RSA, pilier de la cryptographie asymétrique

Jang Schiltz

Assistant-Professeur à l'Université du Luxembourg

séminaire RSA

Introduction

Jean-Claude Asselborn

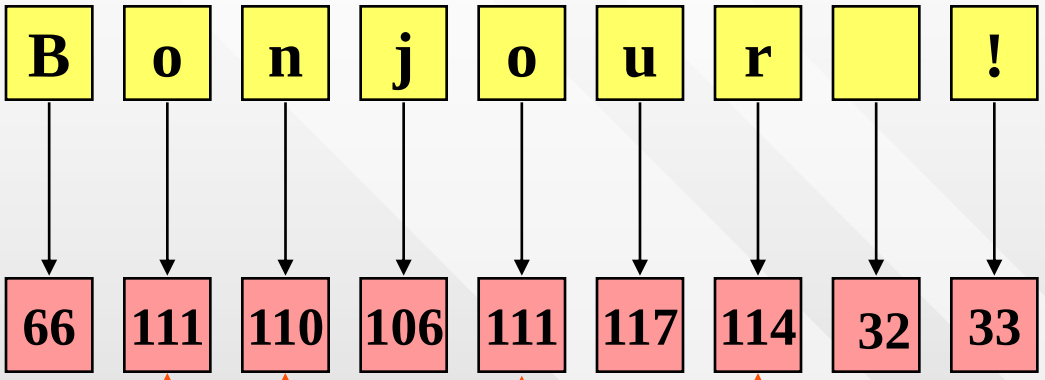
Professeur au Centre Universitaire de Luxembourg

l'écrit électronique : une suite de nombres

Cryptology & Security Initiative

Bonjour !

écrit



suite de caractères

suite de nombres entiers

liste de référence:

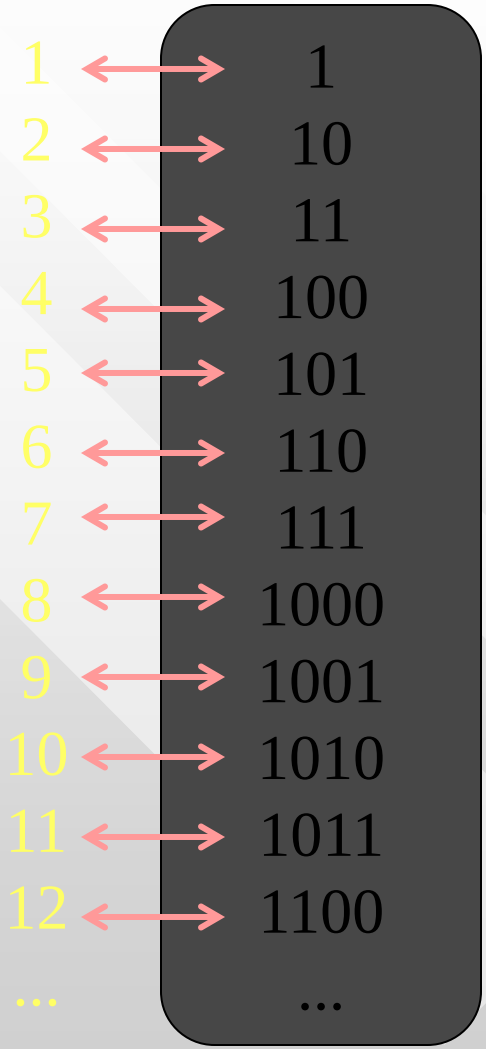
...
110 n
111 o
112 p
113 q
114 r
...

exemple : code ASCII

nombre entier : une suite de chiffres binaires

Cryptology & Security Initiative

liste numérotée



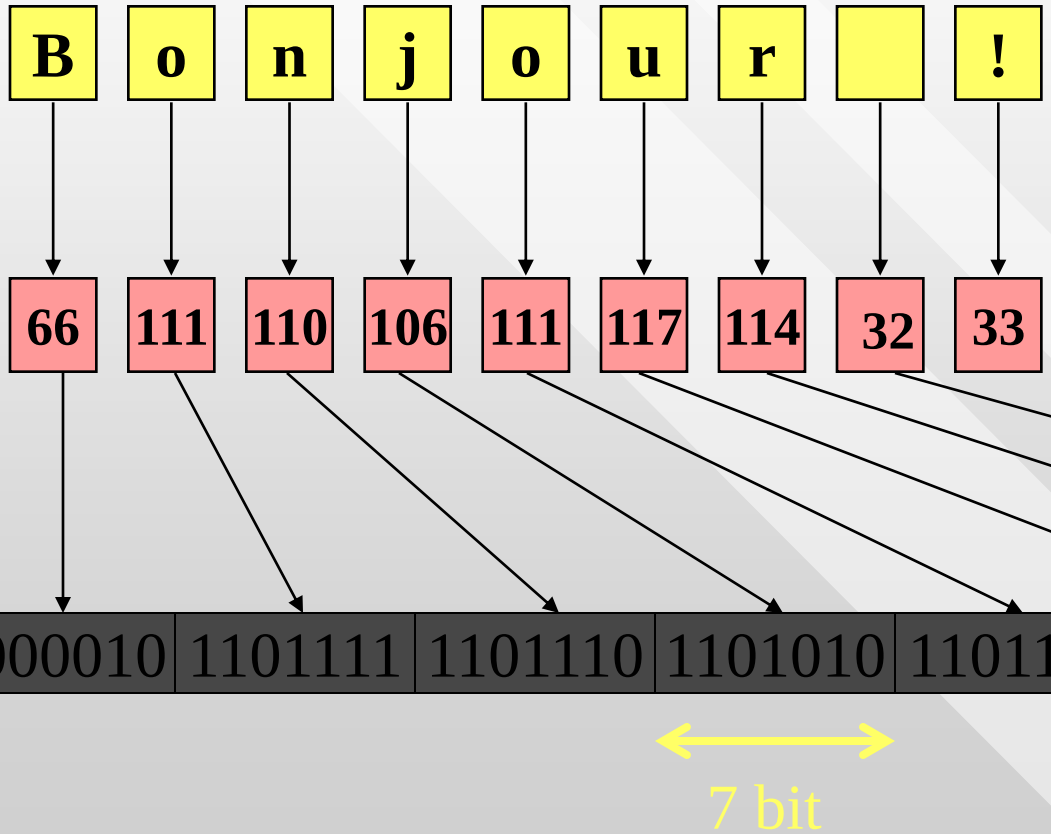
A chaque nombre entier correspond une suite binaire et inversément

l'écrit électronique : une suite binaire

Cryptology & Security Initiative

Bonjour !

écrit



suite de caractères

suite de nombres entiers

suite binaire

message = nombre

Cryptology & Security Initiative

message

Bonjour !

une suite de nombres binaires

1000010	1101111	1101110	1101010	1101111	1110101	1110010	0100000

un grand nombre binaire

1000010110111111011101101010110111111010111100100100000

un grand nombre

37 billiards 646 billions 688 milliards 348 millions 633 mille et 376

Tout message peut être interprété comme un grand nombre entier

calculer avec des messages

Cryptology & Security Initiative

message **Bonjour !**

nombre associé [redacted]

(**Bonjour !**)^2 = ([redacted])^2

= [redacted] 1.417.273.143.619.127.986.862.606.861.157.376

1 quintilliard 417 quintillions 273 quadrillards 143 quadrillions
619 trilliards 127 trillions 986 billiards 862 billions
606 milliards 861 millions 157 mille et 376

En cryptographie on calcule avec des nombres à plusieurs centaines de chiffres décimaux.

conséquences des grands nombres

Cryptology & Security Initiative

Les processeurs calculent seulement sur 64 ou 128 bits

on doit disposer d'algorithmes spéciaux de calcul

âge de l'univers : 10^{10} années = 10^{17} secondes = 10^{26} nanosecondes

l'exploration complète de domaines de grands nombres est pratiquement impossible

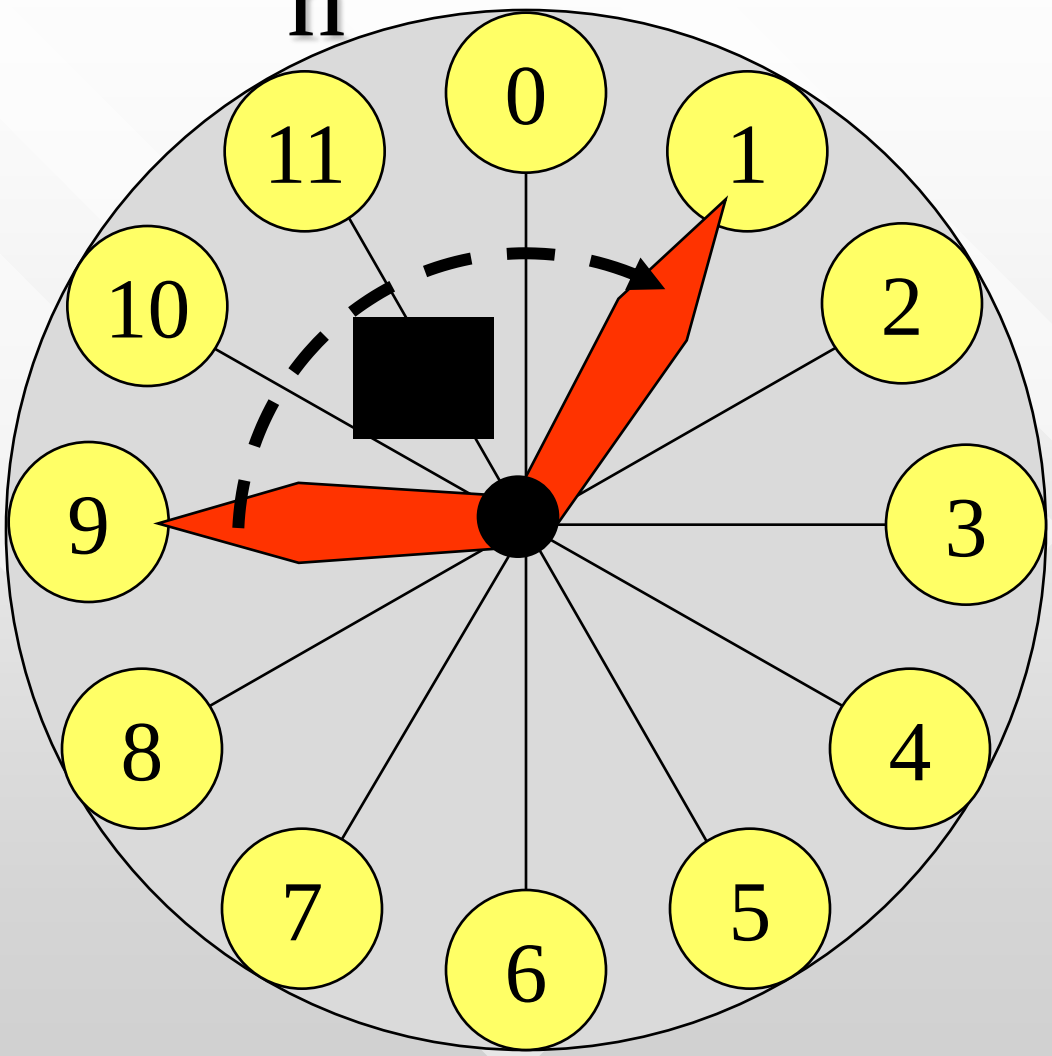
rappel: arithmétique modulo n

Cryptology & Security Initiative

9 + 4 = 13

= 1 (mod 12)

= reste de 13 : 12



Le résultat varie entre 0 et 11

en général: x (mod n) = reste de x : n

conséquences

Cryptology & Security Initiative

Les valeurs varient entre 0 et (n -1)

on peut limiter la taille des grands nombres

Les valeurs des fonctions peuvent sauter brusquement

il existe des fonctions difficiles à inverser

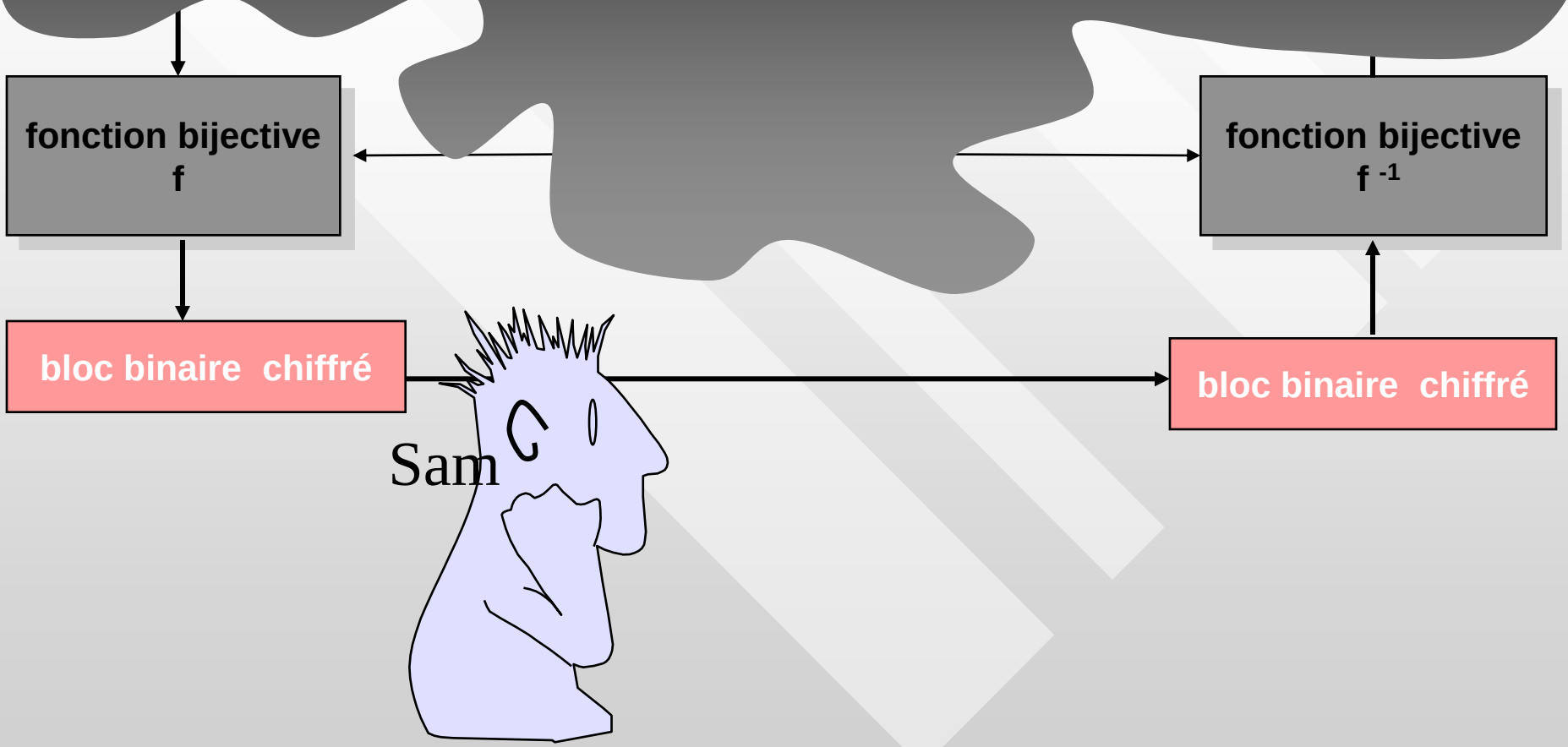
Les messages doivent être inférieurs à n

on doit limiter la taille des messages

- encryptage : on doit fractionner le message en blocs
- signature : on ne signe pas le message proprement dit, mais un condensé du message

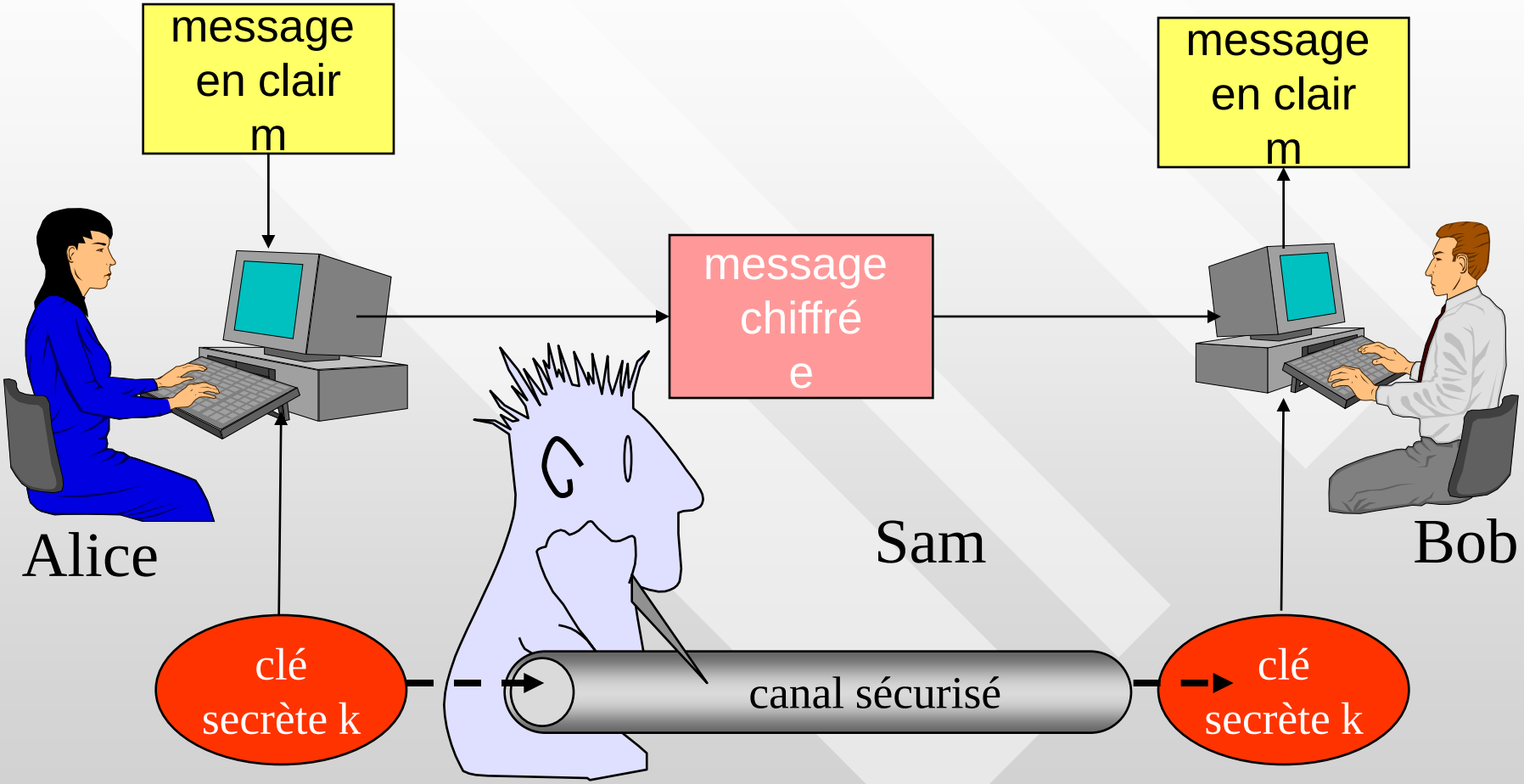
avant 1976 : chiffrage s[redacted]trique

Cryptology & Security Initiative



problème de la transmission de clé

Cryptology & Security Initiative



"New Directions in Crypt..."

Cryptology & Security Initiative

Ralph Merkle

Whitfield **Diffie** Martin **Hellman**

"New Directions in Cryptography"
[IEEE Transactions on Information Theory, November 1976]

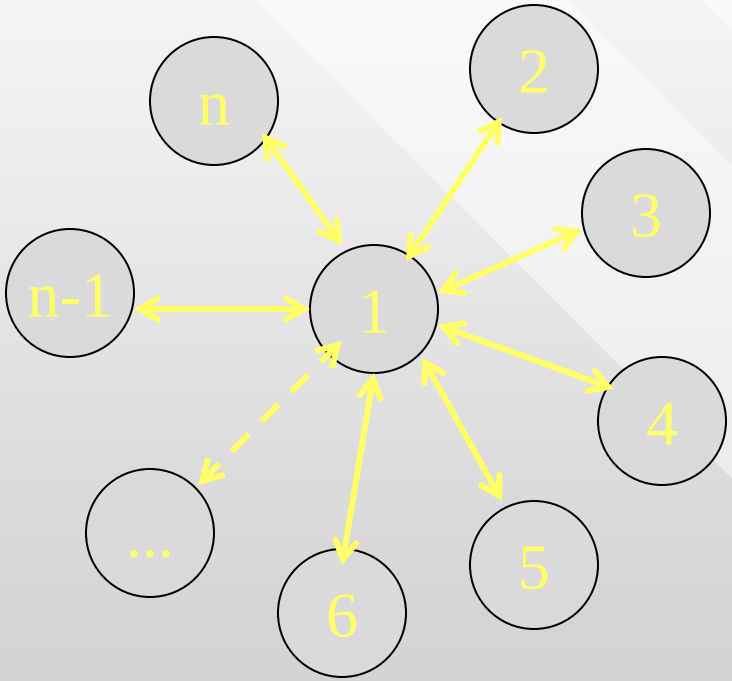
1976

Commerce électronique et clés

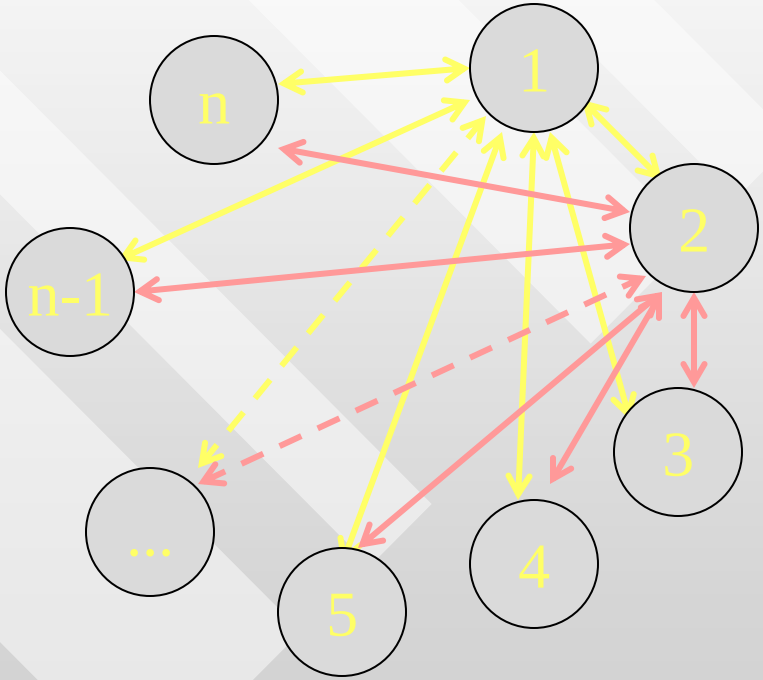
Cryptology & Security Initiative

communication bilatérale :1 clé

réseau ouvert



réseau en étoile: n - 1 clés



n. (n-1) / 2 clés

échange de clé Diffie-Hellman

Cryptology & Security

p, g

x

$A = g^x \pmod p$

y

$B = g^y \pmod p$

A

B

Alice

Bob

Sam

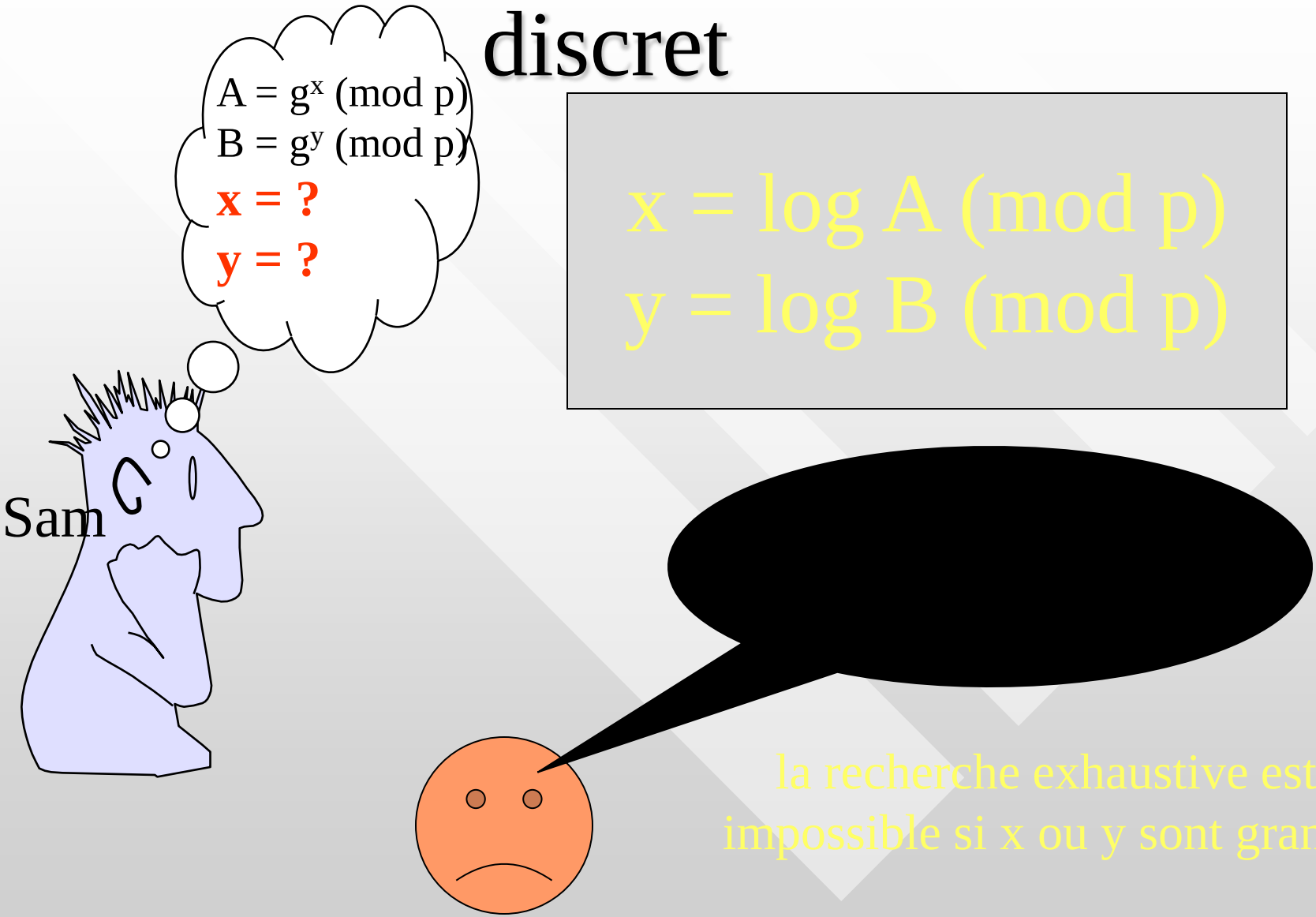
clé $= B^x \pmod p$
$= (g^y)^x \pmod p$
$= g^{x*y} \pmod p$

$A = g^x \pmod p$
$B = g^y \pmod p$
x = ?
y = ?

clé $= A^y \pmod p$
$= (g^x)^y \pmod p$
$= g^{x*y} \pmod p$

le problème du logarithme discret

Cryptology & Security Initiative



$$x = \log A \pmod p$$
$$y = \log B \pmod p$$

la recherche exhaustive est impossible si x ou y sont grands

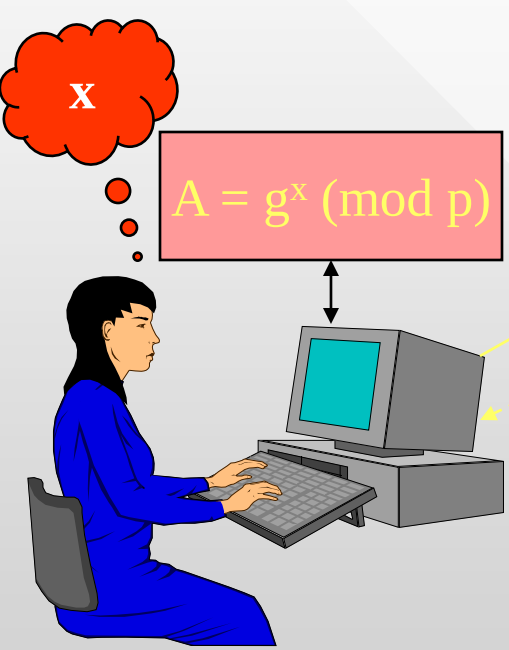
clé privée - clé publique

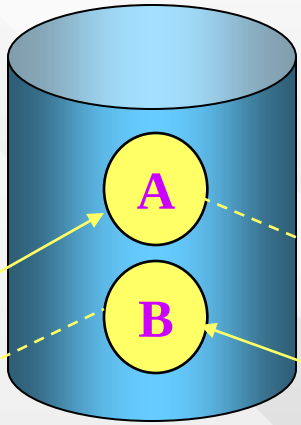
Cryptology & Security Initiative

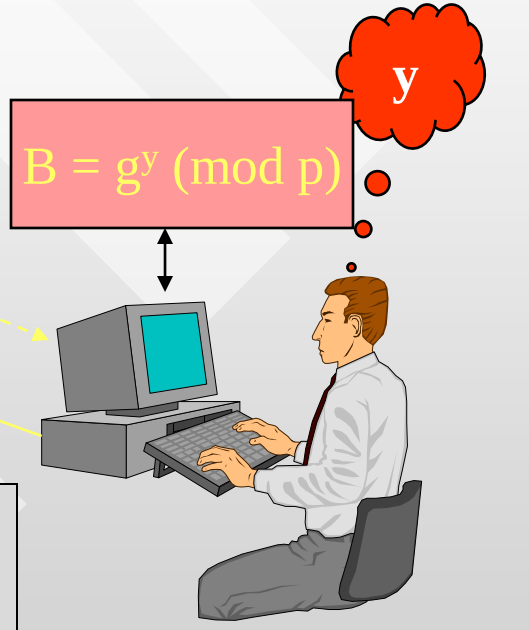
registre des clés publiques

clé privée

clé privée

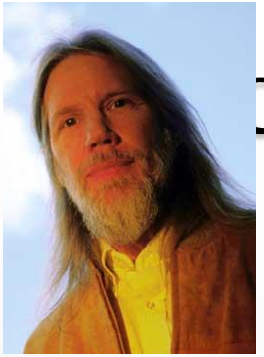






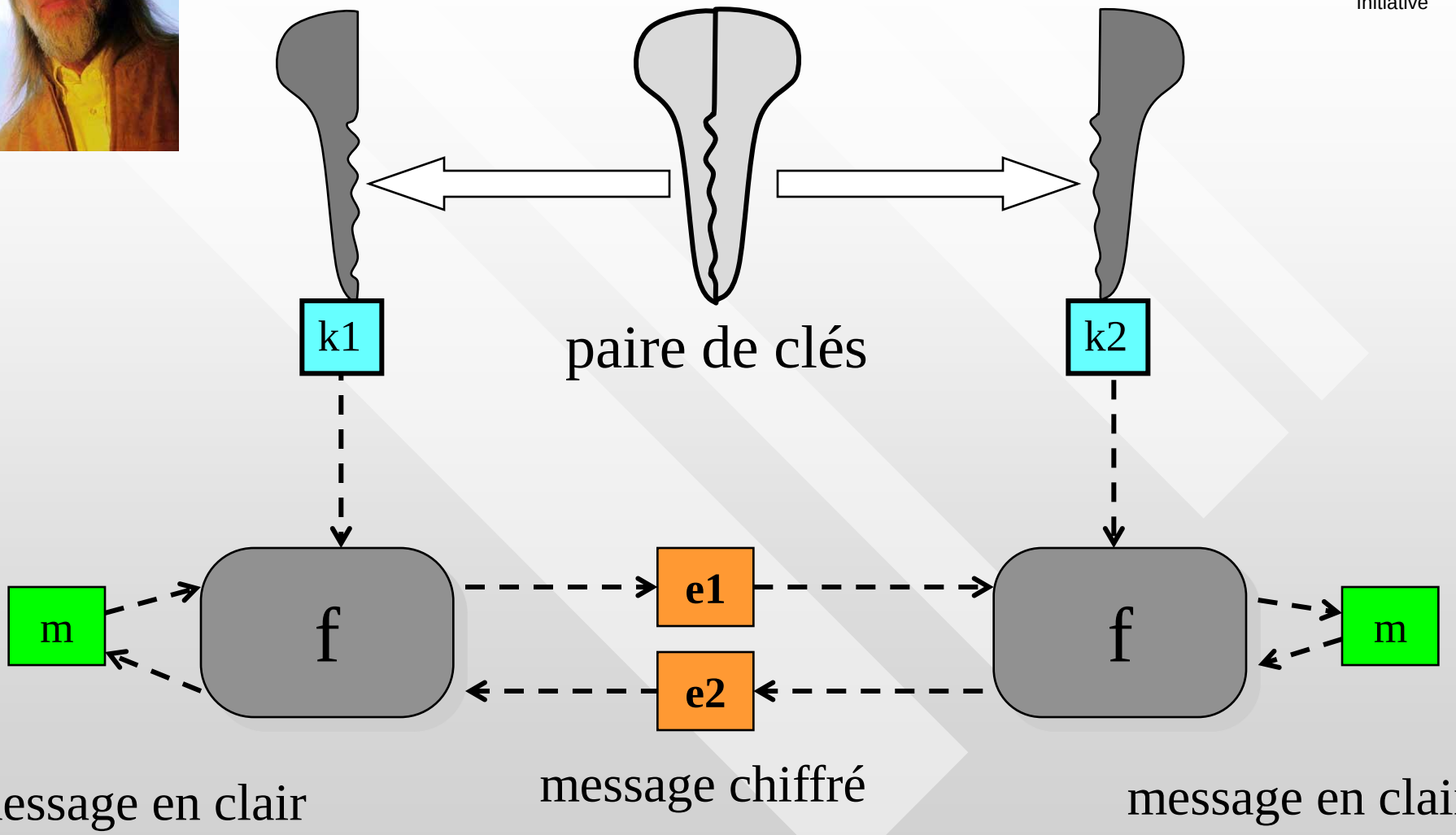
clé secrète commune =

(clé publique de l'autre) ^ ma clé privée



cryptographie asymétrique

Cryptology & Security Initiative



utilisation comme procédé de chiffrage

Cryptology & Security Initiative

clé publique du destinataire

clé secrète du destinataire

message en clair

message chiffré

message en clair

utilisation comme procédé d'authentification

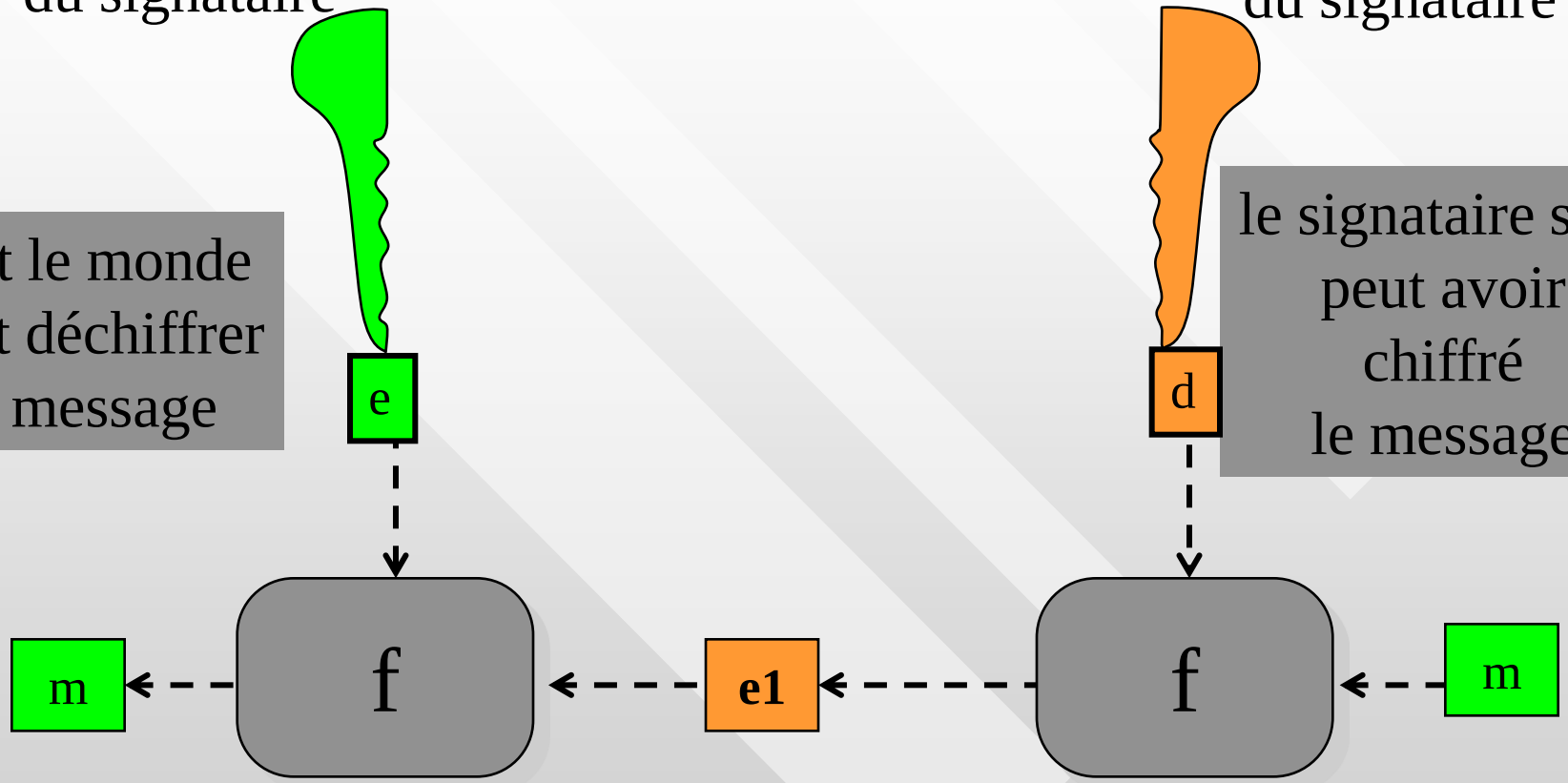
Cryptology & Security Initiative

clé publique du signataire

clé secrète du signataire

tout le monde peut déchiffrer le message

le signataire seul peut avoir chiffré le message



message en clair

message chiffré

message en clair

à la recherche d'un exemple

Cryptology
& Security
Initiative

RSA

Cryptology
& Security
Initiative

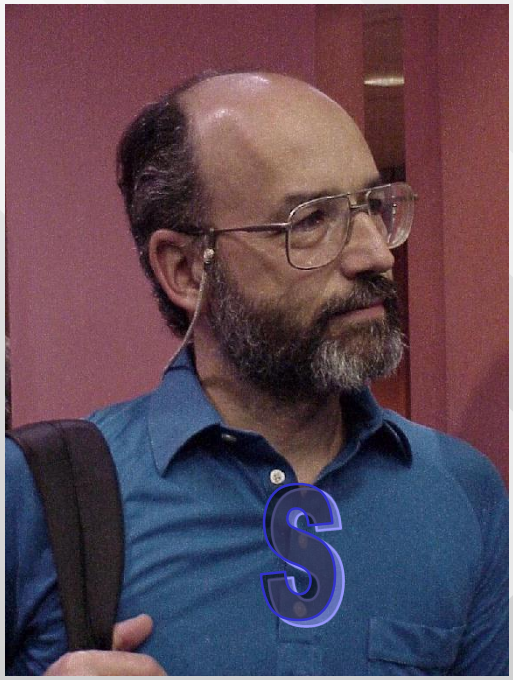
A Method for Obtaining Digital Signatures and Public Key Cryptosystems

[Communications of the ACM, Vol 21, n° 12]

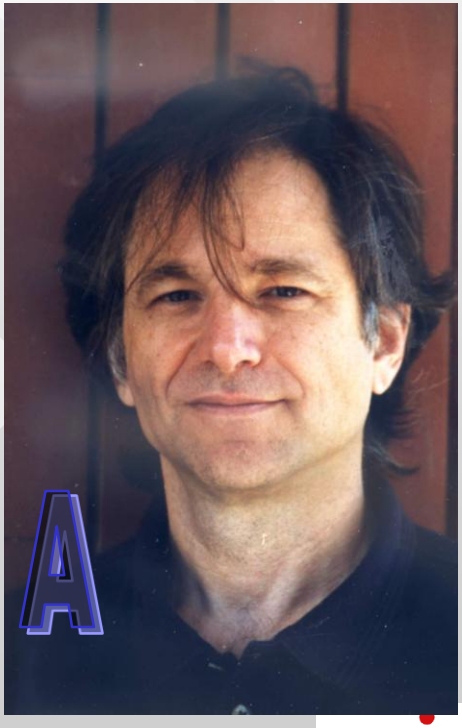
1978

Ronald **Rivest**

Adi **Shamir**



Leonard **Adleman**



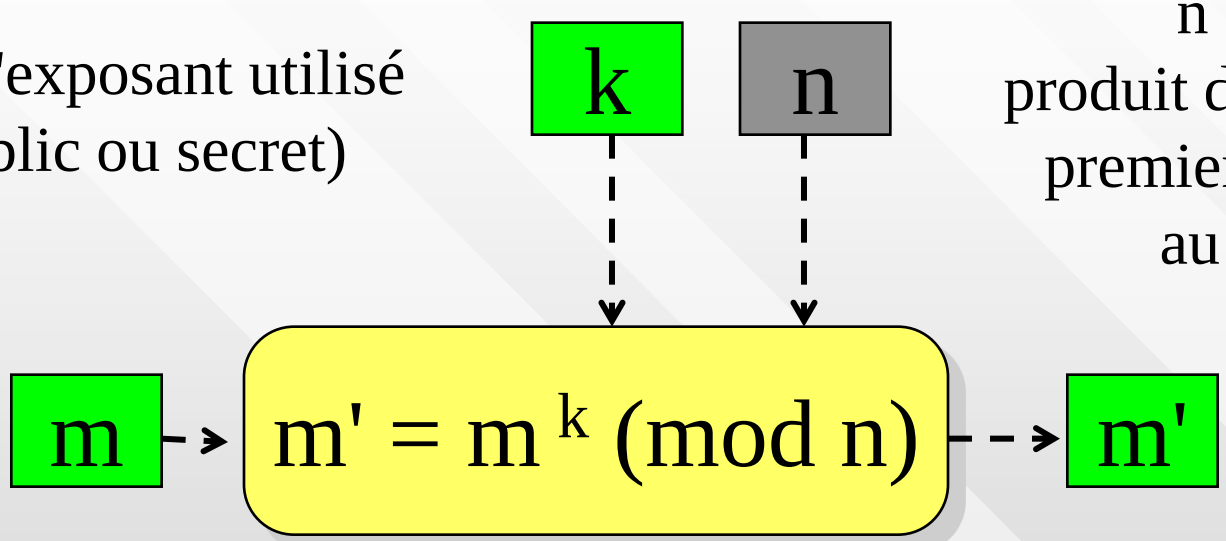
RSA en 1977

codage RSA

Cryptology & Security Initiative

k est l'exposant utilisé (public ou secret)

n est un produit de 2 nombres premiers à 512 bit au moins



message en entrée

message en sortie

n est public, mais ses deux facteurs sont secrets, car ils permettent de déduire l'une des clés à partir de l'autre

exemplé simplifié RSA

Cryptology & Security Initiative

2 nombres premiers secrets $p = 5 \quad q = 11$

$n = p.q = 5 \cdot 11 = 55$

sélection d'un exposant public: $e = 3$

calcul de l'exposant secret: $d = \text{calcul}(p, q, e) = 27$

message à chiffrer: $m = 10011_2 = 19 < 55$

chiffrage:

$$m' = 19^e \pmod n$$
$$= 19^3 \pmod{55}$$
$$= 6859 \pmod{55}$$
$$= 39$$

déchiffrage:

$$m = 39^d \pmod n$$
$$= 39^{27} \pmod{55}$$
$$= 19$$

terminologie

Cryptology & Security Initiative

$n = p \cdot q$	modulus	le module
e	encryption exponent	l'exposant public
d	decryption exponent	l'exposant privé
(n, e)	encryption key	la clé publique
(n, d)	decryption key	la clé privée

séminaire RSA

Bases mathématiques

Jang Schiltz

Enseignant - chercheur au Centre Universitaire de Luxembourg

contenu

Cryptology & Security Initiative

- Divisibilité
- Arithmétique modulaire
- Les théorèmes fondamentaux
- Le RSA
- Le problème de la factorisation

chapitre 1

Cryptology & Security Initiative

Divisibilité

- Arithmétique modulaire
- Thèorèmes fondamentaux
- Le RSA
- Le problème de la factorisation

définition de la divisibilité

Cryptology & Security Initiative

Soient *a* et *b* des entiers. On dit que *a* divise *b* et on note *a*|*b* s'il existe un entier *c* tel que $b = a \cdot c$.

On dit alors que *a* est un diviseur de *b* ou que *b* est divisible par *a*.

1^{er} exemple de divisibilité

Cryptology & Security Initiative

$$24 = 2 \cdot 12 \quad \text{et} \quad 24 = 3 \cdot 8$$

$$\Rightarrow \; 2|24 \quad 12|24 \quad 3|24 \quad 8|24$$

$$24 = (-2)(-12)$$

$$\Rightarrow \; -2|24 \quad -12|24$$

2^{eme} exemple de divisibilité

Cryptology & Security Initiative

Aucun entier non nul *a* n'est divisible par 0

Sinon, il existerait c tel que $a = c \cdot 0$

Mais, $0 = 0 \cdot c$, pour tout c

$\Rightarrow 0|0$

3^{eme} exemple de divisibilité

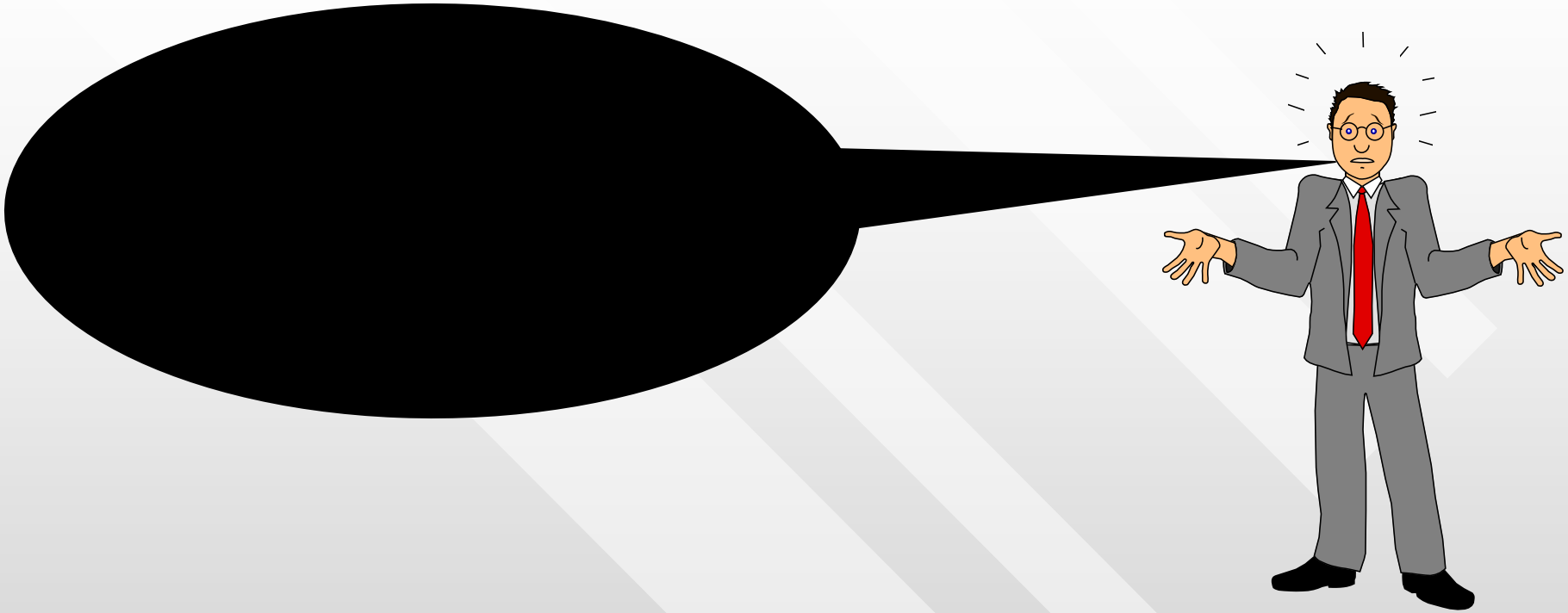
Cryptology & Security Initiative

Tout entier *a* divise 0.

En effet, $0 \cdot a = 0$.

4^{eme} exemple de divisibilité

Cryptology & Security Initiative



Ce sont 1 et 5.

définition d'un nombre premier

Cryptology & Security Initiative

Un nombre entier positif $p > 1$ est appelé nombre premier si ses seuls diviseurs positifs sont 1 et p.

Un nombre non premier est dit nombre composé.

division euclidienne

Cryptology & Security Initiative

Soient deux entiers a et b, avec $b \neq 0$.

Alors, il existe des entiers p et r uniques, tels que $a = bq + r$ et $0 \leq r < |b|$.

q est appelé le quotient de a par b et r le reste.

1^{er} exemple de division euclidienne

Cryptology & Security Initiative

$a = 37 \qquad b = 15$

$37 = 2 \cdot 15 + 7$

$\Rightarrow q = 2 \text{ et } r = 7$

2^{eme} exemple de division euclidienne

Cryptology & Security Initiative

$a = 37 \qquad b = -15$

$37 = (-2)(-15) + 7$

$\Rightarrow q = -2 \text{ et } r = 7$

3^{eme} exemple de division euclidienne

Cryptology & Security Initiative

$a = -37 \qquad b = 15$

$-37 = -2 \cdot 15 - 7$

$\Rightarrow q = -2$ et $r = -7$

Faux !, car $r>0$

3^{eme} exemple de division euclidienne

Cryptology & Security Initiative

$$a = -37 \qquad b = 15$$

$$-37 = -3 \cdot 15 + 8$$

$$\Rightarrow q = -3 \text{ et } r = 8$$

le pgcd

Cryptology & Security Initiative

On appelle plus grand commun diviseur des entiers *a* et de *b* et on note pgcd(*a*,*b*), le plus grand entier positif qui est à la fois diviseur de *a* et de *b*.

Exemple : a = 12 b = 15

Diviseurs de 12: {1, 2, 3, 4, 6, 12}
Diviseurs de 15: {1, 3, 5, 15}

$\Rightarrow$ pgcd(12,15) = 3

entiers premiers entre eux

Cryptology & Security Initiative

On dit que deux entiers *a* et *b* sont premiers entre eux si et seulement si pgcd(*a*,*b*) =1

Exemple : a = 7 b = 12

Diviseurs de 12: {1, 2, 3, 4, 6, 12}
Diviseurs de 7: {1, 7}

$\Rightarrow$ 7 et 12 sont premiers entre eux.

théorème de Bezout

Cryptology & Security Initiative

Deux entiers *a* et *b* sont premiers entre eux si et seulement s'il existe deux entiers *u* et *v* tels que

$$ua + vb = 1$$

chapitre 2

Cryptology & Security Initiative

Divisibilité

Arithmétique modulaire

Théorèmes fondamentaux

Le RSA

Le problème de la factorisation

définition de la congruence

Cryptology & Security Initiative

Si *a*, *b* et *n* sont des entiers, on dit que *a* est congru à *b* modulo *n* et on note $a = b \pmod{n}$, si n | *a-b*

On dit aussi que *b* est un résidu de *a* modulo *n*, ou un reste de *a* modulo *n*.

1^{er} exemple de congruence

Cryptology & Security Initiative

$$9 = 23 - 14$$

$$\Rightarrow \quad 23 = 14 \pmod 9$$

N'importe quels deux nombres de l'ensemble {…, -4, 5, 14, 23, …} sont congrus modulo 9.

2^{eme} exemple de congruence

Cryptology & Security Initiative

Pour tous entiers a et b, il existe c tel que

$$b - a = c \cdot 1$$

$$\Rightarrow a = b \pmod 1$$

remarque sur les congruences

Cryptology & Security Initiative

$a = b \pmod{n}$ si et seulement si $a = b \pmod{-n}$.

Pour cette raison, on ne considère que des modules positifs.

relation d'équivalence

Cryptology & Security Initiative

Soient a, b, c et n des entiers.

$a = a \pmod n$

$a = b \pmod n$ ssi $b = a \pmod n$

Si $a = b \pmod n$ et $b = c \pmod n$,
alors $a = c \pmod n$

La congruence est une relation d'équivalence

remarque sur la relation d'équivalence

Cryptology & Security Initiative

Les classes d'équivalence de cette relation (classes de reste modulo n) sont

$$\mathbb{Z}/n\mathbb{Z} = \{0,1,\ldots,n-1\}$$

calcul avec les congruences

Cryptology & Security Initiative

Soient a, b, c, d et n des entiers.

Si $a = b \pmod n$, alors $ac = bc \pmod n$

Si $a = b \pmod n$ et $c = d \pmod n$,
alors $a + c = b + d \pmod n$

Si $a = b \pmod n$ et $c = d \pmod n$,
alors $ac = bd \pmod n$

Si $a = b \pmod n$, alors $a^k = b^k \pmod n$

1^{er} exemple de calcul

Cryptology & Security Initiative

$$16 = -1 \pmod{17}$$

$$\Rightarrow 16^2 \ (= 256) = 1 \pmod{17}$$

2^{eme} exemple de calcul

Cryptology & Security Initiative

$$2^4 = 16 = 1 \pmod 5$$

$$\Rightarrow 2^8 = (2^4)^2 = 1 \pmod 5$$

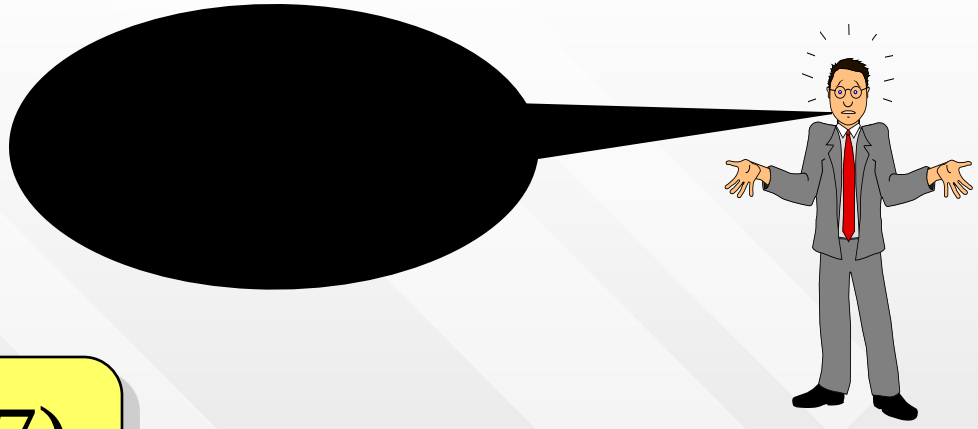
$$\Rightarrow 2^{12} = 2^8\, 2^4 = 1 \pmod 5$$

$$\Rightarrow 2^{4k} = 1 \pmod 5, \text{ pour tout } k.$$

3^{eme} exemple de calcul

Cryptology & Security Initiative

$2^3 = 8 \pmod{17}$



$2^4 = 16 \pmod{17}$

$2^5 = 32 = 15 \pmod{17}$

$2^{10} = (2^5)^2 = 15^2 = 4 \pmod{17}$

$2^{30} = (2^{10})^3 = 4^3 = 64 = 13 \pmod{17}$

$\Rightarrow 2^{32} = 2^{30}\,2^2 = 13 \cdot 4 = 52 = 1 \pmod{17}$

plus de calcul

Cryptology & Security Initiative

Soient a, b, c, d et n des entiers.

Si $a = b \pmod n$ et $d|n$,
alors $a = b \pmod d$

Si $a \cdot c = b \cdot c \pmod n$,
alors $a = b \pmod{n/\mathrm{pgcd}(c,n)}$

exemple

Il faut montrer que
$n^3 - n = 0 \pmod 3$. Or,

$$\mathbb{Z}/3\mathbb{Z} = \{0,1,2\}$$

$0^3-0 = 0 \pmod 3$

$1^3-1= 0 \pmod 3$

$2^3-2= 0 \pmod 3$

$$\Rightarrow 3 | n^3 - n$$

équivalence

Cryptology & Security Initiative

Si pgcd (m,n) = 1, alors

$$[a = b \pmod m \text{ et } a = b \pmod n] \Leftrightarrow a = b \pmod{m \cdot n}$$

Si p et q sont des nombres premiers, alors

$a^2 = 1 \pmod{pq}$ ssi

$a^2 = 1 \pmod p$ et $a^2 = 1 \pmod q$

définition de l'inverse modulo n

Cryptology & Security Initiative

Soient a et n des entiers. Un entier a' est dit inverse de a modulo n si et seulement si $a \cdot a' = a' \cdot a = 1 \pmod{n}$.

On dit que a est inversible modulo n, si a admet un inverse modulo n.

Si a admet un inverse modulo n, alors cet inverse est unique.

1^{er} exemple d'inverse

Cryptology & Security Initiative

$$2 \cdot 6 = 1 \pmod{11}$$

$\Rightarrow$ l'inverse de 2 modulo 11 est 6

$\Rightarrow$ l'inverse de 6 modulo 11 est 2

2^{eme} exemple d'inverse

Cryptology & Security Initiative

$$3 \cdot 3 = 1 \pmod 8$$

$$\Rightarrow \text{l'inverse de 3 modulo 8 est 3}$$

3^{eme} exemple d'inverse

Cryptology & Security Initiative

$$2x = 1 \pmod 8 \Rightarrow 8|2x-1$$

Or, 2x-1 est impair et 8 est pair

$\Rightarrow$ 2 n'admet pas d'inverse modulo 8

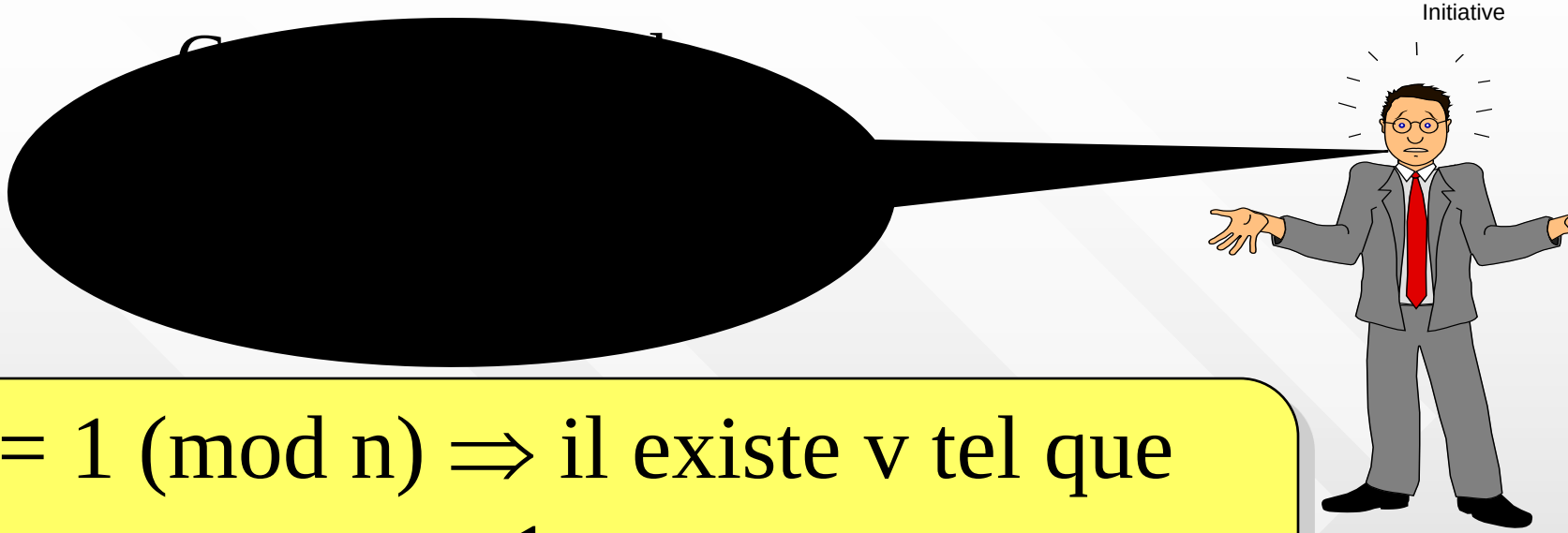
éléments inversibles

Cryptology & Security Initiative

Les éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$ sont les entiers premiers avec n et forment un groupe pour la multiplication noté $(\mathbb{Z}/n\mathbb{Z})^*$.

Si p est un nombre premier, alors $\mathbb{Z}/p\mathbb{Z}$ est un corps.

calcul de l'inverse modulo n



$ux = 1 \pmod{n} \Rightarrow$ il existe v tel que $ux - 1 = vn$

Théorème de Bezout $\Rightarrow$ existence de x et v, si u et n sont premiers entre eux.

Calcul pratique : algorithme d'Euclide étendu

chapitre 3

Cryptology & Security Initiative

Divisibilité

Arithmétique modulaire

Théorèmes fondamentaux

Le RSA

Le problème de la factorisation

le théorème chinois

Cryptology & Security Initiative

Si $m_1, m_2, ..., m_k$ sont des entiers deux à deux disjoints entre eux et si $a_1, a_2, ..., a_k$ sont des entiers quelconques, il existe un entier x tel que, pour tout $i = 1, ..., k$

$$x = a_i \pmod{m_i}$$

corollaire

Cryptology
& Security
Initiative

$$\text{Si } n = \prod_{i=1}^{k} p_i^{\alpha_i}, \text{ alors}$$

$$\mathbb{Z}/n\mathbb{Z} \simeq \prod_{i=1}^{k} \mathbb{Z}/p_i^{\alpha_i}\mathbb{Z}.$$

théorème de Fermat

Cryptology & Security Initiative

Si p est un nombre premier, alors

$a^p = a \pmod p$, pour tout a.

Si pgcd(a,p)=1,

$a^{p-1} = 1 \pmod p$.

exemple 1

$$\text{Fermat} \Rightarrow 2^{12} = 1 \pmod{13}$$

$$50 = 4 \cdot 12 + 2$$

$$\Rightarrow 2^{50} = (2^{12})^4 \, 2^2 = 1 \cdot 4 = 4 \pmod{13}$$

$$\text{Fermat} \Rightarrow 3^{12} = 1 \pmod{13}$$

$$\Rightarrow 3^{50} = (3^{12})^4 \, 3^2 = 1 \cdot 9 = 9 \pmod{13}$$

$$\Rightarrow 2^{50} + 3^{50} = 4 + 9 = 13 = 0 \pmod{13}$$

exemple 2

$$\text{Fermat} \Rightarrow 3^{36} = 1 \pmod{37}$$

$$372 = 10 \cdot 36 + 12$$

$$3^4 = 81 = 7 \pmod{37} \Rightarrow 3^{12} = 7^3 = 7 \cdot 49 = 7 \cdot 12 = 10 \pmod{37}$$

$$\Rightarrow 3^{372} = (3^{36})^{10} 3^{12} = 1 \cdot 10 = 10 \pmod{37}$$

$$\Rightarrow 3^{372} = 10 \pmod{37}$$

l'indicateur d'Euler

Cryptology & Security Initiative

On note $\varphi(n)$ le nombre d 'éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$.
La fonction φ est appelée l'indicateur d 'Euler.

Exemple : $n = 8$

Les éléments inversibles modulo 8 dans $\{0,1,2,\ldots,7\}$ sont $\{1, 3, 5, 7\}$

$\Rightarrow \varphi(8) = 4$

autres exemples

Cryptology & Security Initiative

p premier $\Rightarrow \text{pgcd}(p,a) = 1, \forall a \in \{1,\ldots,p-1\}$

$\Rightarrow \varphi(p) = p-1$

p premier, r entier

$\Rightarrow \varphi(p^r) = p^r - p^{r-1} = p^r (1-1/p)$

propriétés de l'indicateur d'Euler

Cryptology & Security Initiative

Si pgcd(m,n) = 1,
alors $\varphi(m \cdot n) = \varphi(m) \cdot \varphi(n)$.

Si $n = p_1^{\alpha_1} \cdots p_k^{\alpha_k}$, alors

$$\varphi(n) = \prod_{k=1}^{n} p_i^{\alpha_{i-1}} (p_i - 1)$$

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

exemple d'application

Cryptology & Security Initiative

$$\varphi(29 \cdot 5^2) = \varphi(29).\varphi(5^2)$$
$$= 28 \cdot 5^2(1\text{-}1/5)$$
$$= 28 \cdot 20$$
$$= 560$$

Théorème d'Euler

Si a et n sont des entiers premiers entre eux, alors $a^{\varphi(n)} = 1 (\mathrm{mod}\, n)$.

chapitre 4

Cryptology & Security Initiative

Divisibilité

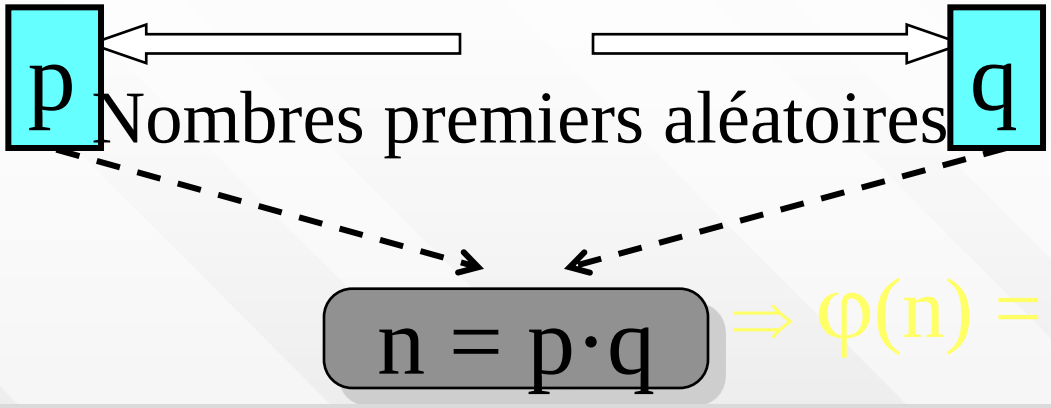
Arithmétique modulaire

Théorèmes fondamentaux

Le RSA

Le problème de la factorisation

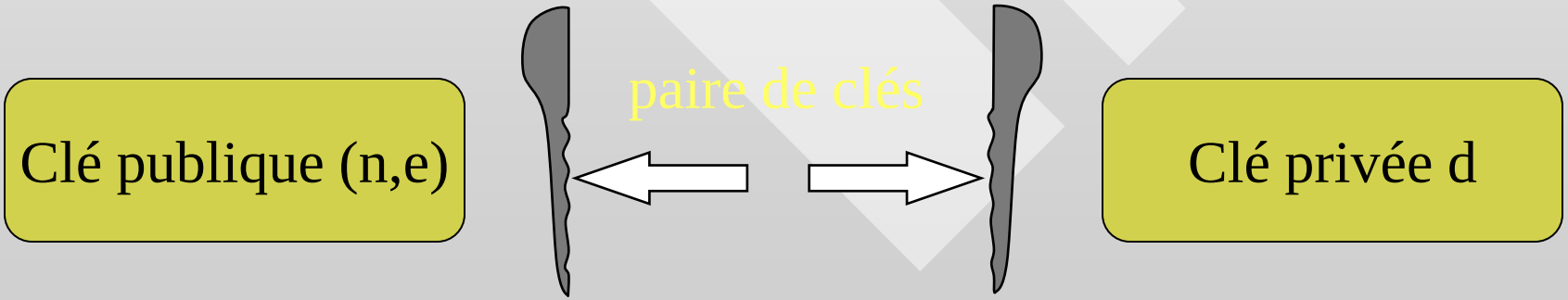
génération des clés



On choisit e tel que

$1 < e < \varphi(n)$ et $\text{pgcd}(e, \varphi(n)) = 1$

On calcule d tel que $e.d = 1 \pmod{\varphi(n)}$



exemple de clés

Cryptology & Security Initiative

$p = 11$ et $q = 23$

$\Rightarrow n = 253$ et $(p-1)(q-1) = 10 \cdot 22 = 2^2 \cdot 5 \cdot 11$

Le plus petit choix pour e est $e = 3$

$\Rightarrow d = 147$

Remarque : Parfois, on remplace la fonction $\varphi(n)$ par $\lambda(n) = (p-1)(q-1)/2$
$\Rightarrow$ accélération du déchiffrage

procédure de chiffrage

Cryptology & Security Initiative

Message m
$0 \leq m < n$

Texte chiffré
$c = m^e \pmod{n}$

Exemple:

$n = 253$ et $e = 3$

$m = 165 \Rightarrow c = 165^3 \pmod{253}$

$\Rightarrow c = 110$

procédure de déchiffrage

Cryptology & Security Initiative

Texte chiffré c

Message original
$m = c^d \pmod n$

Exemple:

$n = 253, e = 3, d = 147$

$c = 110 \Rightarrow m = 110^{147} \pmod{253}$

$\Rightarrow m = 165$

preuve de la procédure de déchiffrage

remarque sur le déchiffrage

Cryptology & Security Initiative

On peut réduire considérablement le temps de calcul du déchiffrage en utilisant le théorème chinois.

~~$m = c^d \pmod n$~~

$$m_p = c^d \pmod p$$

$$m_q = c^d \pmod q$$

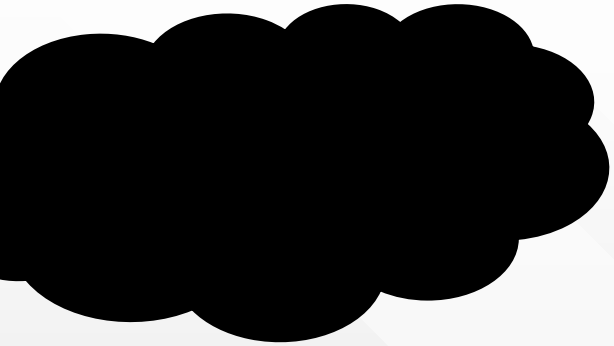
Théorème des restes chinois

$$m = m_p \pmod p$$

$$m = m_q \pmod q$$

chapitre 5

Cryptology & Security Initiative



- Divisibilité
- Arithmétique modulaire
- Théorèmes fondamentaux
- Le RSA

Le problème de la factorisation

théorèmes

Cryptology & Security Initiative

Théorème d'Euclide :
Il existe une infinité de nombres premiers.

Théorème arithmétique fondamental:
Tout nombre entier peut être décomposé de façon unique comme produit de nombres premiers.

méthodes de factorisation

Cryptology & Security Initiative

Méthode exhaustive :
On divise n par tous les entiers entre 1 et $\sqrt{n}$ jusqu'à trouver un diviseur d.
Puis, on recommence avec n/d.

Méthode de Pierre de Fermat (1601-1665) :
Si $n = a^2 - b^2$, alors $n = (a-b)(a+b)$.
En pratique, on calcule $a^2 - n$, ou a^2 est le plus petit carré > n.
Si c'est un carré, on a trouvé.
Sinon, on essaie le carré prochain.

RSA – algorithmes de calcul associés

par Pascal ZEIHEN

Table des Matières

1. Introduction: le crypto-système RSA

Cryptology & Security Initiative

Fabrication des clés:

p, q deux nombres premiers

$n = p \cdot q$ module

$\lambda(n)$ fonction d'Euler

$\lambda(n) = \min\{ k > 0 : a^k = 1 \pmod n, \text{ pour } a = 1, 2, ..., n-1 \}$
$\lambda(n)$ est un diviseur de $\varphi(n)/2$

on peut choisir p et q pour maximiser $\lambda(n) = (p-1)(q-1)/2$

1. Introduction: le crypto-système RSA

Cryptology & Security Initiative

Fabrication des clés:

choisir e tel que pgcd(e, $\lambda(n)$) = 1

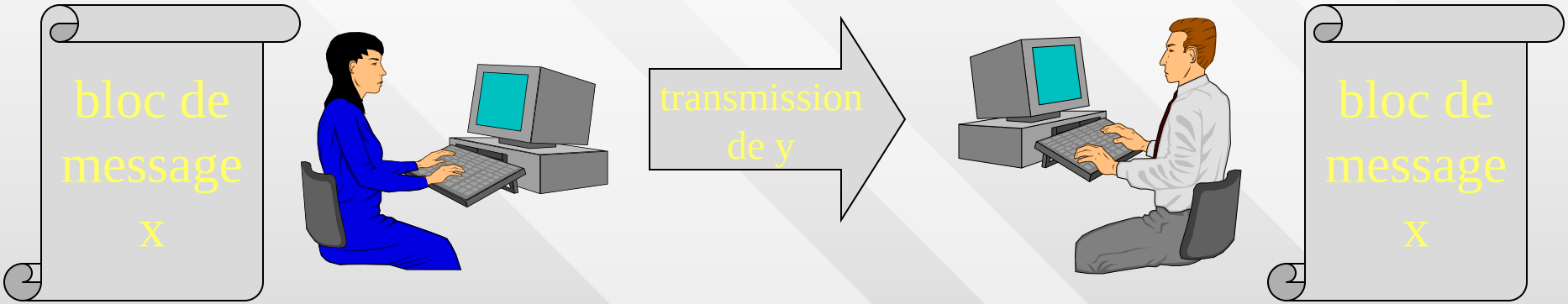
calculer d = 1/e (mod $\lambda(n)$)

clé publique: (n, e) clé privée: d

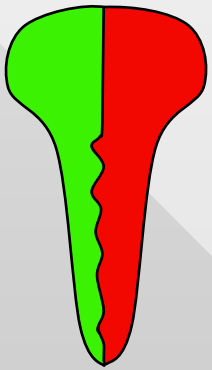
1. Introduction: le crypto-système RSA

Cryptology & Security Initiative

Chiffrement: **Déchiffrement:**



$y = x^e \pmod n$



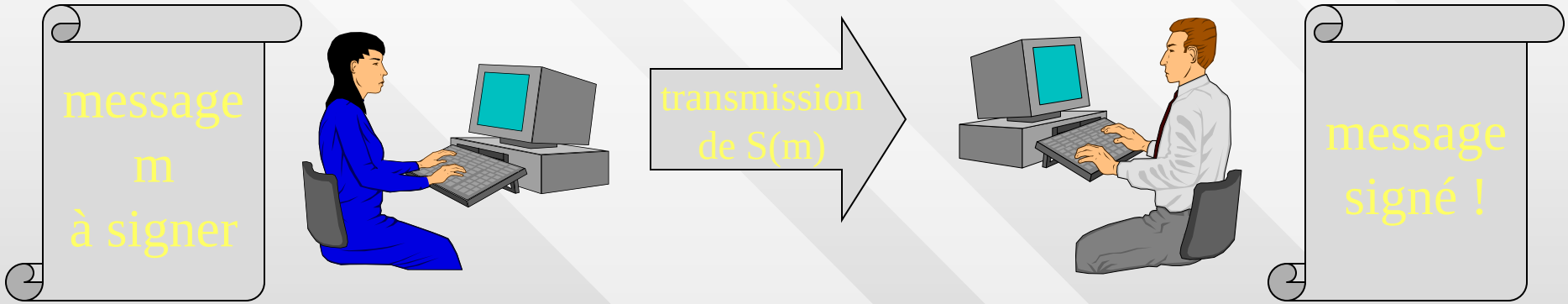
$x = y^d \pmod n$

ou $\pmod{p, q}$

1. Introduction: le crypto-système RSA

Cryptology & Security Initiative

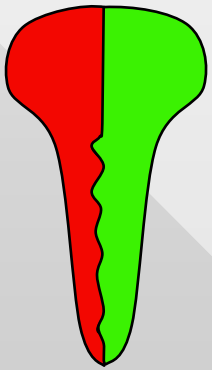
Signature: **Vérification:**



$S(m) = m^d \pmod n$

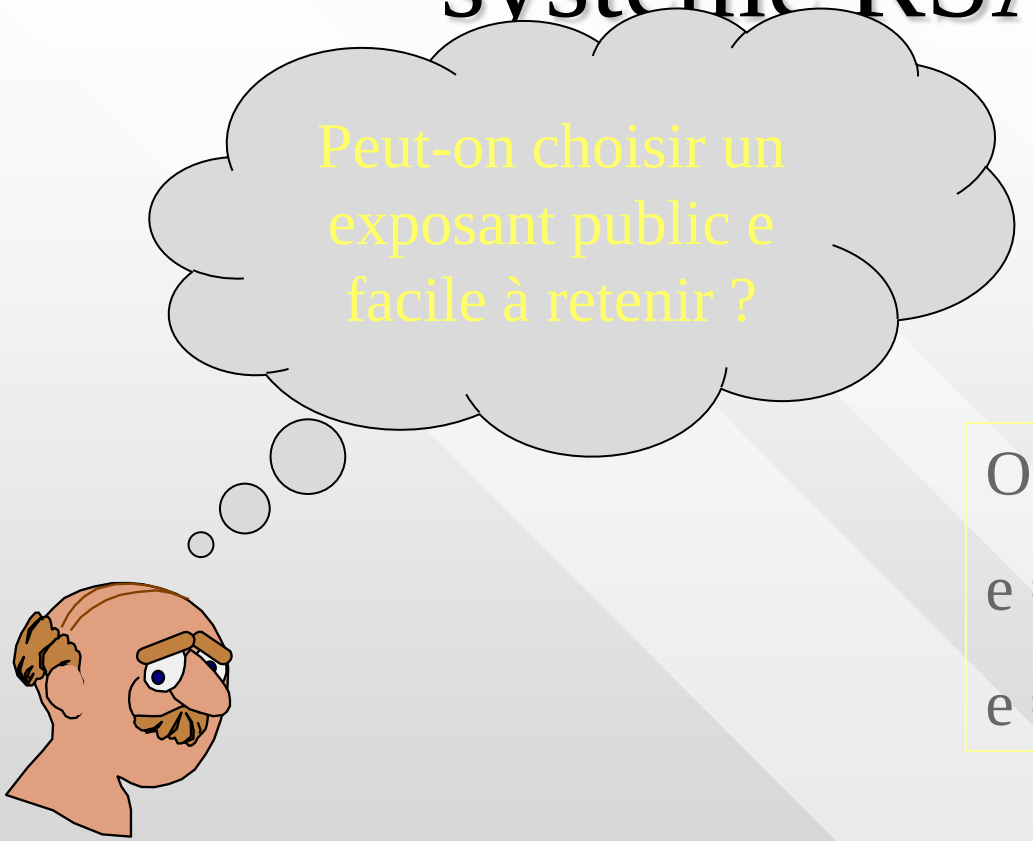
$m = S(m)^e \pmod n$

ou $\pmod{p, q}$



1. Introduction: le crypto-système RSA

Cryptology & Security Initiative



On choisit souvent:

$e = 3$

$e = 2^{16} + 1 = 65537$

2. Exponentiation modulo n

Cryptology & Security Initiative

Comment effectuer les calculs

$$y = x^e \pmod{n}$$

et

$$x = y^d \pmod{n}$$

rapidement ?

exponentiations

- savoir multiplier rapidement
- réduire le nombre de multiplications

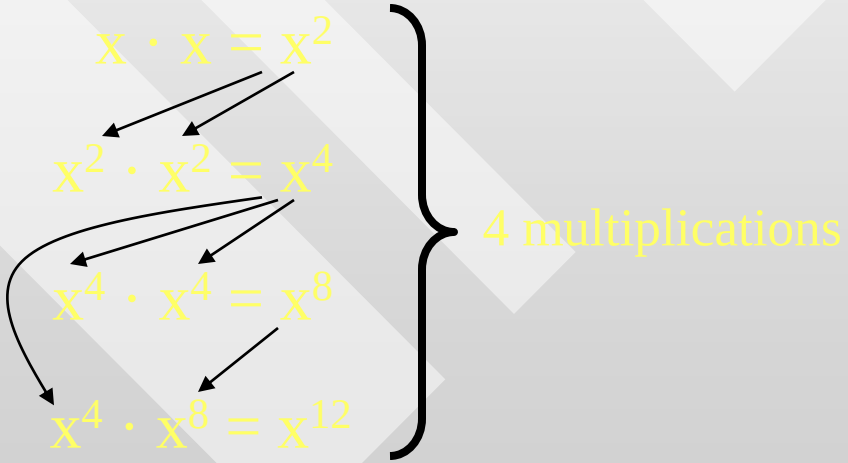
Nombre de multiplications à effectuer

Cryptology & Security Initiative

Exemple: calculer x^{12}

Comment ne pas faire: $x^{12} = \underbrace{x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x \cdot x}_{\text{11 multiplications !}} \cdot x$

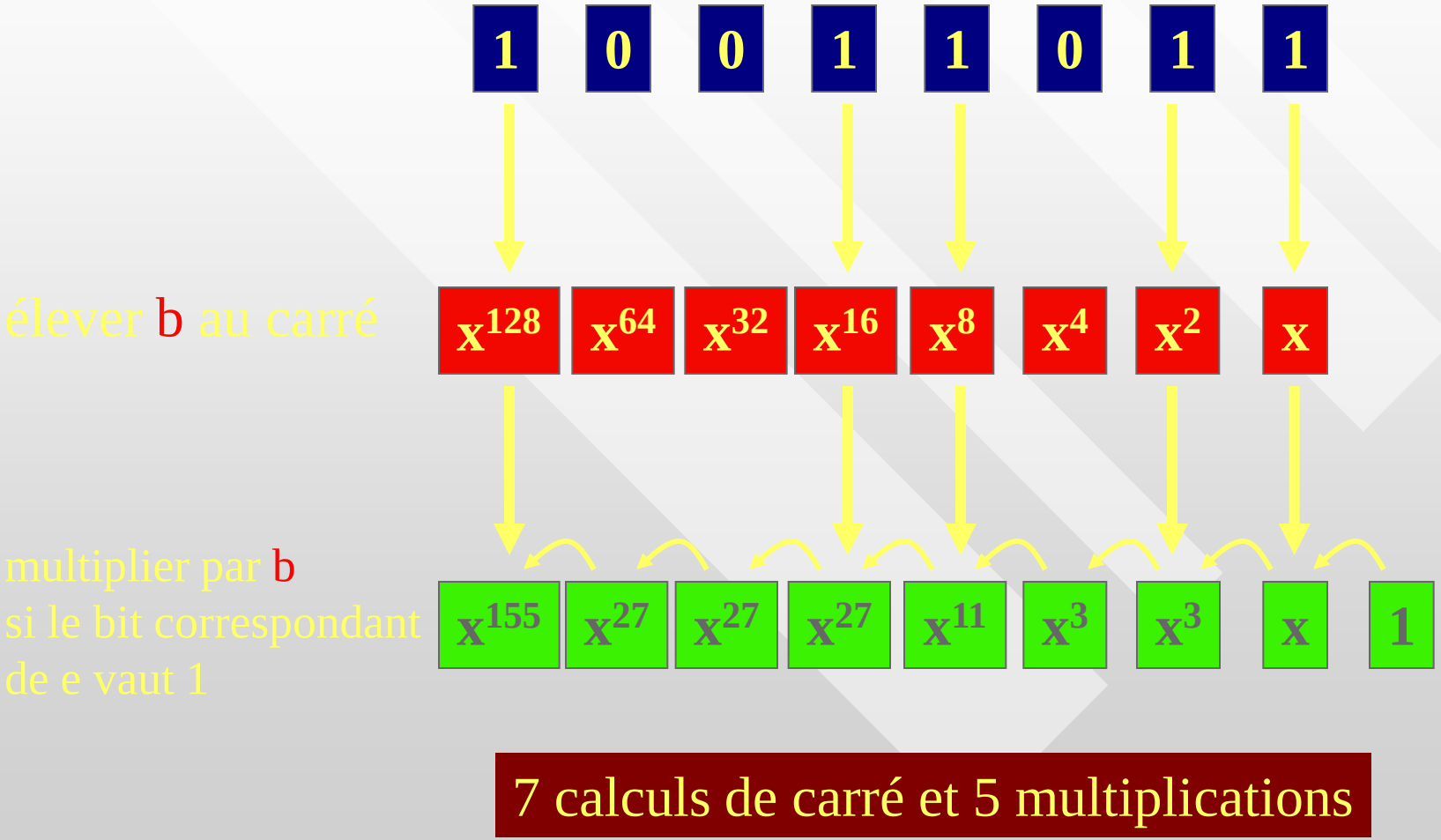
Comment mieux faire:



Algorithme « R2L exponentiation »

Cryptology & Security Initiative

Exemple: exposant $e = 155 = (10011011)_2$



7 calculs de carré et 5 multiplications

Algorithme « R2L exponentiation »

Cryptology & Security Initiative

Exposants fréquents:

$e = 3 = (11)_2$: un calcul de carré et une multiplication

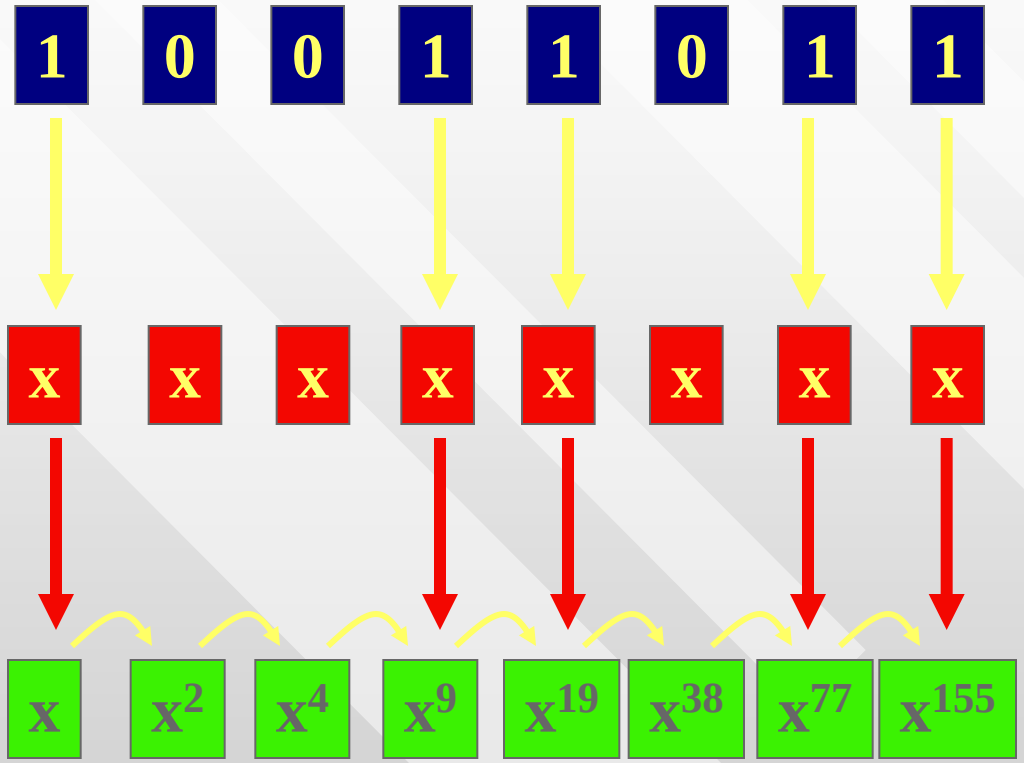
$e = 65537 = (10000000000000001)_2$: 16 calculs de carré et une multiplication

Inconvénient: les multiplications se font avec des facteurs de plus en plus grands !

Algorithme « L2R exponentiation »

Cryptology & Security Initiative

Exemple: exposant $e = 155 = (10011011)_2$



encore 7 calculs de carré et 5 multiplications

Obtention de la clé de déchiffrement

Cryptology & Security Initiative

Fabrication des clés:

choisir e tel que pgcd(e, $\lambda(n)$) = 1

calculer d = 1/e (mod $\lambda(n)$)

clé publique: (n, e) clé privée: d

- calcul du plus grand commun diviseur
- déterminer d et k tels que $d \cdot e + k \cdot \lambda(n) = 1$

3. Algorithme d'Euclide étendu

Cryptology & Security Initiative

Soient $a \geq b > 0$ deux nombres naturels.

Algorithme d'Euclide étendu:

déterminer u, v et g tels que

$$a \cdot u + b \cdot v = g$$

avec $g = \text{pgcd}(a, b)$

Algorithme d'Euclide

Cryptology & Security Initiative

Soient $a \geq b > 0$ deux nombres naturels.
Déterminer $g = \text{pgcd}(a, b)$.

Division euclidienne (avec reste):

$$a = b \cdot q + r$$

quotient

reste ($0 \leq r < b$)

si $r = 0$: $\text{pgcd}(a, b) = b$
si $r > 0$: $\text{pgcd}(a, b) = \text{pgcd}(b, r)$; remplacer a par b, et b par r

Algorithme d'Euclide étendu

Exemple: a = 6020 et b = 1001

a	=	b	·	q	+	r
6020	=	1001	·	6	+	14
1001	=	14	·	71	+	7
14	=	7	·	2	+	0

pgcd(6020, 1001) = 7

$14 = 6020 - 1001 \cdot 6$

$1001 - 14 \cdot 71 = 7$

$1001 - (6020 - 1001 \cdot 6) \cdot 71 = 7$

$6020 \cdot (-71) + 1001 \cdot 427 = 7$

u = -71

v = 427

Algo d'Euclide étendu: propriétés

Cryptology & Security Initiative

Algorithme extrêmement rapide !

p. ex.: a avec 300 chiffres décimaux, b choisi au hasard:

au maximum 1445 divisions euclidiennes

souvent b est faible (RSA: b = e = 3 ou 65537)

p. ex.: b = 65537 et a choisi au hasard:

en moyenne 9 divisions euclidiennes

4. Opérations arithmétiques

Cryptology & Security Initiative

On a déjà vu:

- comment transformer l'exponentiation en multiplications
- comment calculer le pgcd et un inverse-modulo

Il reste à voir:

- la multiplication de grands nombres
- le calcul modulo un grand nombre

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

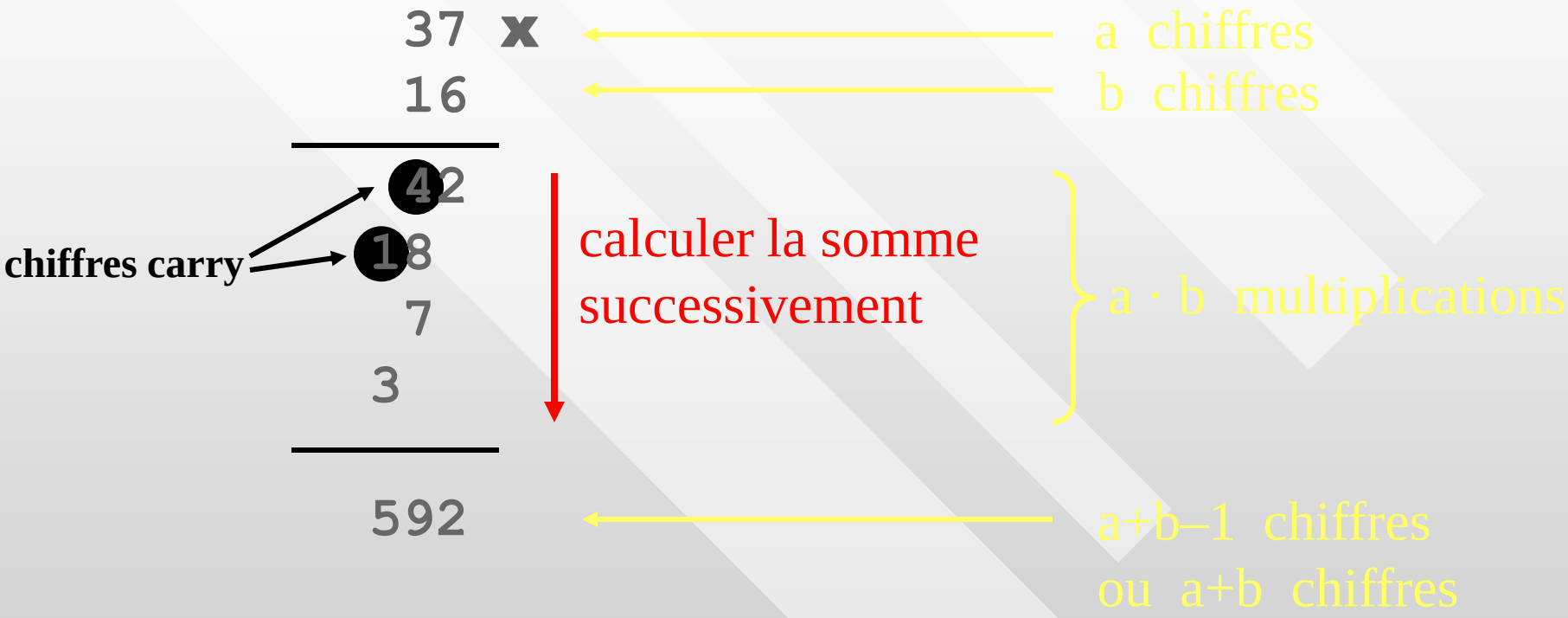
$$x = y^d \pmod n$$

reste d'une division de grands nombres (opération la plus coûteuse !)

Multiplication de grands nombres

Cryptology & Security Initiative

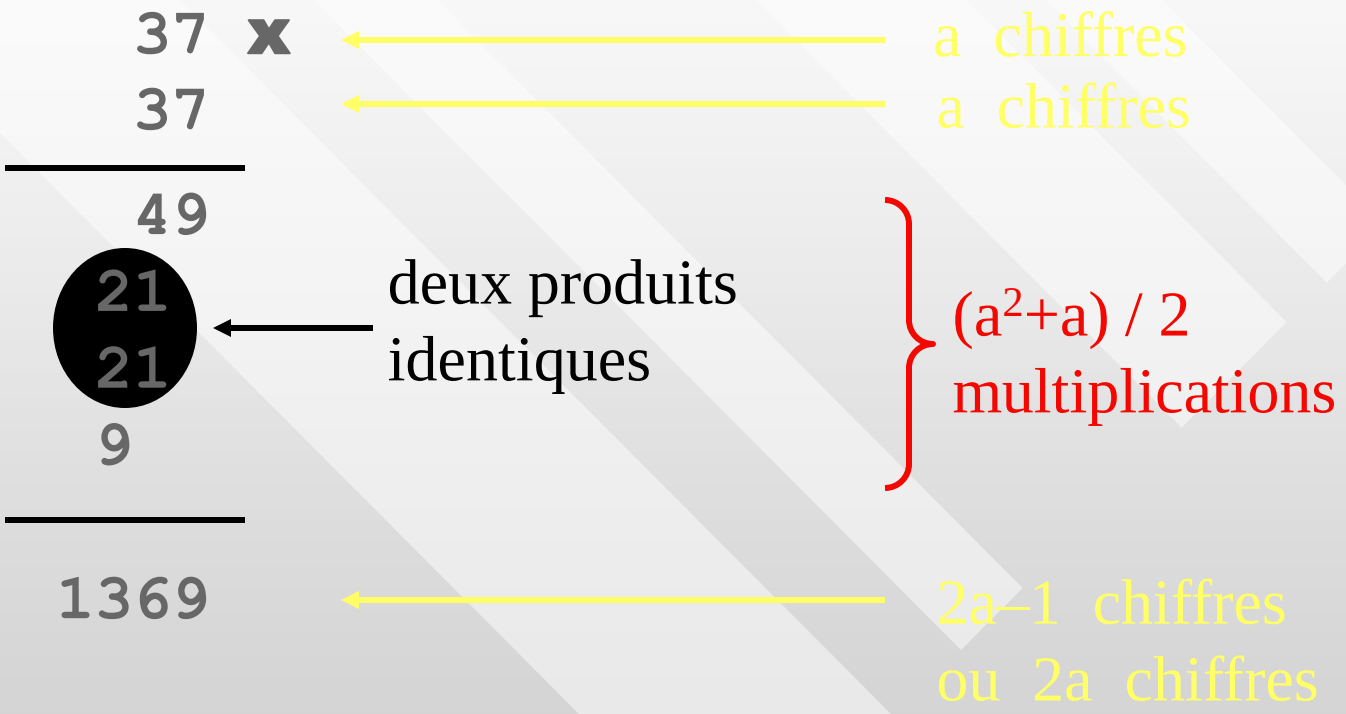
Disposition pratique:



Calcul du carré de grands nombres

Cryptology & Security Initiative

Disposition pratique:



Multiplication de grands nombres

Cryptology & Security Initiative

Il existe des techniques plus rapides et plus compliquées:

- transformation de Fourier rapide (FFT)
- méthode de Karatsuba

MAIS: leur implémentation (sur carte à puce p.ex.) serait très difficile, souvent impossible !

SOLUTION: combiner l'algorithme classique à la réduction de Montgomery

Réduction de Montgomery

Cryptology & Security Initiative

Comment calculer $x \pmod n$ ou $x \cdot y \pmod n$ rapidement ?

Soient $x \geq 0$, $n \geq 2$.

Idée: choisir R tel que $R > n$, $\text{pgcd}(n, R) = 1$ et $x < nR$.

Calculer: $x / R \pmod n$

$xR^{-1} \pmod n$ est appelé réduction de x modulo n par rapport à R

Les calculs se font rapidement lorsque $R = 2^k$;
cela est possible lorsque n est impair !

Multiplication de Montgomery

Cryptology & Security Initiative

Si x et y ont a chiffres en base b : choisir $R = b^a$

On peut calculer $xyR^{-1} \pmod n$ avec

- 2a(a+1) multiplications de chiffres en base b ; rapide !
- 2a divisions par b ; trivial !

(simple manipulation de bits)

MAIS: l'accélération des calculs de carré sur base des doubles produits n'est plus possible.

Exponentiation modulo n

Cryptology
& Security
Initiative

Exponentiation de Montgomery

Cryptology & Security Initiative

Exemple: calculer x^{12} (mod n)

Calcul de x^{12} avec l'algorithme L2R:

on calcule successivement

x x^2 x^3 x^6 x^{12}

Calcul de x^{12} avec l'algorithme L2R-Montgomery:

on calcule successivement

$z = xR$ z^2R^{-1} z^3R^{-2} z^6R^{-5} $z^{12}R^{-11}$ (mod n)

Le produit-Montg. de $z^{12}R^{-11}$ par 1 donne $z^{12}R^{-12} = x^{12}$ (mod n)

5. Génération de nombres premiers

Cryptology & Security Initiative

Fabrication des clés:

p, q	deux nombres premiers
$n = p \cdot q$	module
$\lambda(n)$	fonction d'Euler

Comment trouver des nombres premiers très grands ?

$\lambda(n) = \{ k > 0 : a^k = 1 \pmod n$, pour $a = 1, 2, ..., n-1 \}$
$\lambda(n)$ est un diviseur de $\varphi(n)/2$

on peut choisir p et q pour maximiser $\lambda(n) = (p-1)(q-1)/2$

Stratégie à suivre

Cryptology & Security Initiative

1ère étape: choisir un grand nombre au hasard

on obtient un nombre N

2e étape: tester si N est premier (test de primalité)

si N est composé:
remplacer N par N±(petit nombre) et refaire le test

Nombres aléatoires

Cryptology & Security Initiative

Ordinateur, choisis-moi un nombre au hasard !

C'est quoi ça, le hasard ?

nombre **pseudo-** aléatoire = nombre inattendu

Générateurs de nombres pseudo-aléa.

Cryptology & Security Initiative

- Dispositifs physiques
- Algorithmes déterministes

Linear congruential generator:

$$X_i = a \cdot X_{i-1} + c \pmod m$$

nombres trop prévisibles !

- Algorithmes CSPRBG (cryptographically secure pseudo-random bit generator)

générateur basé sur RSA,
générateur Blum-Blum-Shub

ces générateurs sont basés sur des nombres pseudo-aléatoires !

Initialisation des générateurs

Cryptology & Security Initiative

Problème supplémentaire: obtenir une valeur initiale pour faire démarrer le générateur de nombres pseudo-aléatoires

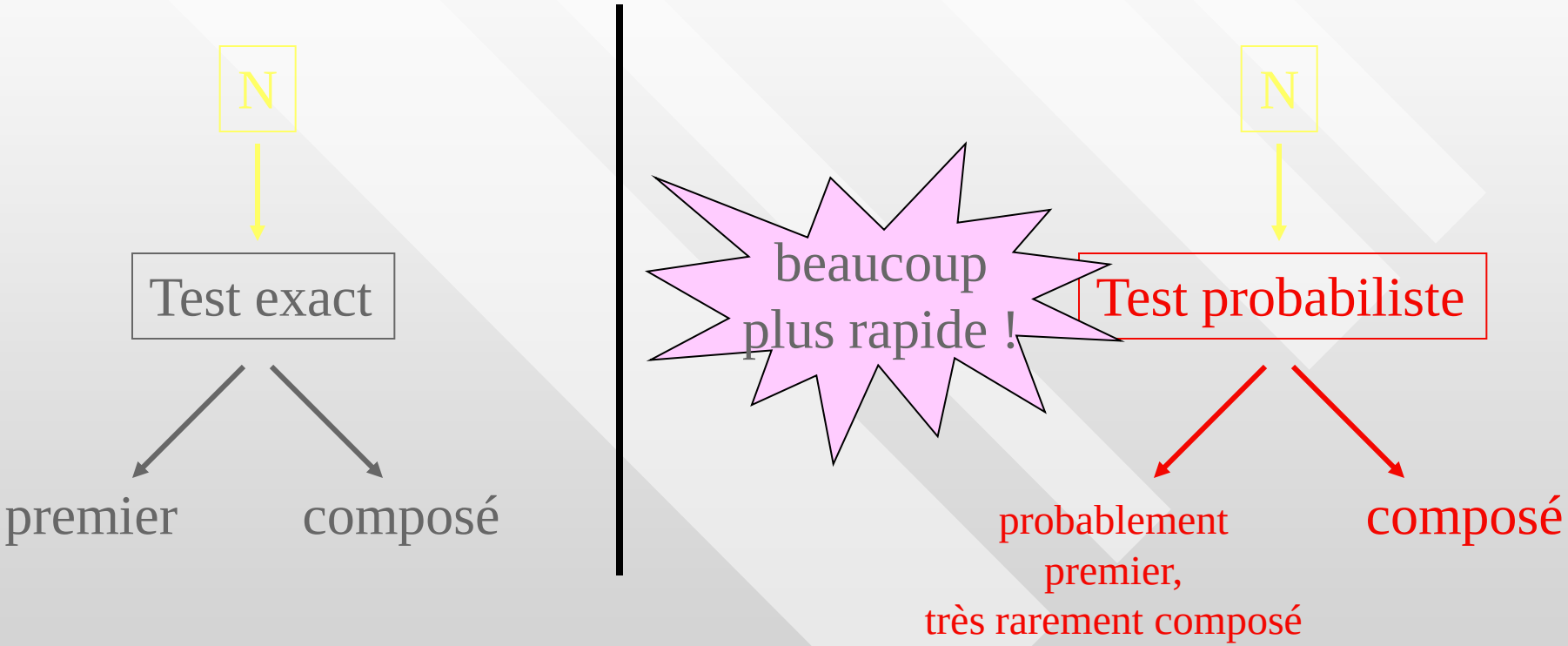
Ne pas se baser simplement sur l'horloge-système (trop prévisible) !

Faire intervenir des effets physiques, le comportement inattendu (pression des touches du clavier, mouvement de la souris) de l'utilisateur, etc.

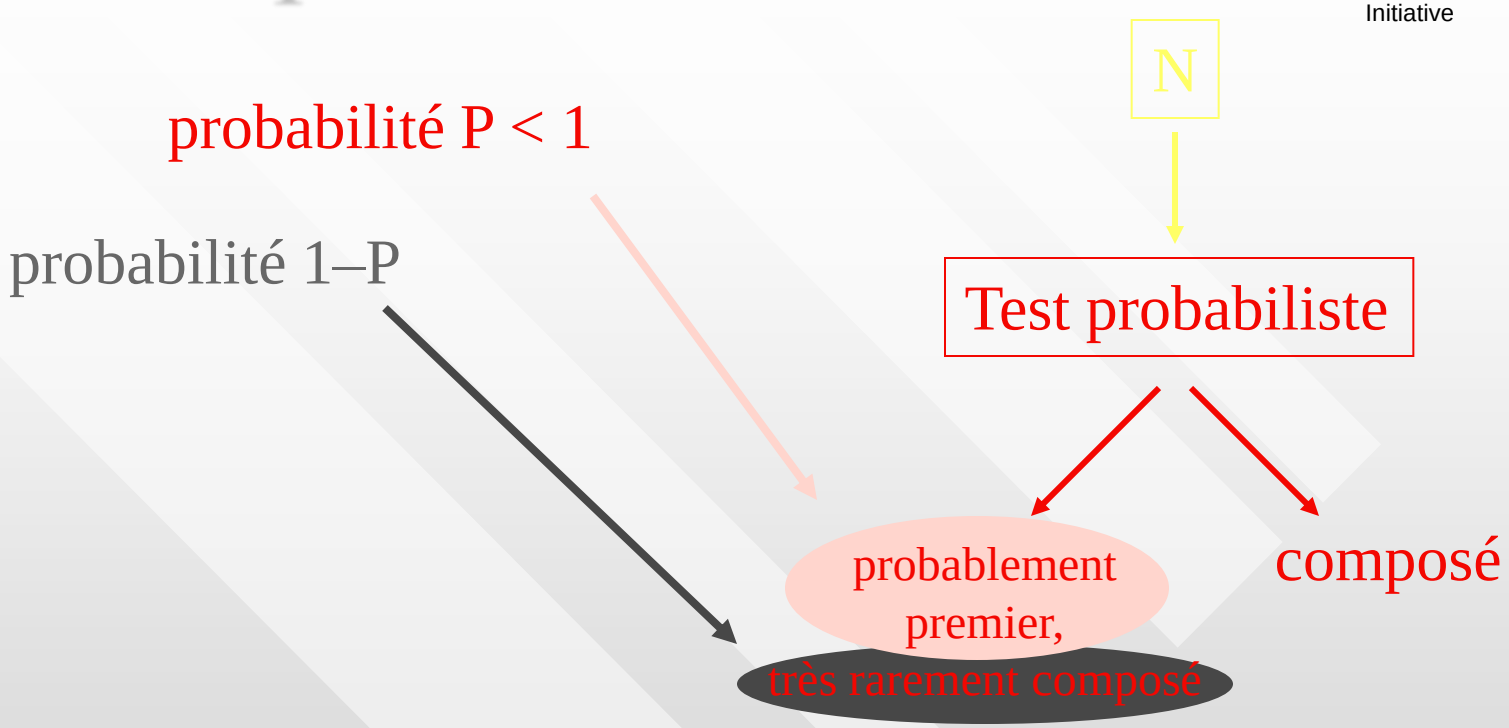
Tests de primalité

Cryptology & Security Initiative

Deux types de tests:



Tests probabilistes



Exemples:

- test de Fermat
- test de Solovay-Strassen — après t essais: $1–P < (1/2)^t$
- test de Miller-Rabin — après t essais: $1–P < (1/4)^t$

Tests exacts

exigent une structure particulière du nombre N

exemple: nombres de Mersenne $M_s = 2^s - 1$

test de Lucas-Lehmer: $\underbrace{2^{13466917} - 1}$ **est premier !**

nombre à 4 053 946 chiffres décimaux

autre exemple: test de Pocklington-Lehmer

la décomposition de $N-1$ doit être connue (partiellement)

6. Factorisation de grands nombres

Cryptology & Security Initiative

Tout nombre naturel $n > 1$ peut être décomposé en produit de nombres premiers

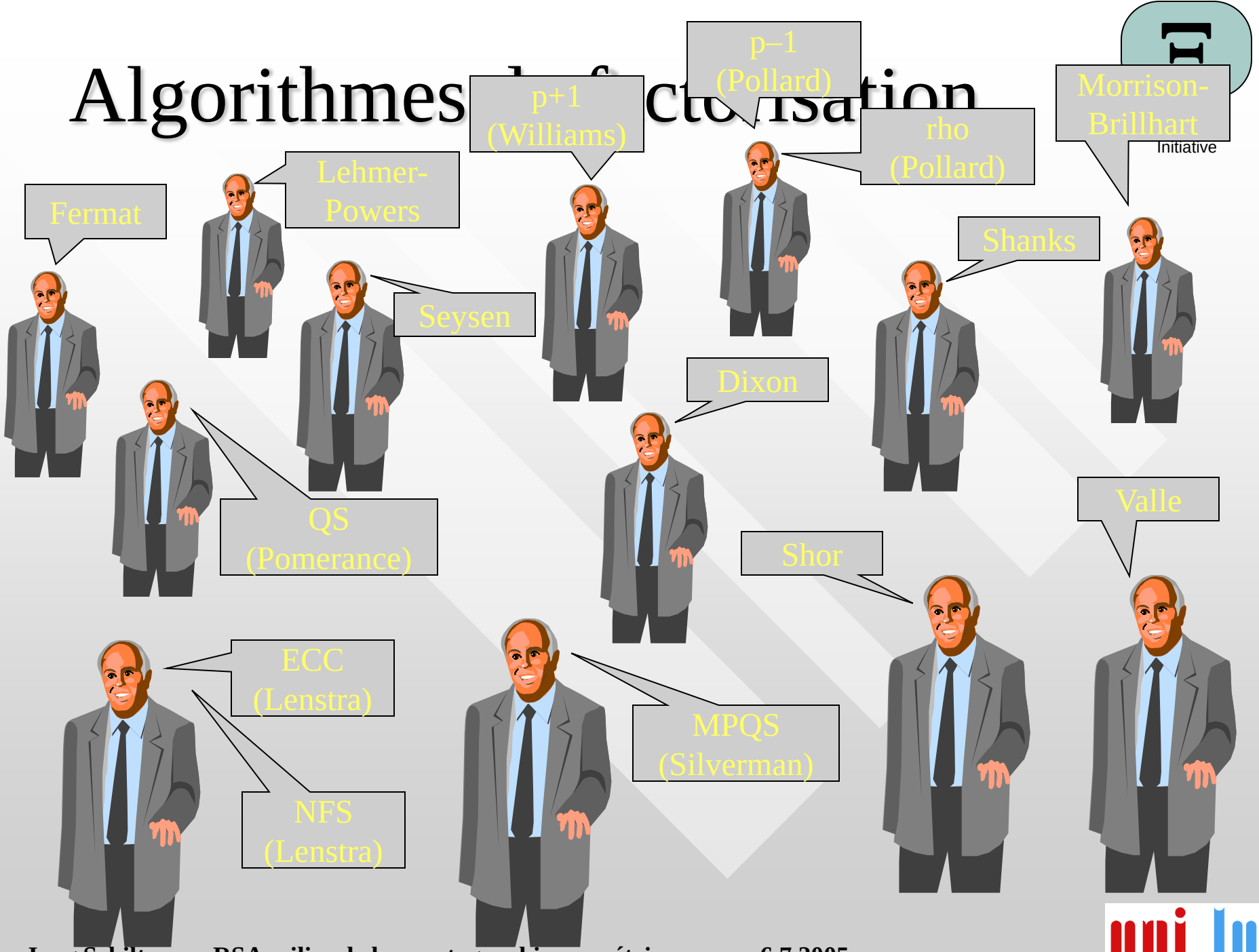
$$p_1 p_2 \cdots p_t \quad \text{avec} \quad p_1 \leq p_2 \leq \ldots \leq p_t$$

Factorisation par divisions:

essayer successivement les diviseurs premiers 2, 3, 5, 7 inférieurs ou égaux à $\sqrt{n}$

algorithme facile à implémenter, mais extrêmement lent si n est grand (plus de 20 chiffres décimaux)

Algorithmes de factorisation



Stratégie raisonnable

Cryptology & Security Initiative

1. factorisation par divisions jusqu'à un seuil s_1
2. ~~méthode rho de Pollard jusqu'à un seuil $s_2 > s_1$~~
3. méthode ECC (Lenstra) jusqu'à un seuil $s_3 > s_2$
4. méthodes QS ou NFS jusqu'à un seuil $s_4 > s_3$

2. méthode p+1 (Williams) ou p–1 (Pollard) lorsque
   les facteurs de p+1 resp. p–1 sont inférieurs à un certain seuil

Exemples spectaculaires

Cryptology & Security Initiative

facteurs à 86 et 73 chiffres

$2^{953} + 1$ (287 chiffres) décomposé en janvier 2002 par NFS

RSA-155 (155 chiffres) décomposé en août 1999 par NFS

RSA-140 (140 chiffres) décomposé en février 1999 par NFS

$(6^{43}–1)^{42}+1$ (1406 chiffres) décomposé par ECC

facteurs à 73 et 54 chiffres

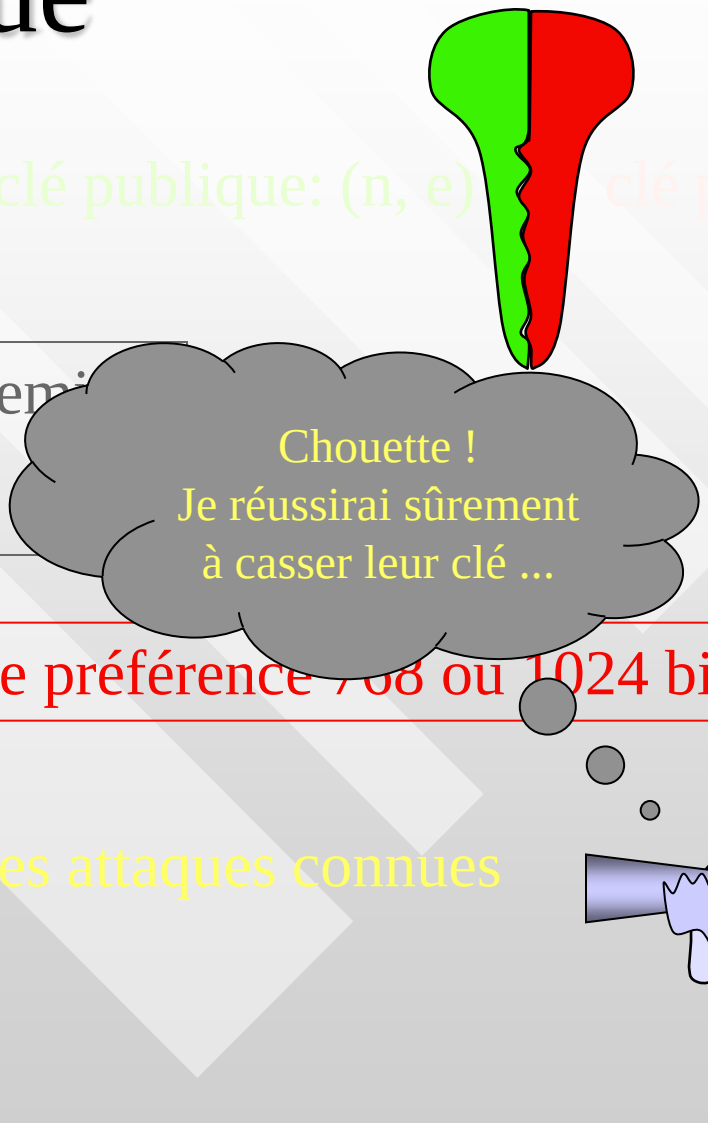
autres exemples: décomposition des nombres de Fermat

7. Génération de clés RSA en pratique

Cryptology & Security Initiative

clé publique: (n, e) clé privée: d

p, q	deux nombres premi
n = p · q	module



taille de n: au moins 512 bits, de préférence 768 ou 1024 bits

choisir p et q de façon à éviter des attaques connues

Consignes sur choix de p et q

Cryptology & Security Initiative

- la différence $p–q$ ne doit pas être proche de 0
- p et q doivent avoir (presque) le même nombre de chiffres
- $\text{pgcd}(p–1, q–1) = 2$
- p et q doivent être des nombres premiers forts:
  p est fort ssi p+1 a un grand facteur et
  $p–1$ a un grand facteur p_1 tel que
  $p_1–1$ a aussi un grand facteur

Autres consignes

Cryptology & Security Initiative

- il faut minimiser le nombre de blocs de message qui ne seront pas chiffrés: $(1+\text{pgcd}(p-1, e-1)) \cdot (1+\text{pgcd}(q-1, e-1))$
- la clé d doit être suffisamment grande: $d > n^{0.3}$
- $d \pmod{p-1}$ et $d \pmod{q-1}$ doivent être grands

choisir un bon codage des données (padding)