrastructure: Le contenus de vos postes

## Demo

# Sécurité Internet

- └ HTTPS/Secure HTTP
- └ Secure Electronic Transaction
- └ SSH

## • Sécurité Internet – HTTPS

- Besoin comment sécuriser des communications sur internet sécurisée ?  
→ utilisation de http over SSL/TLS → HTTPS
- Nouveau port de communication 443 (http port 80)
- SSL utilise le chiffrement asymétrique afin de fournir
  - Chiffrement de données (via une clé de session)
  - L'authentification du serveur (et celle du client optionnelle)
  - L'intégrité des données



## • Sécurité Internet – HTTPS

### ❑ Pourquoi toutes les communications ne sont pas en HTTPS ?

- Consommation de ressources (lié au chiffrement asymétrique)
- Certificat du server web (coût)
- Communications plus lentes (liées au chiffrement)



- Sécurité Internet – HTTPS

**Le maillon faible**



**VOUS !**

## • Sécurité Internet – HTTPS

The screenshot shows a dual-pane interface. On the left, a yellow warning icon with a shield and a crossed-out padlock is displayed, accompanied by the text "Échec de la connexion sécurisée". Below this, it says "localhost.com utilise un certificat de sécurité non valide". A detailed error message follows: "Le certificat n'est valide que pour sécuriser le site https://www.iceweasel.org". It also includes a code snippet "(Code d'erreur : ssl\_error\_bad\_cert\_domain)". A bulleted list provides troubleshooting steps:

- Ceci peut-être dû à un problème de configuration du serveur.
- Si vous êtes déjà connecté avec succès, il peut-être temporaire et vous pouvez essayer de nouveau.

A yellow callout box at the bottom left states: "Vous ne devez pas ajouter d'exception si vous ne connaissez pas la personne à laquelle vous n'avez pas totalement confiance ou si vous recevez un avertissement pour ce serveur." It contains two buttons: "Quitter cette page" and "Ajouter une exception".

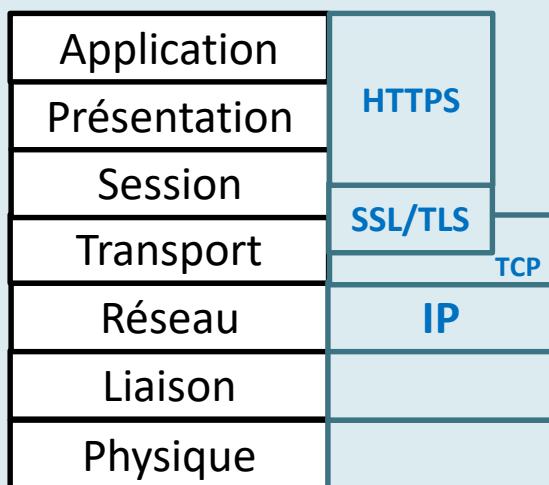
The right pane is a modal dialog titled "Ajout d'une exception de sécurité". It contains the following text:  
"Vous êtes en train de passer outre la façon dont Iceweasel identifie ce site.  
Les banques, magasins et autres sites Web publics légitimes ne vous demanderont pas de faire cela."

It has sections for "Serveur" (Address: https://localhost.com/, Obtain the certificate button) and "État du certificat" (Information invalides, Voir... button). A "Mauvais site" section notes: "Le certificat appartient à un site différent, ce qui pourrait indiquer un vol d'identité."

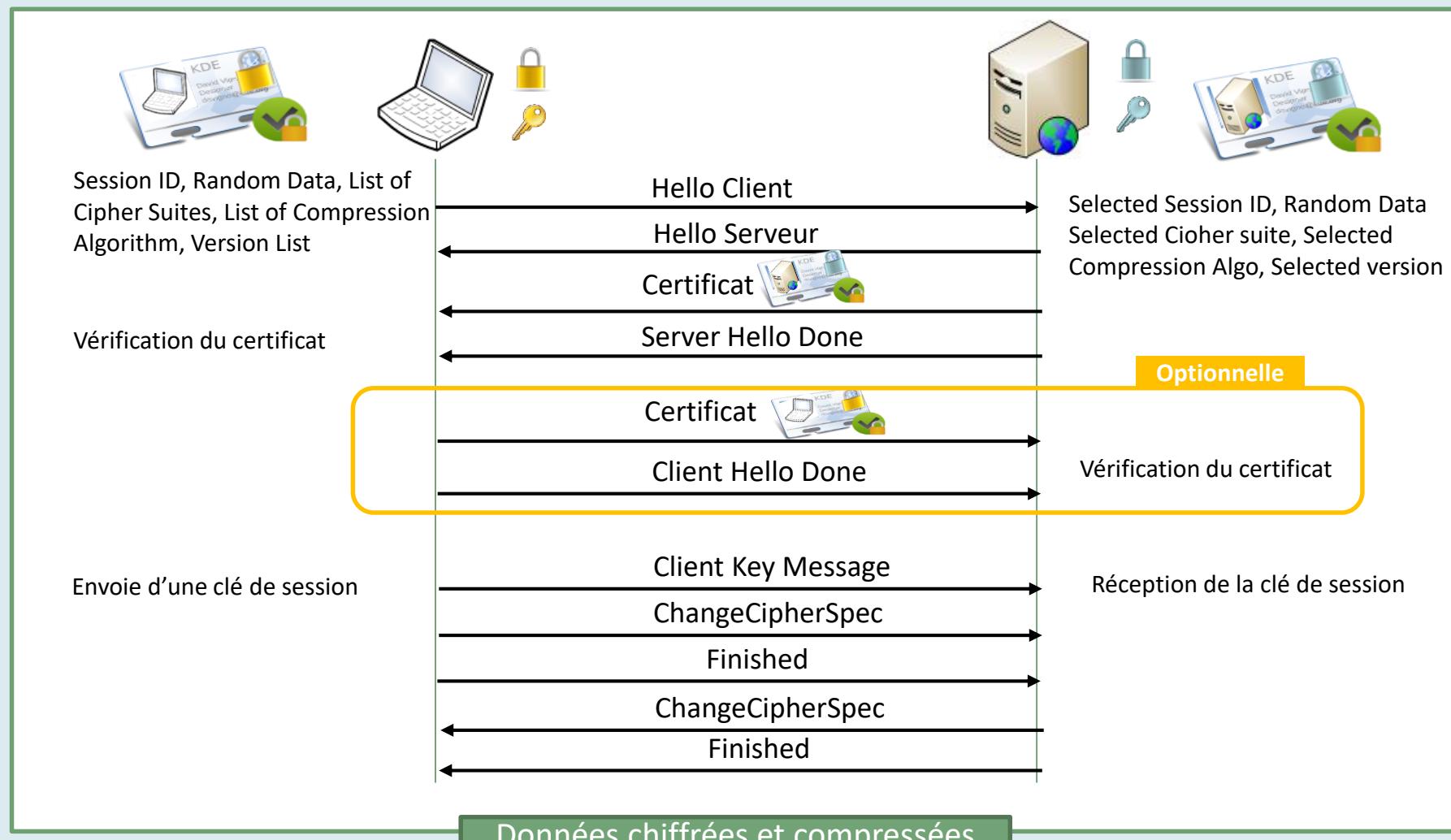
At the bottom, there is a checkbox for "Conserver cette exception de façon permanente" (checked), a "Confirmer l'exception de sécurité" button, and an "Annuler" button.

## • Sécurité Internet – HTTPS

- ❑ TLS Transport Layer Security autrefois (SSL Secure Socket Layer)



## • Sécurité Internet – HTTPS – SSL/TLS



- Sécurité Internet – HTTPS – SSL/TLS

## Demo wireshark

# Sécurité Internet

- └ HTTPS/Secure HTTP
- └ Secure Electronic Transaction
- └ SSH

## • Sécurité Internet – Secure Electronic Transaction

- ❑ 1996 VISA/MasterCard
- ❑ Objectif: Sécuriser les transactions bancaires sur un réseau non sécurisé
- ❑ Repose essentiellement sur le chiffrement asymétrique et la signature numérique
- ❑ Permet d'assurer l'authenticité des utilisateurs, la confidentialité de l'information et l'intégrité du paiement

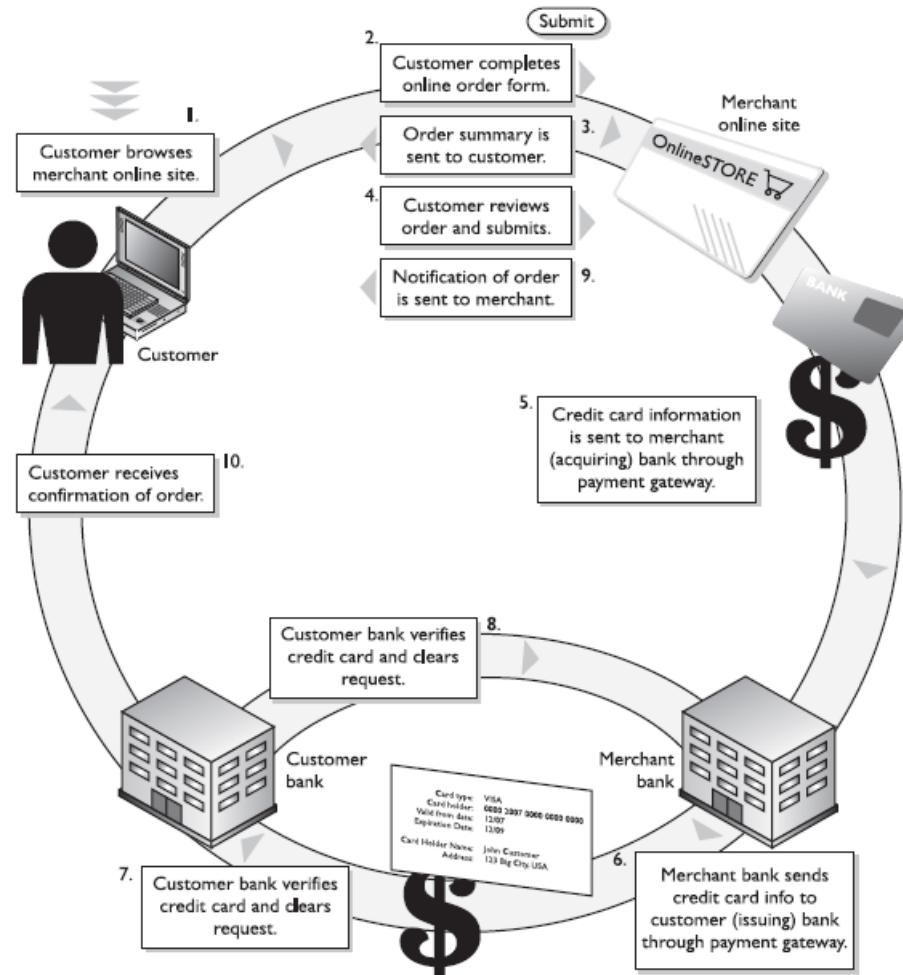


## • Sécurité Internet – Secure Electronic Transaction

- Les entités
  - Banque du demandeur (issuer)
  - L'utilisateur de la carte de crédit (Cardholder)
  - Marchand (merchant)
  - Banque du marchand (Acquierer)
  - Passerelle de paiement (Payment gateway)



## • Sécurité Internet – Secure Electronic Transaction

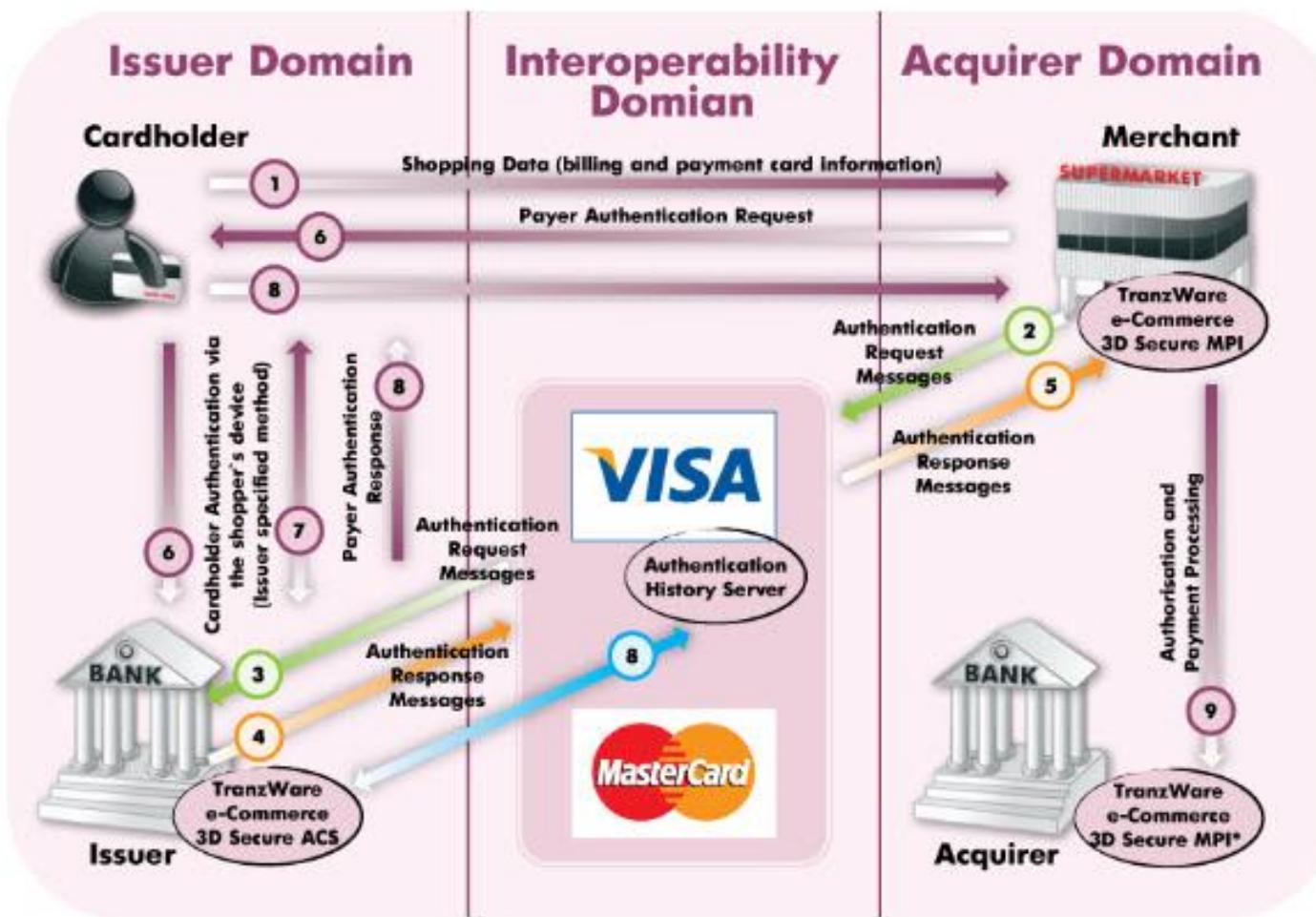


## • Sécurité Internet – 3D-Secure

- ❑ VISA/MasterCard
- ❑ Objectif: Autorisation financière avec authentification en ligne
- ❑ Actuellement le système de paiement le plus utilisé



- Sécurité Internet – 3D Secure



# Sécurité Internet

- └ HTTPS/Secure HTTP
- └ Secure Electronic Transaction
- └ SSH

## • Sécurité Internet – Secure Shell (SSH)

- ❑ Protocole de communication V1 (1995), V2 (2006)
- ❑ Utilisation du port 22
- ❑ Mode client serveur
- ❑ Redirection de port (forwarding)
  
- ❑ Objectif
  - ❑ Chiffrer et compresser un canal de communication
  - ❑ Ensemble d'outils permettant de remplacer des outils de connexions non sécurisés (rpc, rlogin, rsh, telnet)
  - ❑ Mots de passe et données chiffrées lors de la communication



- **Sécurité Internet – Secure Shell (SSH)**

- Exemple d'utilisation d'algorithme sous linux

- Chiffrement asymétrique

- RSA, DSA

- Chiffrement symétrique

- 3DES, Blowfish, AES..

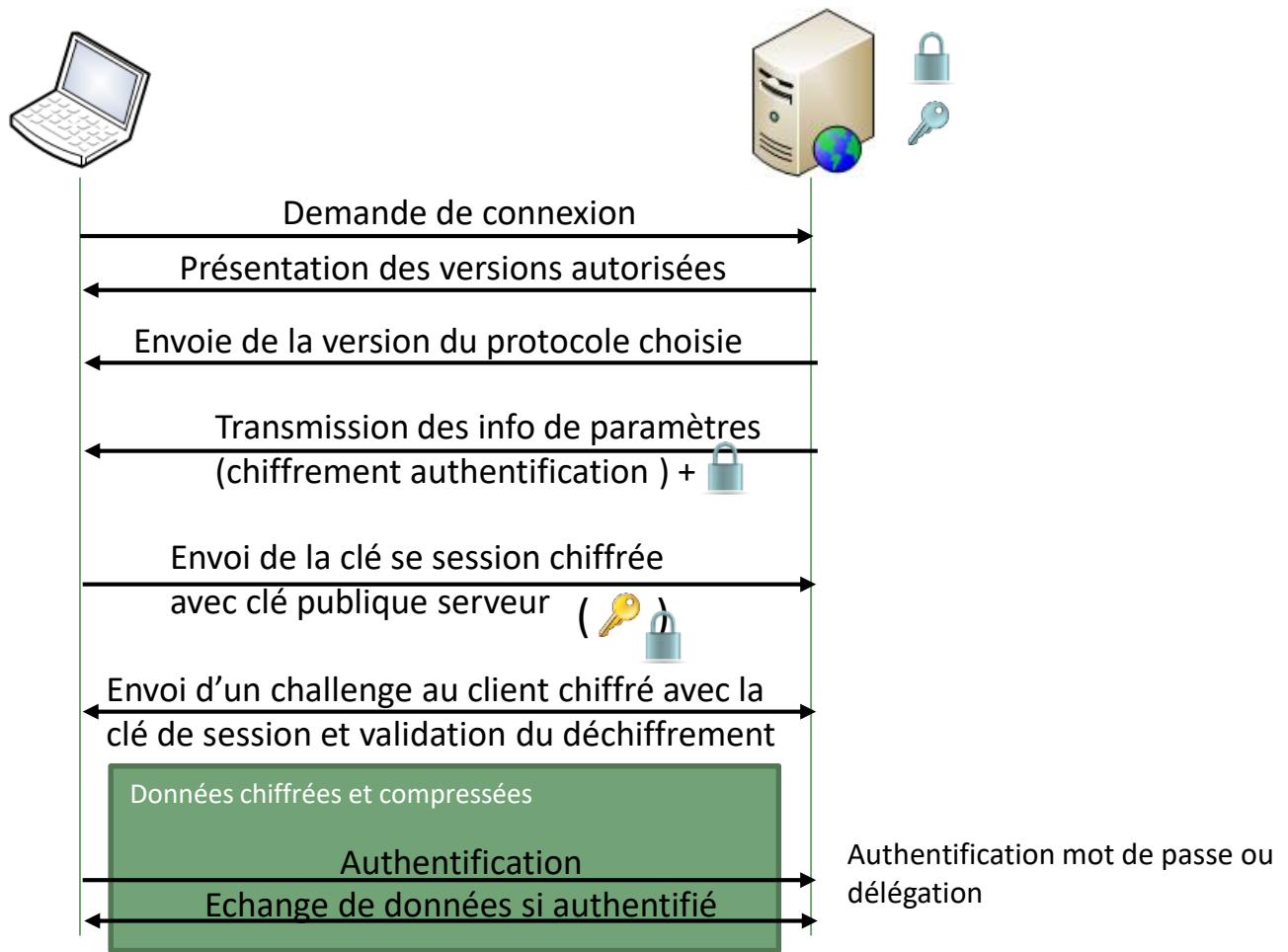
- Gestion de l'authentification

- Possibilité d'activer le support de l'interface PAM

- (Pluggable Authentication Modules)



## • Sécurité Internet – SSH



# Conclusion

---

# Questions ?

---