

6 – Cryptographie à clef publique – I

MOTIVATION

- ▶ la cryptographie à **clef secrète** sait être fiable (cf. Cours 5)
- ▶ quelles raisons ont pu motiver un nouveau système ?
- ▶ dès les années 70, de nouvelles questions émergent :
 - ▶ comment s'échanger une clef secrète simplement ?
 - ▶ comment éviter un échange de clefs avant une communication sécurisée ?
- ▶ en 1976, **Diffie** et **Hellman** posent les jalons de la **cryptographie à clef publique**
 - « Nous nous trouvons aujourd'hui à l'aube d'une révolution en cryptographie ... »
- ▶ le système cryptographique devient **asymétrique** car les mécanismes de chiffrement et de déchiffrement diffèrent
- ▶ dès lors, la clef de chiffrement n'a plus besoin d'être tenue secrète
- ▶ le chiffrement se fait donc grâce à une **clef publique** alors que le déchiffrement s'effectue au moyen d'une **clef privée**
- ▶ de façon générale, ces algorithmes sont plus lents que la cryptographie à clef secrète
- ▶ l'analogie est cette fois celle de la **boîte-aux-lettres**.

PROTOCOLE DE DIFFIE-HELLMAN

- ▶ Diffie et Hellman ouvrent la voie de la **cryptographie moderne**, fondée sur les **fonctions à sens unique**
- ▶ en cela, ils résolvent le problème du **partage d'un secret** jusqu'alors jugé insoluble :
 1. **Alice** et **Bob** se mettent d'accord **publiquement** sur un grand nombre **premier** p et sur g une **racine primitive** modulo p
 2. **Alice** choisit un nombre a aléatoire qu'elle gardera secret
 3. **Alice** transmet **publiquement** à Bob $g^a \bmod p$
 4. **Bob** choisit de même un nombre b
 5. **Bob** transmet **publiquement** à Alice $g^b \bmod p$
 6. le **secret commun** est le nombre $s = g^{ab} \bmod p$
- ▶ Eva qui espionne le réseau ne peut pas déduire s car il lui faudrait résoudre le **problème du logarithme discret** (cf. TD 4).

FONCTION À SENS UNIQUE

- ▶ il existe des fonctions E bijectives dont le calcul de la réciproque $D = E^{-1}$ peut prendre ... des années : on les appelle des **fonctions à sens unique**
- ▶ par exemple, E l'**exponentielle discrète** et sa réciproque E^{-1} quand elle existe, le **logarithme discret** :
$$E_a : \begin{array}{l} \mathbb{Z}_n \rightarrow \mathbb{Z}_n \\ b \mapsto a^b \bmod n \end{array}$$
- ▶ le calcul de la puissance modulaire est rendu efficace par **élevations successives au carré** (cf. TP 6)
- ▶ quand E^{-1} existe, elle est très coûteuse à calculer ce qui pour de grandes valeurs rend son calcul en pratique impossible
- ▶ en outre, une **fonction à sens unique** est dite à **trappe** si un indice sur E^{-1} suffit à rendre son calcul facile
- ▶ la cryptographie à clef publique est fondée sur de telles fonctions et joue avec les limites de la technologie.

PROBLÈME DU LOGARITHME DISCRET

Soit p un nombre premier.

- ▶ **théorème** l'ensemble $\mathbb{F}_p \simeq \{0, \dots, p-1\}$ des classes de congruence modulo p muni de l'addition modulo p et de la multiplication modulo p est un corps
- ▶ **théorème** le groupe multiplicatif d'un corps fini est un **groupe cyclique**
- ▶ autrement dit, si p est premier et que g est une racine primitive modulo p , alors l'**exponentielle discrète** f est bijective :

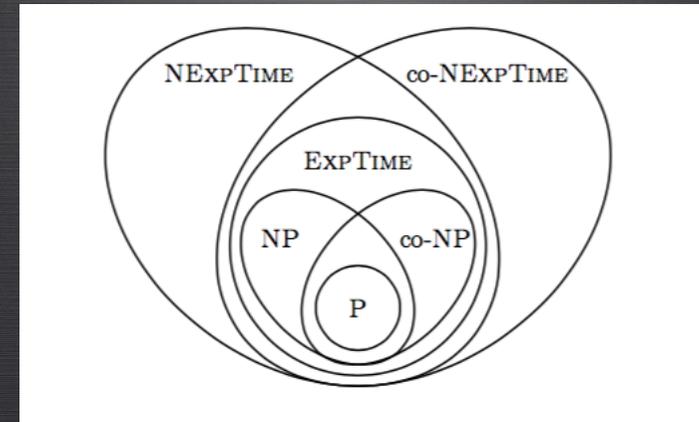
$$f: \begin{array}{ccc} \mathbb{Z}_p^* & \rightarrow & \mathbb{Z}_p^* \\ x & \mapsto & g^x \pmod p \end{array}$$

- ▶ une **racine primitive modulo p** est un entier g **générateur** du groupe cyclique \mathbb{Z}_p^* et on a :

$$\mathbb{Z}_p^* = \mathbb{Z}_p \setminus \{0\} = \{g^k \pmod p, k \in \{1, \dots, p-1\}\}$$

- ▶ pour p assez grand (plus d'une centaine de chiffres), cette fonction peut être considérée comme étant **à sens unique** : il est très coûteux avec la technologie actuelle de calculer le **logarithme discret** $f^{-1}(y)$ pour un $y \in \mathbb{Z}_p^*$ donné
- ▶ le problème de décision associé $LOGD \in NP \cap co-NP$

CLASSES DE COMPLEXITÉ TEMPORELLE



CRYPTOGRAPHIE À CLEF PUBLIQUE

- ▶ avec M le message **clair**, E_{K_e} et D_{K_d} les fonctions de **chiffrement** et **déchiffrement** et C le message **chiffré**, on a déjà vu que :

$$E_{K_e}(M) = C$$

$$D_{K_d}(C) = M$$

- ▶ du fait de l'asymétrie, on a en outre :

$$K_e \neq K_d$$

- ▶ K_e est une **clef publique** publiée sur un annuaire sous la forme d'un **certificat** tandis que K_d est une **clef privée** secrète
- ▶ le secret réside dans $H(K_d/K_e)$ c'est-à-dire dans la révélation de K_d sachant K_e
- ▶ la fonction E_{K_e} est publique et la fonction D_{K_d} est l'inverse de E_{K_e}
- ▶ on pourrait penser qu'il n'y a aucune incertitude sur K_d sachant K_e
- ▶ mais K_e peut être choisie de telle sorte que K_d soit **très difficile à calculer**
- ▶ c'est là qu'entrent en jeu les **fonctions à sens unique**.

RSA

- ▶ cet algorithme de chiffrement a été inventé en 1978 par **R. Rivest**, **A. Shamir** et **L. Adleman**
- ▶ breveté par le MIT en 1983, tombé dans le domaine public en 2000
- ▶ c'est le chiffrement asymétrique le plus employé aujourd'hui (cartes bleues, Internet, commerce électronique, ...)
- ▶ il exploite les réalités suivantes :
 - il est **facile** de construire de **grands nombres premiers**
 - il est **difficile** pour un ordinateur de **factoriser** un grand nombre
 - il est **difficile** de résoudre le **problème du logarithme discret**
- ▶ le problème de la **factorisation des entiers** est dans $NP \cap co-NP$
- ▶ en théorie et **seulement depuis 2002**, il est devenu **facile** de décider si un grand nombre est premier :

$$\text{Prime} \in P$$

INDICATRICE D'EULER

- ▶ l'indicatrice d'Euler d'un entier n notée $\varphi(n)$ est le nombre d'entiers compris entre 1 et n qui sont premiers avec n :

$$\varphi(n) = \text{card} \{i \in \{1, \dots, n\} / \text{pgcd}(i, n) = 1\}$$

- ▶ $\varphi(1) = 1$ et pour tout entier premier p : $\varphi(p) = p - 1$
- ▶ décomposition de n en produit de facteurs p_i premiers avec $1 \leq i \leq k$:

$$n = \prod_{i=1}^k p_i^{\alpha_i}$$

INDICATRICE D'EULER

- ▶ l'indicatrice d'Euler d'un entier n notée $\varphi(n)$ est le nombre d'entiers compris entre 1 et n qui sont premiers avec n :

$$\varphi(n) = \text{card} \{i \in \{1, \dots, n\} / \text{pgcd}(i, n) = 1\}$$

- ▶ $\varphi(1) = 1$ et pour tout entier premier p : $\varphi(p) = p - 1$
- ▶ décomposition de n en produit de facteurs p_i premiers avec $1 \leq i \leq k$:

$$n = \prod_{i=1}^k p_i^{\alpha_i}$$

- ▶ si p est premier, un élément m de $\{1, \dots, p^\alpha\}$ n'est pas premier avec p^α ssi $p \mid m$ donc $\varphi(p^\alpha) = p^\alpha - p^{\alpha-1} = (p-1)p^{\alpha-1}$

INDICATRICE D'EULER

- ▶ l'indicatrice d'Euler d'un entier n notée $\varphi(n)$ est le nombre d'entiers compris entre 1 et n qui sont premiers avec n :

$$\varphi(n) = \text{card} \{i \in \{1, \dots, n\} / \text{pgcd}(i, n) = 1\}$$

- ▶ $\varphi(1) = 1$ et pour tout entier premier p : $\varphi(p) = p - 1$
- ▶ décomposition de n en produit de facteurs p_i premiers avec $1 \leq i \leq k$:

$$n = \prod_{i=1}^k p_i^{\alpha_i}$$

- ▶ si p est premier, un élément m de $\{1, \dots, p^\alpha\}$ n'est pas premier avec p^α ssi $p \mid m$ donc $\varphi(p^\alpha) = p^\alpha - p^{\alpha-1} = (p-1)p^{\alpha-1}$
- ▶ si x et y sont premiers, on a $\varphi(x.y) = \varphi(x).\varphi(y)$ d'où le calcul de $\varphi(n)$ pour $n > 1$:

$$\varphi(n) = \prod_{i=1}^k (p_i - 1)p_i^{\alpha_i-1} = n \prod_{i=1}^k \left(1 - \frac{1}{p_i}\right)$$

Exemple

INDICATRICE D'EULER

- ▶ l'indicatrice d'Euler d'un entier n notée $\varphi(n)$ est le nombre d'entiers compris entre 1 et n qui sont premiers avec n :

$$\varphi(n) = \text{card} \{i \in \{1, \dots, n\} / \text{pgcd}(i, n) = 1\}$$

- ▶ $\varphi(1) = 1$ et pour tout entier premier p : $\varphi(p) = p - 1$
- ▶ décomposition de n en produit de facteurs p_i premiers avec $1 \leq i \leq k$:

$$n = \prod_{i=1}^k p_i^{\alpha_i}$$

- ▶ si p est premier, un élément m de $\{1, \dots, p^\alpha\}$ n'est pas premier avec p^α ssi $p \mid m$ donc $\varphi(p^\alpha) = p^\alpha - p^{\alpha-1} = (p-1)p^{\alpha-1}$
- ▶ si x et y sont premiers, on a $\varphi(x.y) = \varphi(x).\varphi(y)$ d'où le calcul de $\varphi(n)$ pour $n > 1$:

$$\varphi(n) = \prod_{i=1}^k (p_i - 1)p_i^{\alpha_i-1} = n \prod_{i=1}^k \left(1 - \frac{1}{p_i}\right)$$

Exemple $504 = 2^3 \times 3^2 \times 7$

$$\varphi(504) = (2-1) \cdot 2^{3-1} \times (3-1) \cdot 3^{2-1} \times (7-1) \cdot 7^{1-1} = 144$$

$$\varphi(504) = 504 \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{3}\right) \left(1 - \frac{1}{7}\right) = 144$$

THÉORÈME DE FERMAT-EULER

▶ petit théorème de Fermat (1640)

pour p premier et m tel que $0 < m < p$:

$$m^{p-1} \equiv 1 \pmod{p}$$

et sa généralisation :

▶ théorème d'Euler (1761)

$$m^{\varphi(n)} \equiv 1 \pmod{n} \text{ si } \text{pgcd}(m, n) = 1$$

ENCORE UN PEU D'ARITHMÉTIQUE MODULAIRE ...

▶ théorème de Bézout

soient a et b deux entiers naturels, il existe des entiers relatifs u et v tels que :

$$\text{pgcd}(a, b) = u \cdot a + v \cdot b$$

▶ conséquence : si a et b sont premiers entre eux, il existe u tel que

$$u \cdot a \equiv 1 \pmod{b}$$

ENCORE UN PEU D'ARITHMÉTIQUE MODULAIRE ...

▶ théorème de Bézout

soient a et b deux entiers naturels, il existe des entiers relatifs u et v tels que :

$$\text{pgcd}(a, b) = u \cdot a + v \cdot b$$

▶ conséquence : si a et b sont premiers entre eux, il existe u tel que

$$u \cdot a \equiv 1 \pmod{b}$$

▶ théorème dit des restes chinois ou chinois

si p_1, \dots, p_k sont des entiers 2 à 2 premiers et a_1, \dots, a_k des entiers quelconques alors le système de k équations suivant admet une unique solution modulo $p_1 \dots p_k$:

$$x = a_i \pmod{p_i} \text{ pour } 1 \leq i \leq k$$

LE CRYPTOSYSTÈME RSA

▶ préparation des clefs :

1. le destinataire **Bob** choisit 2 grands nombres premiers distincts p et q (de l'ordre de 10^{100}) et calcule $n = p \cdot q$
2. il choisit e un entier premier à $\varphi(n) = (p-1)(q-1)$
3. il calcule sa **clef privée** d telle que $e \cdot d \equiv 1 \pmod{\varphi(n)}$
4. il publie le **couple** (n, e) : c'est sa **clef publique**

▶ chiffrement de M avec $0 < M < n$:

Alice chiffre son **message** M avec la **clef publique** (n, e) de **Bob** et envoie le **chiffré** C à **Bob** :

$$C = E_e(M) \equiv M^e \pmod{n}$$

▶ déchiffrement de C :

Bob déchiffre le **chiffré** C envoyé par **Alice** avec sa **clef privée** d :

$$M = D_d(C) \equiv C^d \pmod{n}$$

Les **implémentations** sont soit logicielles, soit matérielles (voire mixtes). Sur des cartes dédiées, RSA est environ 2000 fois plus lent que AES.

RSA : EXEMPLE

► préparation des clefs :

1. le destinataire **Bob** choisit 2 nombres premiers distincts $p = 7$ et $q = 11$ puis calcule $n = p \cdot q = 77$
2. il choisit $e = 7$ un entier **premier à** $\varphi(n) = (p-1)(q-1) = 60$
3. d'après Bézout, $\text{pgcd}(7, 60) = 1 = (-17) * 7 + (2) * 60$
4. il calcule sa **clef privée** $d = 43$ telle que $ed \equiv 1 \pmod{\varphi(n)}$
5. il publie le **couple** $(n, e) = (77, 7)$: c'est sa **clef publique**

RSA : EXEMPLE

► préparation des clefs :

1. le destinataire **Bob** choisit 2 nombres premiers distincts $p = 7$ et $q = 11$ puis calcule $n = p \cdot q = 77$
2. il choisit $e = 7$ un entier **premier à** $\varphi(n) = (p-1)(q-1) = 60$
3. d'après Bézout, $\text{pgcd}(7, 60) = 1 = (-17) * 7 + (2) * 60$
4. il calcule sa **clef privée** $d = 43$ telle que $ed \equiv 1 \pmod{\varphi(n)}$
5. il publie le **couple** $(n, e) = (77, 7)$: c'est sa **clef publique**

► chiffrement de M avec $0 < M < n$:

Alice chiffre son message $M = 13$ avec la **clef publique** $(n, e) = (77, 7)$ de **Bob** et envoie le **chiffré** C à **Bob** :

$$C = E_e(M) \equiv M^e \pmod{n} = 13^7 \pmod{77} = 62$$

RSA : EXEMPLE

► préparation des clefs :

1. le destinataire **Bob** choisit 2 nombres premiers distincts $p = 7$ et $q = 11$ puis calcule $n = p \cdot q = 77$
2. il choisit $e = 7$ un entier **premier à** $\varphi(n) = (p-1)(q-1) = 60$
3. d'après Bézout, $\text{pgcd}(7, 60) = 1 = (-17) * 7 + (2) * 60$
4. il calcule sa **clef privée** $d = 43$ telle que $ed \equiv 1 \pmod{\varphi(n)}$
5. il publie le **couple** $(n, e) = (77, 7)$: c'est sa **clef publique**

► chiffrement de M avec $0 < M < n$:

Alice chiffre son message $M = 13$ avec la **clef publique** $(n, e) = (77, 7)$ de **Bob** et envoie le **chiffré** C à **Bob** :

$$C = E_e(M) \equiv M^e \pmod{n} = 13^7 \pmod{77} = 62$$

► déchiffrement de C :

Bob déchiffre le **chiffré** $C = 62$ envoyé par **Alice** avec sa **clef privée** $d = 43$:

$$M = D_d(C) \equiv C^d \pmod{n} = 62^{43} \pmod{77} = 13$$

RSA : JUSTIFICATION

$ed \equiv 1 \pmod{\varphi(n)}$ et $0 < M < n$:

- l'existence de d est due au théorème de Bézout
- si $\text{pgcd}(M, n) = 1$ d'après le théorème de Fermat-Euler :

$$M^{\varphi(n)} \equiv 1 \pmod{n}$$

$$(M^e)^d = M^{ed} \equiv M \pmod{n}$$

- si $\text{pgcd}(M, n) \neq 1$: supposons e.g. $M \equiv 0 \pmod{p}$ et $\text{pgcd}(M, q) = 1$

$$M^{\varphi(n)} \equiv 0 \pmod{p}$$

et d'après le théorème de Fermat-Euler :

$$M^{\varphi(q)} \equiv 1 \pmod{q}$$

$\varphi(n) = \varphi(p)\varphi(q)$ donc $M^{\varphi(n)} \equiv 1 \pmod{q}$

on a obtenu :
$$\begin{cases} M^{ed} \equiv 0 \pmod{p} \\ M^{ed} \equiv M \pmod{q} \end{cases}$$

RSA : JUSTIFICATION (SUITE)

d'après le théorème chinois : $\exists! c$ tel que $M^{ed} \equiv c \pmod{pq}$

$$\begin{cases} M^{ed} = 0 + \lambda p \\ \lambda p \equiv M \pmod{q} \end{cases}$$

p et q étant premiers, on a les coefficients de Bézout u et v tels que

$$u.p + v.q = 1$$

en posant $\lambda = uM$, on trouve la solution :

$$c = upM$$

en effet, comme M est multiple de p , on obtient :

$$c = (1 - v.q).M = M - (v.q.M) \equiv M \pmod{pq}$$

on conclut : $M^{ed} \equiv M \pmod{n}$

COEFFICIENTS DE BÉZOUT

- ▶ l'algorithme d'Euclide étendu prolonge l'algorithme d'Euclide calculant le pgcd de deux entiers a et b
- ▶ il fournit les **coefficients de Bézout**, c'est-à-dire des entiers relatifs u et v qui vérifient :
$$\text{pgcd}(a,b) = u.a + v.b \text{ avec } u,v \in \mathbb{Z}$$
- ▶ notez que pour a et b donnés, le couple (u,v) n'est pas unique

Algorithme récursif en PYTHON (cf. Algo 1 Cours 11) :

```
def bezout(a,b) :
    if b == 0 :
        return (a,1,0)
    q = a // b
    r = a % b
    g,u,v = bezout(b,r)
    return g,v,u-q*v
```

Exemples

27 et 40 sont premiers entre eux : $\text{pgcd}(27,40) = 1 = (3) * 27 + (-2) * 40$

27 et 42 ne sont pas premiers entre eux : $\text{pgcd}(27,42) = 3 = (-3) * 27 + (2) * 42$

CRYPTANALYSE (PAR SEULE FACTORISATION DE n)

Exemple

- ▶ Bob envoie un message à Alice par RSA avec $(n,e) = (133,61)$
 1. quelle est la clef privée d'Alice?
 2. si le chiffré reçu par Alice est 17, quel est le clair correspondant?
(entre 1 et $n-1$)
- ▶ par tâtonnements, on factorise $n = 133$ en $p.q$ avec $p = 7$ et $q = 19$
- ▶ on calcule $\varphi(133) = 6.18 = 108$
- ▶ pour calculer la **clef privée** d d'Alice correspondant à $e = 61$:

$$d \equiv e^{-1} \pmod{\varphi(n)}$$

- ▶ on recherche les coefficients de Bézout sur $(61,108)$:
$$\text{bezout}(61,108) = (1,-23,13)$$
donc $\text{pgcd}(61,108) = 1 = (-23) * 61 + (13) * 108$
- ▶ on en déduit donc que : $d \equiv -23 \pmod{108} \Rightarrow d = 85$
- ▶ pour déchiffrer le message chiffré $C = 17$ destiné à Alice, il suffit de calculer $17^{85} \pmod{133}$:

$$\text{expMod}(17,85,133) = 73$$

L'ATTAQUE DE L'HOMME DU MILIEU

Elle porte sur la communication des clefs :

- ▶ Bob demande sa **clef publique** (n,e) à Alice pour lui envoyer sûrement le message M
 - ▶ Alice envoie (n,e) à Bob
 - ▶ Melchior intercepte (n,e) et envoie à la place (n',e') à Bob
 - ▶ Bob chiffre en utilisant à son insu la clef factice (n',e') de Melchior et envoie le message ainsi chiffré C' à Alice
 - ▶ Melchior intercepte à nouveau le message chiffré C' et le déchiffre aisément en M
...
- Melchior pourrait arrêter là les nuisances, mais non :
- ▶ Melchior rechiffre M avec (n,e) et le transmet à Alice ...

Cette attaque classique porte aussi le nom de *Man in the middle* en anglais.

Elle débouche sur les problèmes d'**authentification** pour éviter l'**usurpation d'identité**.

Les **certificats** et les **chaînes de certification** en sont, à l'heure actuelle, la seule parade.

A suivre ...