Chapitre 5:Chiffrement asymétrique (a clé publique)

Kahya noudjoud

Chiffrement à clé publique

Pour réaliser la confidentialité des données, il existe deux manières :

1. En limitant l'accès (procédure de contrôle d'accès)
2. En rendant les données incompréhensibles aux personnes non-autorisés (chiffrement/déchiffrement)

Chiffrement symétrique à clé secrète (chiffrement conventionnel): Le chiffrement et le déchiffrement d'un texte reposent sur la même clé c-à-d l'émetteur et le récepteur doivent posséder alors la même clé secrète pour correspondre confidentiellement

Exemple: DES, RC4, RC5, Blowfish, IDEA, AES, ….

Chiffrement à clé publique

- Le problème 1: Dans le cas d'un système de chiffrement symétrique, l'émetteur et le récepteur doivent posséder et utiliser la même clé secrète pour rendre confidentielles des données échangées, ceci pose le problème de la diffusion des clés secrètes.
- Le problème 2: Chaque entité doit disposer d'autant de clés secrètes qu'elle a d'interlocuteurs (problème de gestion autant de clés différentes). Considérer N utilisateurs (une clé pour chaque couple) !!!!

Ce type de chiffrement n'est pas pratique dans le cas de l'Internet où les entités communicantes ne se connaissent pas.

Chiffrement à clé publique

- Un chiffrement à clé publique (asymétrique) comprend une entité qui a pour rôle de générer pour chaque interlocuteur, qui désire communiquer des données confidentielles, un couple de clés calculées l'une par rapport à l'autre et indissociables.

Il s'agit d'un couple unique de clés constitué de deux parties complémentaires et d'usage différent.

- Une partie visible = clé publique connue de tous.
- Une partie secrète = clé privée doit être confidentielle et traitée comme un secret.

Chiffrement à clé publique

- Chacun peut connaître la clé publique des partenaires de la communication en la demandant à l'autorité de gestion des clés. Considérer N utilisateurs (une paire de clés pour chaque utilisateur) !!!!
- De manière générale, on chiffre un message avec la clé publique du destinataire qu'il déchiffrera avec sa clé privée.
- Les principaux algorithmes de chiffrement à clé publique, utilisent le plus souvent des clés de longueur variant de 512 à 1024 bits, voire 2048 bits. Exemple: Diffie-Hellman; RSA , El-Gamal
- Inconvénient Vitesse : lent

Chiffrement à clé publique

Pour le chiffrement asymétrique (a clé public) le calcul de la clé de déchiffrement ayant seulement connaissance de {algorithme de chiffrement + clé} impossible sur ordinateur.

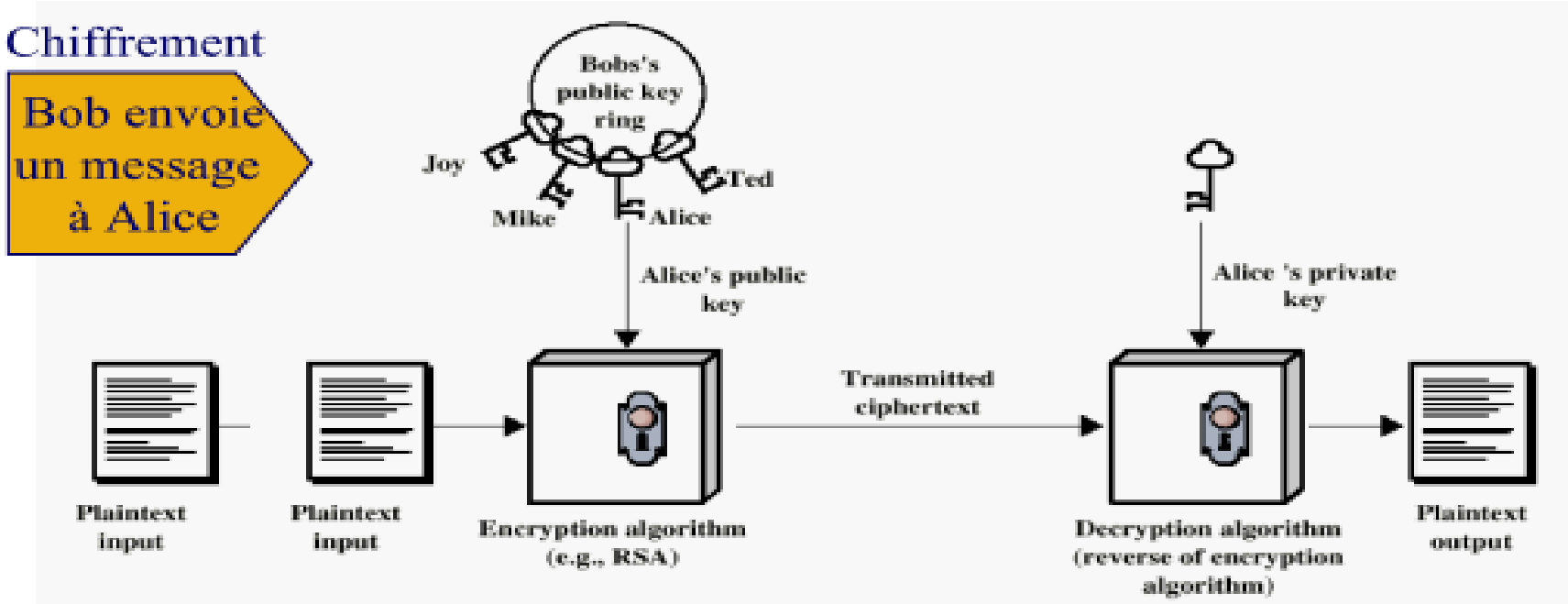
Possibilité de permuter les deux clés publique/privée (spécifique au RSA), une pour chiffrer l'autre pour déchiffrer.

Chiffrement à clé publique

- Chaque bout du système génère une paire de clés (publique /privée)
- Chaque système enregistre sa clé publique (clé de chiffrement) dans un registre publique (fichier)
- Il existe deux méthodes de chiffrement:

Méthode 1:

- Bob envoie un message à Alice, Bob chiffre le message avec la clé publique de Alice
- Alice reçoit le message chiffré et pour le déchiffrer elle se sert de sa clé privé

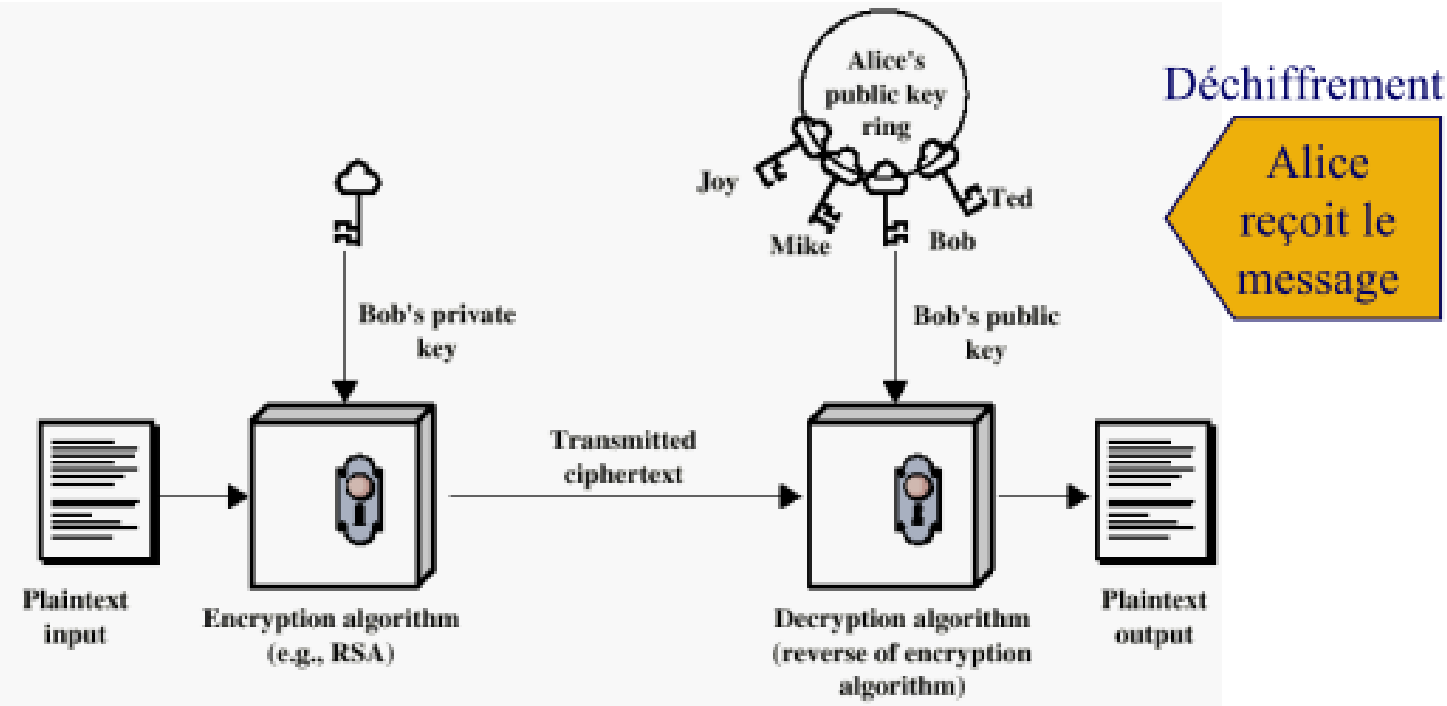


Chiffrement à clé publique

- Méthode 2:

➢ Bob envoie un message à Alice, Bob chiffre (signe) le message avec sa clé privée.

➢ Alice reçoit le message chiffré et pour le déchiffrer elle se sert de la clé publique de Bob



Chiffrement à clé publique

- Afin de permettre à ses correspondants de lui envoyer des messages cryptés, un participant doit calculer sa clé publique qu'il diffusera à tous le monde dans un annuaire, il prépare également sa clé privée qu'il sera seul à connaitre.
- Tous les participants ont accès aux clés publiques
- Les clés privées sont générées localement par chaque participant
- À tout moment un participant peut changer sa clé privée par conséquence il doit enregistrer sa clé publique conjointe.

Chiffrement à clé publique

- Types de clés:

✓ Clé secrète : Km (conventionnelle) m : modification; Ks: clé de session

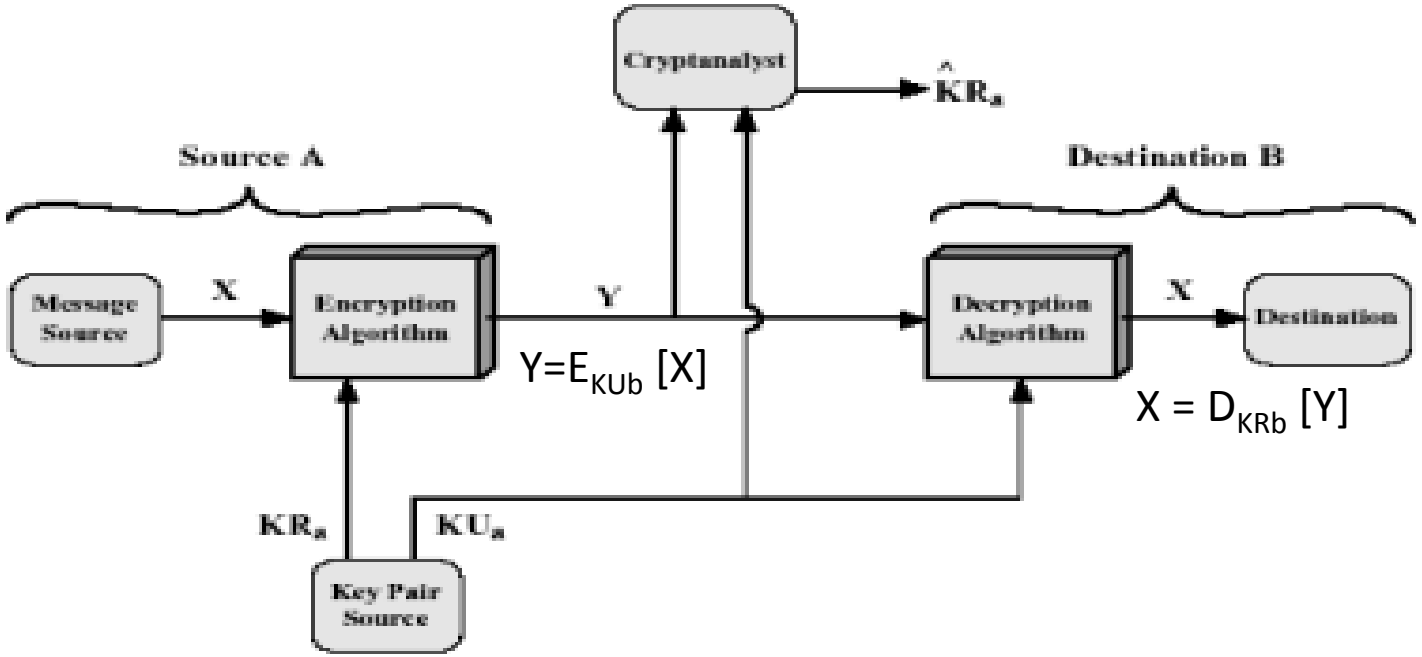
✓ Clé publique: KUa pour l'utilisateur A

✓ Clé privée correspondante : Kra (pour l'utilisateur A)

- Fonctionnement

1. La source A envoie un texte clair X = [X 1 , X 2 , …, X M] vers le destinataire B
2. B génère une paire de clé : KUb (accessible à A) et KRb (connue seulement par B)
3. A forme le message chiffré (Y) par clé publique de B (KUb): $Y=E_{KUb}[X]$
4. Le récepteur B correspondant possède la clé privée capable de déchiffrer Y. $X = D_{KRb}[Y]$

Chiffrement à clé publique



Chiffrement à clé publique

Pour le chiffrement asymétrique:

1. Pour chaque participant exp: Bob, il est facile par ordinateur de générer la paire de clés (clé publique KUb , clé privée KRb
2. Facile à la source de produire un texet X et de générer le texte chiffrée : $Y=E_{KUb}(X)$
3. Facile à la destination de déchiffrer le texte chiffré avec la clé privée: $X=D_{KRb}(Y)=D_{KRb}(E_{KUb}(X))$,
4. Non faisable de déterminer par ordinateur la clé privée de Bob (KRb) connaissant sa clé publique (KUb)
5. Non faisable par ordinateur de recouvrir le message X, connaissant KUb et le texte chiffré Y

Chiffrement à clé publique

6. L'une ou l'autre des deux clés peut être utilisée pour chiffrer alors la second pour déchiffrer:

$X=D_{KRb}(Y)=D_{KRb}(E_{KUb}(X))$ On chiffre avec KU du destinataire et le message ne sera déchiffré que par le vrai destinataire.

$X=D_{KUb}(Y)=D_{KUb}(E_{KRb}(X))$ On chiffre(Signe) avec KR du source et le message ne sera déchiffré que par la clé publique de destinataire. dans ce cas en produit la signature digitale, de cette manière on valide le contenu du message et le auteur de ce message

7. Pour fournir la confidentialité et l'authenticité du message: On chiffre avec KR de l'expéditeur (Alice) et on chiffre le résultat avec la clé KU du destinataire (Bob),

$Y=E_{KUb}(E_{KRa}(X))$ pour le déchiffrement $X=D_{KRb}(D_{KUa}(Y))$

Chiffrement à clé publique

Applications des chiffrements à clé publique

- Chiffrement /déchiffrement : l'expéditeur encrypte le message avec KU du destinataire.
- Signature numérique : l'expéditeur signe le message avec sa clé K.
- Échange de clés : plusieurs approche pour que deux entités coopérantes échangent les clés (exp: diffie hellmen).

Chiffrement à clé publique

Algorithmes	Chiffrement /déchiffrement	Signature numérique	Échange de clés
RSA	Oui	Oui	Oui
DSS	Non	Oui	Non
Diffie-Hellmen	Non	Non	Oui

Chiffrement à clé publique

Exemple de chiffrement a clé publique (Asymétrique)

- Le plus connu d'entre eux : RSA (Rivest, Shamir et Adleman en 1977). Il permet le chiffrement et la signature.
- Le protocole d'échanges de clés Diffie-Hellman (en 1976), protocole à l'origine de la cryptographie moderne.
- La cryptographie sur les courbes elliptiques (Koblitz et Miller en 1985) : ECC (Elliptic Curve Cryptography).
- L'algorithme de signature numérique DSA (Digital Signature Algorithm) proposé par le NIST (National Institute of Standards and Technology) en 1991.
- Et bien d'autres encore...

Algorithme RSA

- L'algorithme RSA est sans doute le plus utilisé des systèmes à clé publique actuellement; il a été publié en 1978 par Ronald L. Rivest, Adi Shamir et Leonard M. Adleman.
- Le RSA est le plus populaire, le plus facile à comprendre et à réaliser de tous les systèmes à clé publique.
- Il a résisté depuis des années de cryptanalyse intensive.
- Il peut être utilisé pour chiffrement, la signature numérique et l'échange des clés secrètes.
- Il nécessite des clés d'au moins 1024 bits pour obtenir une sécurisation satisfaisante.

Algorithme RSA

- **Étape 1: Création des clés**
- Choisir au hasard deux nombres premiers p et q (les deux étant plus grands que 10^{100}) et calcule n=p*q.
- Choisir au hasard e tel que :

$$\begin{cases} 1<e<\phi(n)=(p-1)(q-1) \\ p\gcd(e,\phi(n))=1 \end{cases}$$

- Φ(n): le nombre d'entiers inférieurs à n, premiers avec e
- Calculer l'entier d pour inverser la fonction de chiffrement tel que :

$$\begin{cases} 1<d<\phi(n) \\ ed=1 \, mod \, \phi(n) \end{cases}$$

Algorithme RSA

- La clé publique est (e, n)
- La clé secrète est (d, n)

Etape 2: Chiffrement du message

Bob récupère la clé publique (e, n) d'Alice et souhaite lui envoyer un message m.

Bob calcule : $C= (m^e)mod\ n$

Algorithme RSA

Étape 3: Déchiffrement du message

- Lorsque Alice reçoit le chiffré C, elle calcule C^d , et récupère ainsi le message m puisque :

$m= (C)^d \bmod n = (m^e)^d \bmod n= m^{ed} \bmod n=m \bmod n$

- (d, n) est la clé privée d'Alice.

Algorithme RSA

- Exemple :

1- Choisir deux nombres premiers $p=7$ et $q=17$

2- Calculer $n = pq = 119$

3- Calculer $\phi(n) =(p-1)(q-1)=96$

4- Choisir e premier avec $\phi(n)$ et $e < \phi(n)$ par exemple $e=5$

5- Déterminer d tq $de =1 \bmod 96$ et $d < 96$ on trouve $d=77$

Clé publique $(e,n) = (5, 119)$ et la clé privée $(d, n)= (77, 119)$

Pour chiffrer 19; $c=(m^e)\bmod n = (19)^5 \bmod 119=66$

Pour déchiffrer 66; $m= (C)^d \bmod n$ $(66)^{77} \bmod 119=19$

- Chiffrement du message 'HELLO'.
- a) On prend le code ASCII de chaque caractère et on les met bout à bout: m = 7269767679
- b) Il faut découper le message en blocs qui comportent moins de chiffres que n.
- c) n comporte 4 chiffres, on découpe notre message en blocs de 3 chiffres: 726 976 767 900 (on complète avec des zéros pour avoir une longueur multiple de 3 bits)

- On chiffre chacun de ces blocs :
- 726^71 mod 1073 = 436
- 976^71 mod 1073 = 822
- 767^71 mod 1073 = 825
- 900^71 mod 1073 = 552
- Le message chiffré est donc: 436 822 825 552.

Attaques

- Mathématique Attaques: Factorisation un Grand nombre , retrouver p et q en factoriser le modulo N
  - ➢Méthode Fermat
  - ➢Méthode Euler
  - ➢Méthode Pollard’s Rho
- Implémentation Attaque: Obtention de physique implémentation d’un système cryptographie,
  - ➢Timing Attack

Algorithme RSA

- Le principe de l'algorithme RSA repose sur le fait qu'il est très difficile et très long de factoriser un très grand nombre en deux facteurs premiers, pour cela il faut procéder de la manière inverse : générer les deux nombres premiers (p et q), puis les multiplier pour générer le nombre n.
- le **RSA** est le système de cryptographie asymétrique le plus utilisé actuellement dans les cartes bleues, les logiciels, les cartes bancaires, les messageries, ...

Algorithme Diffie Hellman

- C’est une méthode pratique pour l’échange public d’une clef secrète (ou de session). La méthode de Diffie-Hellman (DH) est basée sur l’élévation à une puissance dans un champ fini, ce qui est facile à calculer.

- Principe
- Soient A et B, les deux parties de la communication.

Algorithme Diffie Hellman

- Avant de commencer, Alice et Bob choisissent un nombre premier p et un générateur g mod p, $(1 < g < p - 1)$ qui peuvent être publics.

Alice génère aléatoirement un nombre a entre 1 et p − 1.
Alice calcule $A = g^a \bmod p$.

Bob génère aléatoirement b entre 1 et p − 1.
Bob calcule $B = g^b \bmod p$.

1. Alice −→ Bob : A
2. Bob −→ Alice : B

Alice calcule $K = B^a \bmod p$

Bob calcule $K = A^b \bmod p$.

Alice et Bob partagent maintenant la même clé secrète K, car $B^a = A^b = g^{ab} \bmod p$.

La sécurité repose sur la difficulté de calculer a et b, c'est-à-dire de calculer le logarithme discret mod p.

Algorithme Diffie Hellman

Exemple :

Alice et Bob souhaitent échanger une clé.

1. Soient les valeurs $p = 353$ et $g = 3$ rendues publiques.

2. Ils sélectionnent chacun une valeur aléatoire : Alice choisit $a = 97$, Bob choisit $b = 233$.

3. Ils calculent la clé publique : $A = 3^{97} \bmod 353 = 40$ (pour Alice) et $B = 3^{233} \bmod 353 = 248$ (pour Bob).

4. Et calculent finalement la clé de session : $K_{AB} = (B)^a \bmod 353 = 248^{97} \bmod 353 = 160$ (pour Alice) et $K_{AB} = (A)^b \bmod 353 = 40^{233} \bmod 353 = 160$ (pour Bob).

- Cette découverte de Diffie et Hellman est une vraie révolution dans l'histoire de la cryptographie.
- Le problème de l'échange des clés est en effet résolu.
- Mais l’inconvénient de ce protocole : il exige la simultanéité des actions d'Alice et de Bob. Si Alice veut envoyer un e-mail à Bob alors que celui n'est pas connecté, elle ne pourra pas le faire immédiatement. C'est pourquoi ce protocole fut en réalité très vite supplanté par les méthodes de chiffrement à clé publique de type RSA, pour lesquels on met à la disposition de tout le monde une clé publique.
- Toutefois, il est utilisé pour les problèmes d'appariement de deux objets dans la technologie Bluetooth.