Sécurité Informatique

Chapitre 3 :

Cryptographie Asymétrique

(Moderne)

Guellil zouaoui

Problématique

- Alice veut communiquer en toute confidentialité
- avec Bob
  - Sans interaction
  - Sans aucun secret commun
- Diffie et Hellman en 1978 « remarquent » que
  - si les clés de chiffrement et de déchiffrement sont différentes,
  - alors la clé de chiffrement peut être rendue publique
- Notion de cryptographie asymétrique

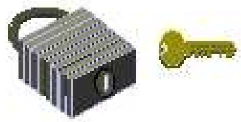
Cryptographie Asymétrique : principe

- Cette technique repose sur le fait que la clé de chiffrement soit différente de la clé de déchiffrement.
- La clé de déchiffrement ne peut pas être calculée à partir de la clé de chiffrement et réciproquement.
- La clé de chiffrement appelée clé publique est destinée à être divulguée,
- La clé de déchiffrement appelée clé privée est gardée secrète.

Cryptographie à clé publique

Etape 1

Fabrication des clés : Bob fabrique une clé publique qui permet de sceller le message dans une boite (ici, le cadenas) et une clé privée qui permet d'ouvrir le cadenas

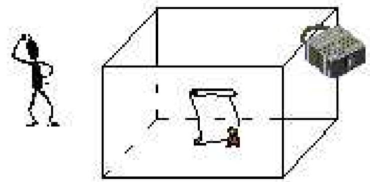


Etape 2

Distribution des clés :
Bob fait parvenir à Alice le cadenas, mais garde la clé pour lui

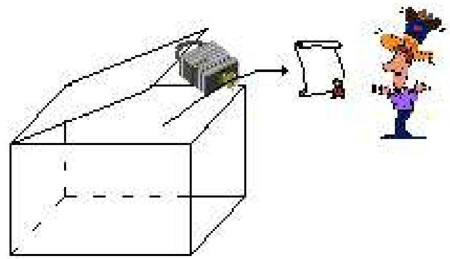
Etape 3

Envoi du message :
Alice met son message dans une boite qu'elle ferme à l'aide du cadenas



Etape 4

Réception du message :
Bob ouvre la boite à l'aide de sa clé et récupère le message. Personne n'a pu l'intercepter car lui seul avait la clé du cadenas.



Source http://www.bibmath.net/crypto/index.php?action=affiche&quoi=moderne/clepub

Cryptographie Asymétrique : Outils de base

- Fonction à sens unique à trappe : facile à calculer, mais difficile à inverser sans la connaissance d'une trappe.
  - factorisation des grands nombres : RSA
  - Logarithme discret : **chiffrement El Gamal**
  - courbes elliptiques: Elliptic Curve Integrated Encryption Scheme (ECIES)

Le cryptosystème RSA

- 1978: Rivest, Shamir Adleman
- Le niveau de sécurité dépend de la difficulté de factoriser des grands nombres.
- Les clé publiques et privées sont des fonctions d'une paire de grands nombres premiers.
- Clef publique = (n, e) ; Clef privee = (n, d) , d calculé à partir de p,q (secrets)
- n produit de p q premiers
- Le chiffrement de x est

$$y= x^e \bmod n$$

- Le déchiffrement de y est

$$x=y^d \bmod n$$

- Afin d’assurer qu’il n’y ait aucune ambiguïté dans la reconstitution de x à travers le module n, il suffit de découper le message en blocs codés par des entiers m qui soient tous <= n -1.

Définition

- Indicatrice d'Euler ϕ(n) est le nombre d’entier positive inferieur à n et premier avec n.
- Exemple : ϕ(8) = 4 car parmi les nombres de 1 à 8, seuls les quatre nombres 1, 3, 5 et 7 sont premiers avec 8,

n	ϕ(n)	condition
p	p-1	p est premier.
p^n	p^n-p^{n-1}	p est premier.
s.t	ϕ(s).ϕ(t)	s et t premier entre eux.
p.q	(p-1)(q-1)	p et q sont premier.

Génération de clef

publique « e » et secrète « d »

- $n = p.q$
- $\phi(n) = (P-1)(q-1)$
- Choisir e tel que **pgcd(**$\phi(n)$,e)=1
- $d.e = 1 \bmod \phi(n)$ ou bien $d = e^{-1} \bmod \phi(n)$ (inversion modulaire)

RSA : Example 1/3

- Choisir deux nombre premier , $p = 7$ et $q = 17$.
- Calculer $n = p \times q = 7 \times 17 = 119$.
- Calculer $\phi(n) = (p-1)(q-1) = 96$.
- Choisir e tel que e et $\phi(n)$ soit premier entre eux et e $<\phi(n)$; $e = 5$.
- Déterminer d tel que $d \times e = 1 \bmod 96$ et $d < 96$. dans ce cas $d = 77$, puisque $77 \times 5 = 385 = 4 \times 96+1$.

RSA : Example 2/3

- Clef publique : n = 119, e = 5; Clef privé : n = 119, d = 77.
- Message M=ali => M= 01;11;08
- X= 011;108
- E(011) = 11^5 mod 119 = 044
- E(108)= 108^5 mod 119 = 075
- C=044;075

RSA : Example 3/3

- Clef publique : n = 119, e = 5; Clef privé : n = 119, d = 77.
- Message crypté C = 044;075
- D(44)= 44^{77} mod 119 = 011
- D(075)= 075^{77} mod 119 = 108

Exponentiation modulaire ($c = b^e \bmod n$) 1/2

- permettre d'évité le dépassement de capacité lors de calcul de m^e.
- Principe : x= a mod n , y= b mod n => x*y = ab mod n

```
function powmod(b, e, n)
 si m = 1 alors retourner 0
 c = 1;
 pour a = 1; a<= e;a++
   c = (c * b) mod n
   fin pour
 retourner c
```

Exemple : $11^5 \bmod 119$

b	e	n	a	c
11	5	119	1	1
11	5	119	1	11
11	5	119	2	2
11	5	119	3	22
11	5	119	4	4
11	5	119	5	44

```
w  le (exp > 0) { if ((exp & 1) > 0) result = (result * base) % m; exp >>= 1; base = (base * base) % m; } return result;
```

Exponentiation modulaire (c= b^e mod n) 2/2

- L'écriture binaire d'un nombre entier $e = \sum_{i=0}^{n-1} a_i . 2^i \; tel \; que a_i \in \{0,1\}$
- Ce qui donne $c = b^e \; mod \; n = b^{\sum_{i=0}^{n-1} a_i . 2^i} \; mod \; n = \prod_{i=0}^{n-1} \left(b^{2^i}\right)^{a_i}$

```
function powmod(b, e, n)
c = 1
while (e> 0) {
   if ((e & 1) > 0) c= (c* b) % n;
   e >>= 1; //décalage à droite d'un bit
   b = (b * b) % n;
}
 return c;
```

Exemple : 11^5 mod 119; $(119)_{10}$ =$(1110111)_2$

b	e	n	c
11	5	1110111	1
11	5	111011	11
11	5	11101	2
11	5	1110	2
11	5	111	22
11	5	11	4
11	5	1	44
11	5	0	44

Sécurité RSA

- Fondée sur la difficulté de trouver deux nombres premiers p and q à partir de leur produit n
- Ainsi RSA est sur a condition qu'on ne puisse pas factoriser de façon efficace le produit de deux grands nombres premiers.
- Cependant, il pourrait y avoir d'autres méthodes pour obtenir de l'information sur le texte clair qui ne nécessitent pas le calcul de d.
- A l'état de nos connaissances, attaquer RSA pourrait être plus facile que factoriser le produit de deux grands nombres premiers.

Date: Mon, 9 May 2005 From: "Thorsten Kleinjung"
Subject: rsa200
We have factored RSA200 by GNFS. The factors are
35324619344027701212726049781984643686711974001976250236493034687761212536794232000585479565280883 49
and
79258699544783330333470858414800596877379758573642199607343303414557678728181521353814093047401854 67
More details will be given later.
F. Bahr, M. Boehm, J. Franke, T. Kleinjung

Recommandations RSA

- Pour garantir une bonne sécurité, il faut respecter certains règles telles que :
  - Ne jamais utiliser de valeur n trop petite,
  - N'utiliser que des clés fortes (p-1 et q-1 ont un grand facteur premier),
  - P et q ne doivent pas avoir le même nombre de chiffres,
  - Ne pas utiliser de petit valeur de n (blocs trop courts) ,
  - Ne pas utiliser de n communs à plusieurs clés,
  - Si (d,n) est compromise ne plus utiliser n.

Le cryptosystème d'ElGamal

- Présenté en 1984 par Taher Elgamal
- Basé sur le problème du logarithme discret
- utilisé par le logiciel libre GNU Privacy Guard GPG.

ElGamal: Génération de la Clé

- On commence par choisir un nombre premier p.
- On choisit ensuite deux entiers :
- <u>a</u> tels que $a \in [0, p - 2]$ et
- <u>g</u> tel que $g \in [0, p - 1]$ et $k \in [1, p - 2]$:

$$g^k \neq 1 \bmod p$$

- On pose alors $A \equiv g^a \bmod p$.
- La **clé publique** sera le triplet (p, g, A) et la **clé secrète** sera l'entier <u>a</u>.
- Exemple :
  - P=11, g = 2, a = 8
  - $A = 2^8 \pmod{11} = 3$

ElGamal : Chiffrement

- Soit (p,g,A) une clé publique.
- On commence par choisir un entier b aléatoirement tel que $0 \leq b \leq p-1$.
- Un bloc de chiffres x du message d'origine tel que $x<p$ sera alors chiffré par un couple de blocs de chiffres (y_1,y_2) vérifiant $y_1 \equiv g^k \bmod p$ et $y_2 \equiv x.A^k \bmod p$.
- Exemple : (p,g,A) = (11, 2, 3), soit le message à chiffrer x = 7
- On choisi b = 4
- $E(m) = (2^4 \bmod 11, 7 * 3^4 \bmod 11) = (5, 6)$

ElGamal : déchiffrement

- Soit (p,g,A) une clé publique et a clé secrete.
- On commence par choisir un entier b aléatoirement tel que $0 \leq b \leq p-1$.
- Un couple de blocs de chiffres (y1,y2) du message chiffré correspondra au bloc de chiffres x du message d'origine vérifiant

$$x = y_1^{p-1-a} . y_2 \; mod \; p$$

- Exemple : (p,g,A) = (11, 2, 3), clé secrete a = 8 et message chiffrer (5, 6)
- X= 5^{11-1-8} . 6 mod 11 = 25. 6 mod 11 = 150 mod 11 = 7

- La sécurité du système El Gamal repose sur la difficulté de calculer la clé secrète <u>a</u> alors que l'on connait la clé publique,
- Cette opération revient en effet à retrouver la valeur de a à partir de celle de A.
- Ce problème, connu sous le nom de calcul du logarithme discret, est certes résolvable mais en un temps relativement long.
- Mais il n’est pas prouvé que la cryptanalyse d’un message chiffré avec El Gamal est équivalente au logarithme discret.
- En d’autres termes, rien ne prouve qu’il n’est pas cassable par un autre moyen.

SYMÉTRIE VS ASYMÉTRIQUE

	Symétrique	Asymétrique
Avantages	• Rapidité (jusqu'à 1000 fois plus rapide) • Facilité d'implantation sur hardware • Taille de clé : 128 bits (16 caractères: mémorisable)	• Distributions des clés facilitées : pas d'authentification • Permet de signer des messages facilement • Nombre de clés à distribuer est réduit par rapport aux clés symétriques
Inconvénients	• Nombre de clés à gérer • Distribution des clés (authentification, confidentialité) • Certaines propriétés (p.ex. signatures)sont difficiles à réaliser	• Taille des clés • Vitesse de chiffrement