Mathématiques en technologies de l'information 1

Chapitre 2
Notions de Cryptographie

Quelques notions de théorie des nombres supplémentaires et utiles

L'utilisation de nombres et de calculs remonte aux origines de l'humanité.

Les premières traces de leur étude remontent à 1800 Av J.-C. (liste de triplets tels que $a^2 + b^2 = c^2$).

Il existe de nombreux problèmes dits «ouverts», facile à comprendre mais qui n'ont pas encore été prouvés.

… mais surtout, la théorie des nombres est une base indispensable à la cryptographie !!!

Quelques notions de théorie des nombres supplémentaires et utiles

Exemples :

- Existe-t-il une infinité de nombres *premiers jumeaux* ? (p premier et $p + 2$ également) ?

- Conjecture de Goldbach : tout entier pair ≥ 4 peut s'écrire comme la somme de deux premiers.

Un exemple très célèbre

<u>Conjecture de Fermat (1601-1655):</u>

L'équation $x^n + y^n = z^n$ n'a pas de solution entière strictement positive pour $n > 2$.

Fermat dit : *« J'ai trouvé une merveilleuse démonstration de cette proposition, mais la marge est trop étroite pour la contenir. »*

La preuve officielle n'arrivera qu'en 1995 par Andrew Wiles, après 350 ans de tentatives infructueuses… Et ladite preuve s'étend sur plus de 1000 pages !

Quelques principes élémentaires

- Il existe une infinité de nombre premiers, mais…

- Existe-t-il un moyen de les générer ?
  Aucune formule n'existe pour TOUS les générer,
  Il est prouvé qu'il n'existe aucun polynôme non constant $P(n)$ tel que $P(n)$ soit premier pour tout n assez grand
  On ignore s'il existe un polynôme permettant de générer une infinité de nombres premiers !

- Crible d'Eratosthène
  Dans une tables de nombres de 1 à N, éliminer successivement tous les multiples des nombres premiers antérieurs

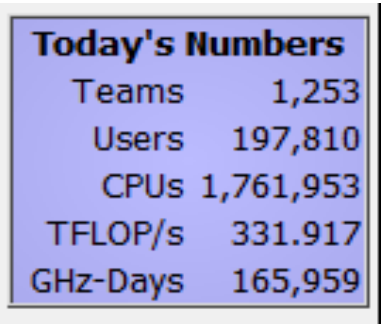
Exercice : appliquer le crible d'Eratosthène pour N = 100.

Quelques principes élémentaires

- Pour vérifier qu'un nombre n est premier, aucun nombre premier de 2 à $\sqrt{n}$ n'est diviseur de n.

- Quel est le plus grand nombre premier connu ?

On étudie les nombres de Mersenne $2^p - 1$ avec p premier
Projet GIMPS (Great Internet Mersenne Prime Search)
www.mersenne.org (oct. 2018)
(Mersenne est un mathématicien Français du XVII^e s.)



26 Déc. 2017 : $2^{77,232,917} - 1$ est premier !
C'est le 50^e nombre de Mersenne !
23.2 Mio de caractères !
Plus de 9000 pages !

Quelques principes élémentaires

Définition:

Deux nombres entiers a et b sont dits premiers entre eux si

$$PGCD(a, b) = 1$$

PGCD = Plus Grand Commun Diviseur

Le PGCD et les nombres premiers entre eux sont des fondamentaux pour la cryptographie (nous verrons certains exemples plus tard).

Méthode d'Euclide

<u>Calcul du PGCD selon la méthode d'Euclide</u>

Pour le calcul de $PGCD(a, b)$, nous supposerons (sans perte de généralité) que $a \geq b$

1. Calculer $r = a \; mod \; b$
2. Tant que $(r > 1)$ faire
   - Stocker $res \leftarrow r$
   - Redéfinir les variables $a \leftarrow b$, $b \leftarrow r$
   - $r = a \; mod \; b$
3. Si $r = 0$, alors $PGCD(a, b) = res$
   Si $r = 1$, alors $PGCD(a, b) = 1$ (a et b sont premiers entre eux)

Théorème Fondamental de l'arithmétique (Gauss, 1777-1855)

Tout nombre entier $n \in \mathbb{N}, n \geq 2$ peut être écrit comme un produit fini de nombres premiers de manière unique (à l'ordre des facteurs près).

NOTE : cela implique que 1 n'est PAS un nombre premier !!!

(la preuve ne sera pas donnée dans ce cours !)

Celle-ci s'appelle la *décomposition en facteurs premiers* !

Factorisation en nombres premiers

La factorisation d'un nombre $a \in \mathbb{N}$ se base sur du «trial-and-error» en passant, itérativement, les diviseurs premiers…

- Tant que $a \; mod \; 2 = 0$, effectuer $a \leftarrow a/2$;
- Si $a \; mod \; 2 \neq 0$, alors passer au premier suivant (3) et tant que $a \; mod \; 3 = 0$, effectuer $a \leftarrow a/3$;
- Continuer jusqu'à ce que $a = 1$.

Cette approche est extrêmement coûteuse !!!

Exemple

$$32 = 2 \times 2 \times 2 \times 2 \times 2 = 2^5$$

$$168 = 2 \times 2 \times 2 \times 3 \times 7 = 2^3 \times 3 \times 7$$

$$770 = 2 \times 5 \times 7 \times 11$$

Calcul du PPCM

$PPCM(a, b)$ (**P**lus **P**etit **C**ommun **M**ultiple) de deux nombres a et b est le plus petit entier naturel r tel que

a divise r ($r \; mod \; a = 0$) et
b divise r ($r \; mod \; b = 0$).

Comment calculer le PPCM ?

Méthode 1:
A l'aide de la décomposition en facteurs premiers de a et b :
$PPCM(a, b)$ est le produit du plus grand nombre de tous les facteurs présent dans les deux décompositions.

Exemple

$32 = 2 \times 2 \times 2 \times 2 \times 2 = 2^5$

$168 = 2 \times 2 \times 2 \times 3 \times 7 = 2^3 \times 3 \times 7$

$770 = 2 \times 5 \times 7 \times 11$

- $PPCM(32, 168) = 2^5 \times 3 \times 7 = 672$
- $PPCM(168, 770) = 2^3 \times 3 \times 5 \times 7 \times 11 = 9240$

Propriété intéressante

Pour toute paire de nombres $a, b \in \mathbb{N}$, on a que

$$a \times b = PGCD(a, b) \times PPCM(a, b)$$

Autrement dit : si on connait $a \times b$ et $PGCD(a, b)$, alors

$$PPCM(a, b) = \frac{a \times b}{PGCD(a, b)}.$$

Théorème de Bachet-Bézout (XVIIe s)

Soient $a, b \in \mathbb{Z}^*$, alors il existe deux entiers relatifs $u, v \in \mathbb{Z}$ tels que

$$a \times u + b \times v = PGCD(a, b).$$

Conséquence : si a et b sont premiers entre eux, alors $PGCD(a, b) = 1$.

Autrement dit, l'équation $a \times u + b \times v = 1$ a (au moins) une solution entière si a et b sont premiers entre eux !

Méthode d'Euclide étendue

Soient $a, b \in \mathbb{N}^*$ avec $a > b$ (sans perte de généralité), trouvons les coefficients de Bézout $x, y \in \mathbb{Z}^*$ tels que

$$PGCD(a, b) = x \times a + y \times b$$

<u>Algorithme:</u>

0) $r_0 = a = x_0 \times a + y_0 \times b \qquad r_0 = a, x_0 = 1, y_0 = 0$

1) $r_1 = b = x_1 \times a + y_1 \times b \qquad r_1 = b, x_1 = 0, y_1 = 1$

Tant que $r_i \neq 0$ faire

i) Résoudre $r_i = r_{i-2} - q_i \times r_{i-1}$ (par la division euclidienne)
Poser $x_i = x_{i-2} - q_i \times x_{i-1}$

Quand $r_i = 0$, alors $r_{i-1} = PGCD(a, b)$, $x = x_{i-1}$ et $y = y_{i-1}$

Exemple

Calculer $PGCD(168,\ 68) = x \times a + y \times b$

Algorithme:

0) $r_0 = 168,\ x_0 = 1, y_0 = 0$

1) $r_1 = 68,\ \ x_1 = 0, y_1 = 1$

2) $\mathrm{r}_2 = 32 = 168 - 2 \times 68\ (q_2 = 2)$

Poser $x_2 = x_0 - 2 \times x_1 = 1 - 2 \times 0 = 1$

et $y_2 = y_0 - 2 \times y_1 = 0 - 2 \times 1 = -2$

Exemple $PGCD(168,\ 68) = x \times a + y \times b$

3) $\mathrm{r}_3 = 4 = 68 - 2 \times 32$ $(q_3 = 2)$
Poser $x_3 = x_1 - 2 \times x_2 = 0 - 2 \times 1 = -2$
et $y_3 = y_1 - 2 \times y_2 = 1 - 2 \times (-2) = 5$

4) $\mathrm{r}_4 = 0 = 32 - 8 \times 4$

Réponse : $PGCD(186, 68) = 4 = -2 \times 168 + 5 \times 68$,
Les coefficients de Bézout sont $[-2; 5]$.

Arithmétique modulaire

L'un des fondements de la cryptographie se base sur l'arithmétique modulaire.

Inventé par Carl Friedrich Gauss en 1801 (à 24 ans).

Tout se base sur l'opérateur «***modulo***» que nous noterons mod.

En calcul $\mathrm{mod}\ n$, il n'y a que n valeurs possibles : $0, \ldots, n-1$.

Définition :
Pour $a, b, n \in \mathbb{Z}$,
Alors $a \bmod n = b$ si $a = b + k \times n$ pour un $k \in \mathbb{Z}$.

Exemples

Calculez les résultats suivants :

- $11 \bmod 2 = 1$,
- $2 \bmod 11 = 2$,
- $-2 \bmod 11 = 9$,
- $-11 \bmod 2 = 1$.

Arithmétique modulaire

L'opérateur «modulo» s'applique en soi à un seul élément.

Il permet toutefois de définir une nouvelle égalité que nous noterons $\equiv_n$. Dans ce cas,

$$a \equiv_n b$$

est équivalent à prendre le modulo de deux côtés de l'égalité classique :

$$a \pmod{n} = b \pmod{n}$$

Donc a et b sont égaux «à multiples de n près».

Nous disons aussi que a et b sont ***congruents*** $\mod n$.

Propriétés de la congruence

Dans les opérations modulo, les principes connus restent vrais :

- Commutatif: $(a + \mathrm{b}) \equiv_n (b + a)$
  $(a\mathrm{b}) \equiv_n (b \times a)$

- Associatif : $(a + \mathrm{b}) + \mathrm{c} \equiv_n a \bmod n + (b + \mathrm{c})$
  $(a \times \mathrm{b}) \times \mathrm{c} \equiv_n a \times (b \times \mathrm{c})$

- Distributif: $(a + \mathrm{b}) \times \mathrm{c} \equiv_n (a \times c) + (b \times \mathrm{c})$

Un des principaux résultats de l'arithmétique modulaire est que l'opérateur peut également être distribué sur + et – en utilisant la congruence :

$$a + b \bmod n \equiv_n [(a \bmod n) + (b \bmod n)]$$
$$a \times b \bmod n \equiv_n [(a \bmod n) \times (b \bmod n)]$$

Propriétés du modulo

Pourquoi ce dernier résultat est-il intéressant ?

Calculons $1341 \times 679 \; mod \; 22$.

Le calcul brut nous donne que :

$$1341 \times 679 \; mod \; 22 = 910'539 \; mod \; 22 = 41'388 \times 22 + 3 \; mod \; 22 = 3.$$

Or en utilisant la distributivité du modulo, nous obtenons que :

$$1341 \; mod \; 22 = 60 \times 22 + 21 \; mod \; 22 = 21$$
$$679 \; mod \; 22 = 30 \times 22 + 19 \; mod \; 22 = 19$$

Donc le résultat est

$$21 \times 19 \; mod \; 22 = 399 \; mod \; 22 = 18 \times 22 + 3 \; mod \; 22 = 3.$$

Propriétés du modulo

Certes, le résultat précédent n'est pas très parlant.

Rappelons nous toutefois que sur une machine à calculer, la grandeur des nombres est limitée. La distributivité du modulo nous permet d'éviter d'utiliser de très grands nombres.

De plus, essayons de calculer

$$13^{241} \; mod \; 17.$$

Calculer 23^{241} dépasse largement la capacité des nombres à 64 bits.

Algorithme d'Exponentiation Rapide

Objectif : calculer $a^X \ mod \ n$ (avec typiquement X très grand)

Données : $a, X \in \mathbb{N}$ et $n \in \mathbb{N}^*$

Algorithme :

Si $a = 0$ => $r = 0$ **STOP**
Si $X = 0$ => $r = 1$ **STOP**
Poser $r = 1, e = X$ et $b = a \ mod \ n$,

while $(e > 0)$ **do**
$\boldsymbol{y} = e \ mod \ 2$
$r = (r \times b^y) \ mod \ n$
$b = (b \times b) mod \ n$
$e = e/2$ // division entière !
endwhile
STOP la réponse est r

Exemple

Cacluler $6^{55}\ mod\ 23$

$r = 1$, $e = 55$ et $b = 6$,

1. $r = 1 \times 6\ mod\ 23 = 6$ (e impair),
   $b = 6 \times 6\ mod\ 23 = 13$,
   $e = \frac{55}{2} = 27.$

2. $r = 6 \times 13\ mod\ 23 = 78\ mod\ 23 = 9$ (e impair),
   $b = 13 \times 13\ mod\ 23 = 78\ mod\ 23 = 8$,
   $e = \frac{27}{2} = 13.$

3. $r = 9 \times 8\ mod\ 23 = 72\ mod\ 23 = 3$ (e impair),
   $b = 8 \times 8\ mod\ 23 = 64\ mod\ 23 = 18$,
   $e = \frac{13}{2} = 6.$

Exemple

4. $r = 3$ (e pair – pas de changement),
   $b = 18 \times 18 \; mod \; 23 = 324 \; mod \; 23 = 2$,
   $e = \frac{6}{2} = 3$.

5. $r = 3 \times 2 \; mod \; 23 = 6 \; mod \; 23 = 6$ (e impair),
   $b = 2 \times 2 \; mod \; 23 = 4 \; mod \; 23 = 4$,
   $e = \frac{3}{2} = 1$.

6. $r = 6 \times 4 \; mod \; 23 = 1$ (e impair),
   $b = 4 \times 4 \; mod \; 23 = 16 \; mod \; 23 = 16$,
   $e = \frac{1}{2} = 0$.

STOP $6^{55} \; mod \; 23 = 1$**.**

D'où vient cette algorithme

L'idée est d'écrire X sous forme binaire et de calculer le produit des puissances modulo n :

$(55)_{10} = (110111)_2$

Donc $6^{55} = 6^{32} \times 6^{16} \times 6^4 \times 6^2 \times 6^1$

L'algorithme calcule simultanément $6^{2^i} \bmod n$ et les résultat intermédiaire

	$(6^{16})^2$	$(6^8)^2$	$(6^4)^2$	$(6^2)^2$	$(6^1)^2$	6^1
$(55)_{10}$	1	1	0	1	1	1
$mod\ 23$	$2 \times 2 \equiv_{23} 4$	$18 \times 18 \equiv_{23} 2$	$8 \times 8 \equiv_{23} 18$	$13 \times 13 \equiv_{23} 8$	$6 \times 6 \equiv_{23} 13$	6
r (résultat intermédiaires)	$6 \times 4 \equiv_{23} 1$	$3 \times 2 \equiv_{23} 6$	3 (pas de calcul)	$9 \times 8 \equiv_{23} 3$	$6 \times 13 \equiv_{23} 9$	6

Propriétés du modulo

Pourtant, le calcul peut se faire : notons que $13^2 = 169 = 10 \times 17 - 1$

Par conséquent

$$13^2 \; mod \; 17 = -1 \bmod 17 = 16.$$
$$13^3 \; mod \; 17 = 16 \times 13 \; mod \; 17 = 4$$
$$13^4 \; mod \; 17 = 4 \times 13 \; mod \; 17 = 1$$
$$13^5 \; mod \; 17 = 1 \times 13 \; mod \; 17 = 13$$

Il y a donc une séquence – et toutes les puissances paires suivent la suite $13^{2i} \; mod \; 17 = (-1)^i \; mod \; 17.$

Donc

$$13^{241} \; mod \; 17 = (13^2)^{110} \times 13 \; mod \; 17$$
$$= [(13^2 \; mod \; 17)^{120}] \times 13 \; mod \; 17 = (-1)^{120} \times 13 \; mod \; 17$$
$$= 1 \times 13 \; mod \; 17 = 13 \; !!!$$

Nous avons réussi à calculer à la main le modulo d'un nombre qu'un ordinateur usuel ne saurait pas représenter !!!

Existence d'un opposé

Pour l'addition, il est clair que pout tout $a, n \in \mathbb{Z}^*$, il existe un élément opposé $-a$ de telle sorte que

$$a + (-a) \; mod \; n = 0.$$

Notez toutefois qu'il existe, en fait, une infinité d'opposés à a, qui sont tous de la forme $-a + k \times n$, pour $k \in \mathbb{Z}$.

Existence d'un inverse

Qu'en est-il de l'inverse ?

Pour la multiplication classique, nous savons que tout élément $a \in \mathbb{Z}$ a un inverse tel que $a \times a^{-1} = 1$. Cependant $a^{-1} \notin \mathbb{Z}$: l'inverse n'est PAS un nombre entier.

Qu'en est-il du modulo ?

Selon le théorème de Bachet-Bézout, nous savons que pour deux nombre $a, b \in < \mathbb{Z}^*$, alors il existe deux entiers relatifs $u, v \in \mathbb{Z}$ tels que

$$a \times u + b \times v = PGCD(a, b).$$

Si ces deux nombres sont premiers entre eux, on sait alors qu'il existe deux entiers relatifs $u, v \in \mathbb{Z}$ tels que

$$a \times u + b \times v = 1.$$

Existence d'un inverse

Quelle intérêt ?

Soient $a, n \in \mathbb{Z}^*$ premiers entre eux, alors nous savons qu'il existe deux entiers $u, v \in \mathbb{Z}$ tels que

$$a \times u + n \times v = 1.$$

Appliquons $mod\ n$ des deux côtés :

$$(a \times u + n \times v)\ mod\ n = 1\ mod\ n.$$

Autrement dit

$$a \times u\ mod\ n = 1.$$

Si $a\ et\ n$ sont premiers entre eux, alors a admet un élément inverse dans $\mathbb{Z}$!!

Existence d'un inverse

Soient $a, n \in \mathbb{Z}^*$, alors a admet un inverse modulaire ***si et seulement si***

$$PGCD(a, n) = 1.$$

Donc, si a et n ne sont PAS premiers entre eux, a n'a pas d'inverse modulaire, modulo n !

Existence d'un inverse

L'existence d'un inverse entier relatif est en soi une propriété remarquable – elle nous permet de grandement simplifier les équations du modulaires.

Exemple :

Résoudre l'équation $14 \times x \equiv_5 3$.

14 et 5 sont premiers entre eux. 14 a donc un inverse $modulo\ 5$, il s'agit de 4 :

En effet : $14 \times 4 \equiv_5 56 \equiv_5 1$!

Il nous suffit alors de multiplier par 4 des deux côtés pour obtenir

$$1 \times x \equiv_5 12,$$

Autrement dit $x \equiv_5 2$.

Et nous vérifions : $14 \times 2 = 28 \equiv_5 3$.

Unicité de l'inverse

L'inverse est-il unique ?

NON : En fait, il y en a une infinité !

Par exemple, si 14 a pour inverse $4\ mod\ 5$, c'est aussi le cas pour $4 + k \times 5$, $k \in \mathbb{Z}$!

En revanche, si $n \in \mathbb{Z}^* \backslash \{-1, 1\}$, alors l'inverse est unique $modulo\ n$: a n'a qu'un seul inverse entre 0 et $n - 1$!

Une curiosité : dans la multiplication, un seul nombre est son propre inverse : $1 = 1^{-1}$!

Ce n'est pas le cas dans l'arithmétique modulaire.

Exemple : 4 est sont propre inverse $modulo\ 5$: en effet $4 \times 4 = 16 \equiv_5 1$!!

Petit Théorème de Fermat

Soit p un nombre premier, alors pour tout $a \in \mathbb{Z}$ non-divisible par p, on a

1. $(a^p)\ mod\ p\ = (a)\ mod\ p,$
2. $(a^{p-1})\ mod\ p\ = 1,$
3. Il existe un entier $k \in \mathbb{N}^*$ tel que $(a^k)\ mod\ p = 1$. De plus, le plus petit $k > 0$ vérifiant cette égalité divise $p - 1$.

Exercice

Vérifiez les propriétés avec les paires suivantes :

1. $a = 7, p = 5,$
2. $a = 10, p = 3.$

Petit Théorème de Fermat

Quel intérêt ?

Avec 2 nombres premiers p et q, on sait qu'on aura forcément que

$$q^{p-1} \equiv_p 1 \,,$$

car p et q étant premiers, on sait que p ne divise pas q!

De plus, il permet de prouver que pour tout $a, n \in \mathbb{N}^*$ tels que $PGCD(a, n) = 1$, on a

$$a^{\varphi(n)} \equiv_n 1,$$

où $\varphi(n)$ est ***l'indice d'Euler*** : si $n \in N$, alors

$\varphi(n) =$ nombre de facteurs premiers à n compris entre 1 et n (inclus).

Notez que si $n = p \times q$ (facteur de deux premiers), alors

$$\varphi(n) = (p-1) \times (q-1).$$

Petit Théorème de Fermat

Indice d'Euler : si $n \in N$, alors

$\varphi(n) =$ nombre de facteurs premiers à n compris entre 1 et n (inclus).

Notez que :

- Si p est un nombre premier, alors

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

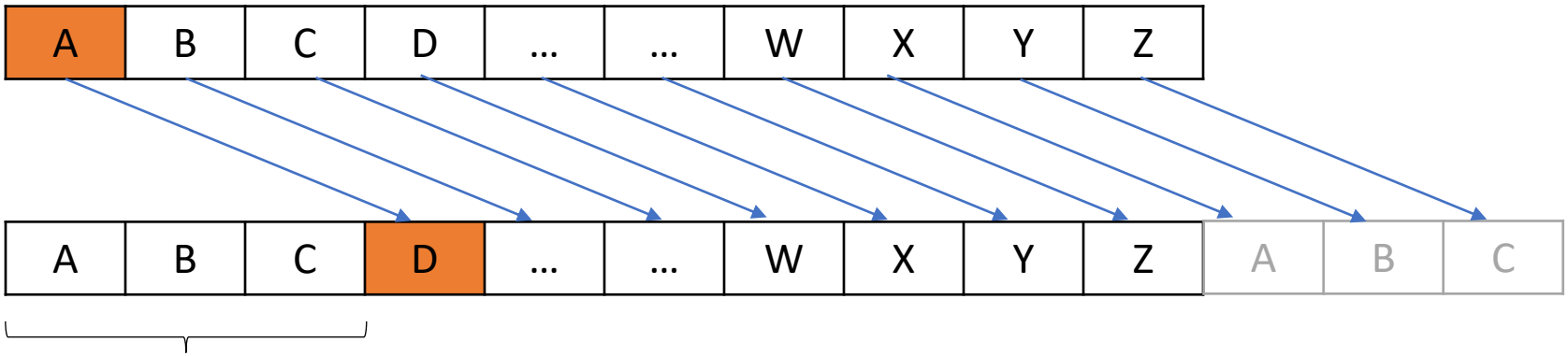
- Si $n = p \times q$ (facteur de deux premiers), alors

$$\varphi(n) = (p - 1) \times (q - 1).$$

Quel est l'intérêt des nombres premiers ?

Il sont à l'origine des méthodes de cryptographie modernes !

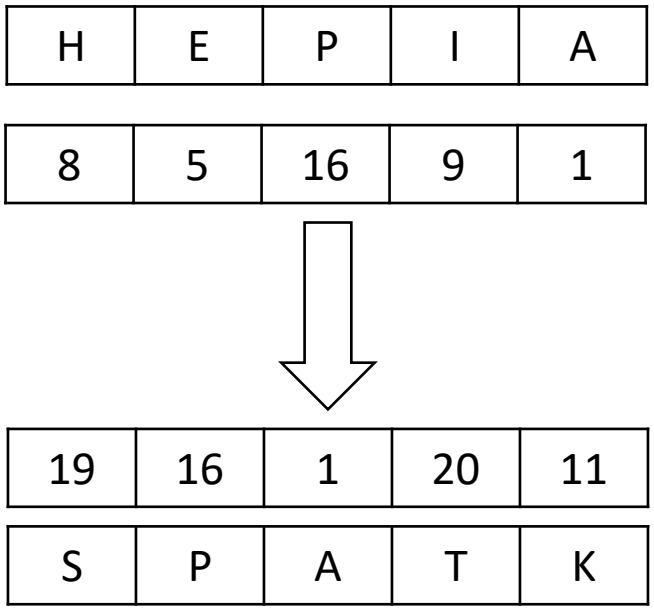
La première méthode de cryptage communément admise est le Chiffre de César (chiffrement par décalage)



Le chiffre de César est ici de 3.

Exemple

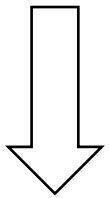
Cryptage de «HEPIA» avec le Chiffre de César = 11
Astuce : passage par les nombres avec a = 1, b=2, …



Exemple

Quelle est la formule pour cette transformation ?

x_1	x_2	x_3	…	x_N



$$y_i = 1 + (x_i + N_{César}) mod\ 26$$

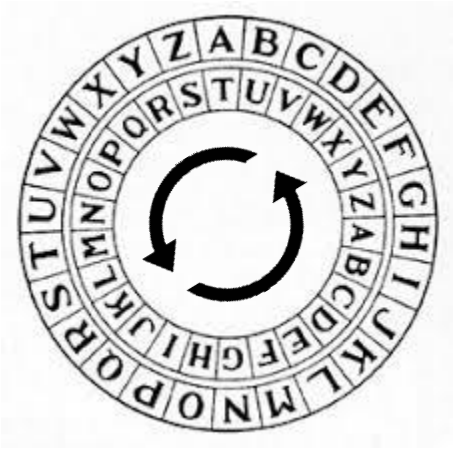
y_1	y_2	y_3	…	y_N

Cela vous semble-t-il connu ?

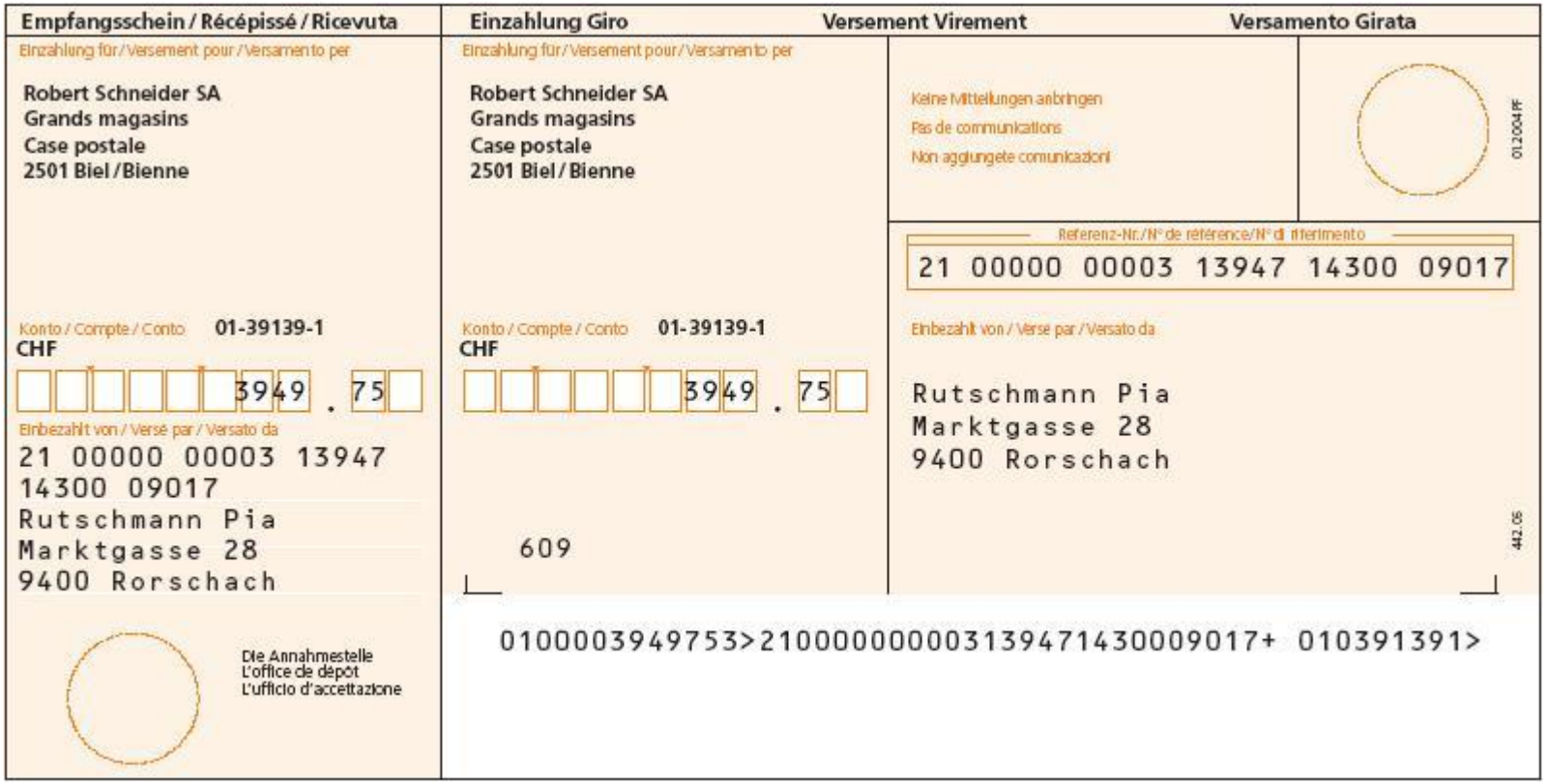
Cela ressemble fortement à la somme sur un nombre de bits finis (avec overflow) !!!

Exercice :

Prouver que tout nombre de César est équivalent à un nombre entre 0 et 26.



Un exemple peu connu mais très utilisé en Suisse…



Empfangsschein / Récépissé / Ricevuta

Einzahlung für / Versement pour / Versamento per

Robert Schneider SA
Grands magasins
Case postale
2501 Biel / Bienne

Konto / Compte / Conto 01-39139-1
CHF

3949 . 75

Einbezahlt von / Versé par / Versato da

21 00000 00003 13947
14300 09017
Rutschmann Pia
Marktgasse 28
9400 Rorschach

Die Annahmestelle
L'office de dépôt
L'ufficio d'accettazione

Einzahlung Giro **Versement Virement** **Versamento Girata**

Einzahlung für / Versement pour / Versamento per

Robert Schneider SA
Grands magasins
Case postale
2501 Biel / Bienne

Konto / Compte / Conto 01-39139-1
CHF

3949 . 75

609

Keine Mitteilungen anbringen
Pas de communications
Non aggiungete comunicazioni

012004 FF

Referenz-Nr. / N° de référence / N° di riferimento

21 00000 00003 13947 14300 09017

Einbezahlt von / Versé par / Versato da

Rutschmann Pia
Marktgasse 28
9400 Rorschach

442.05

0100003949753>210000000003139471430009017+ 010391391>

Algorithme Modulo 10 récursif

Il permet de vérifier si une séquence de chiffres contient une erreur.

- Soit la table de reports définie comme suit :

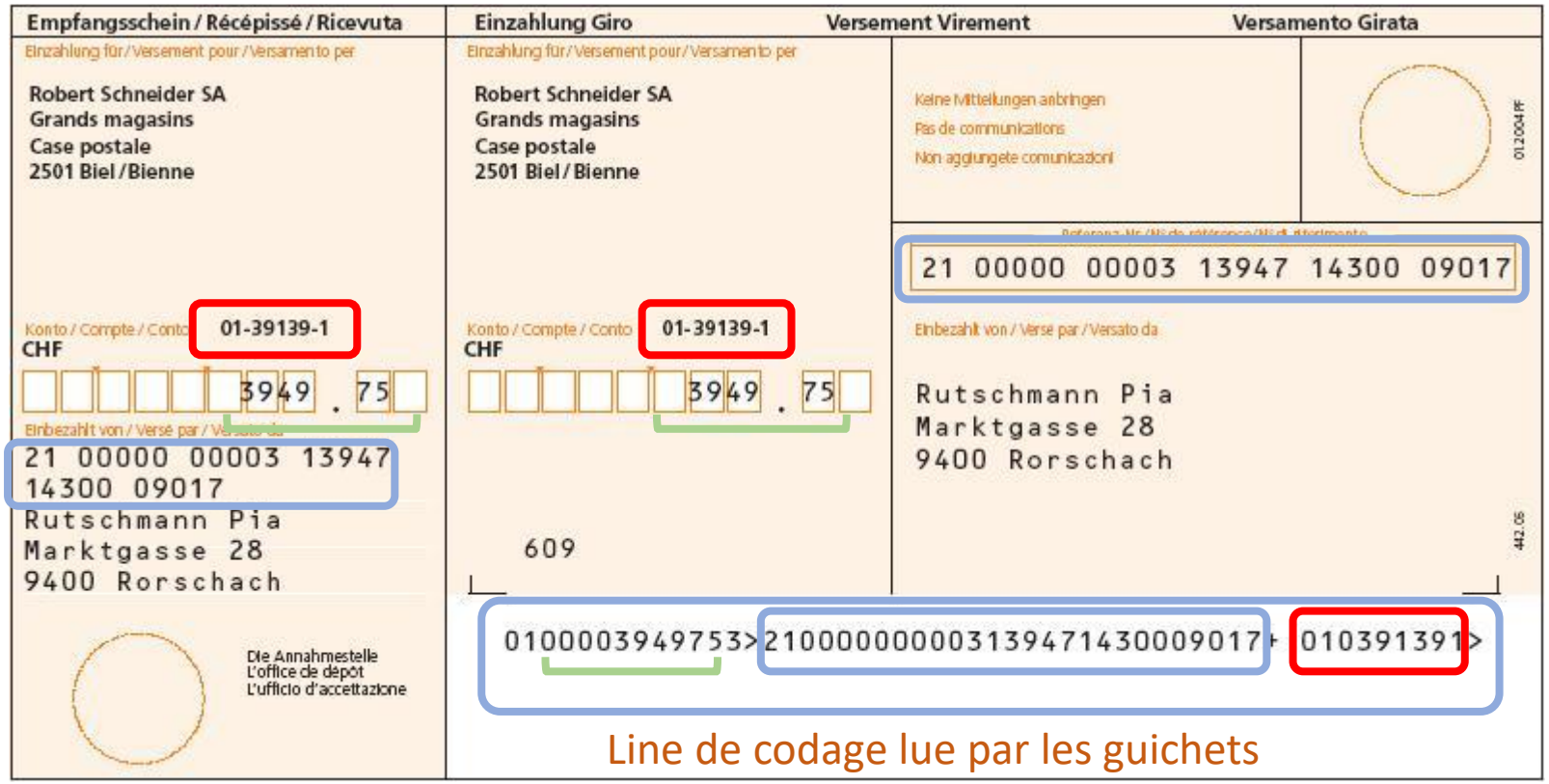
$Table =$

0	9	4	6	8	2	7	1	3	5

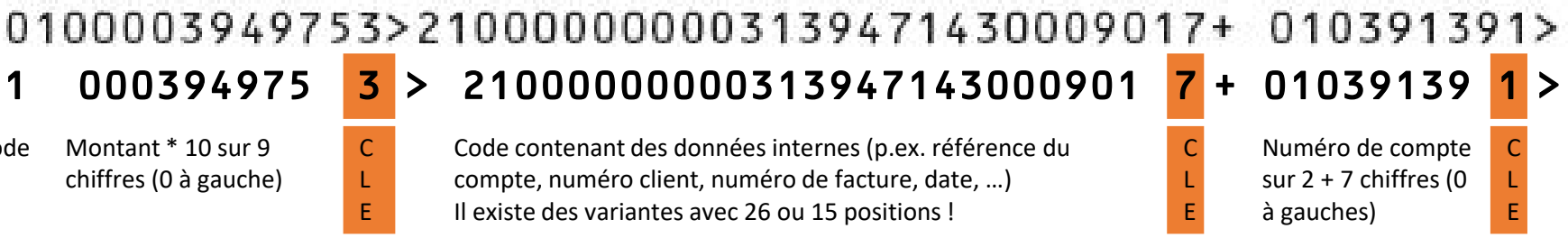
- Initialiser $r = 0, i = 0$

- Pour chaque chiffre x_i dans la séquence
  $r = Table[(x_i + r) \bmod 10]$
  $i = i + 1$
  Si $i > \#chiffres$: STOP : retourner $10 - r \ mod \ 10$.

Ce qu'on voit sur les BVR



Décomposition de la ligne de codage BVR

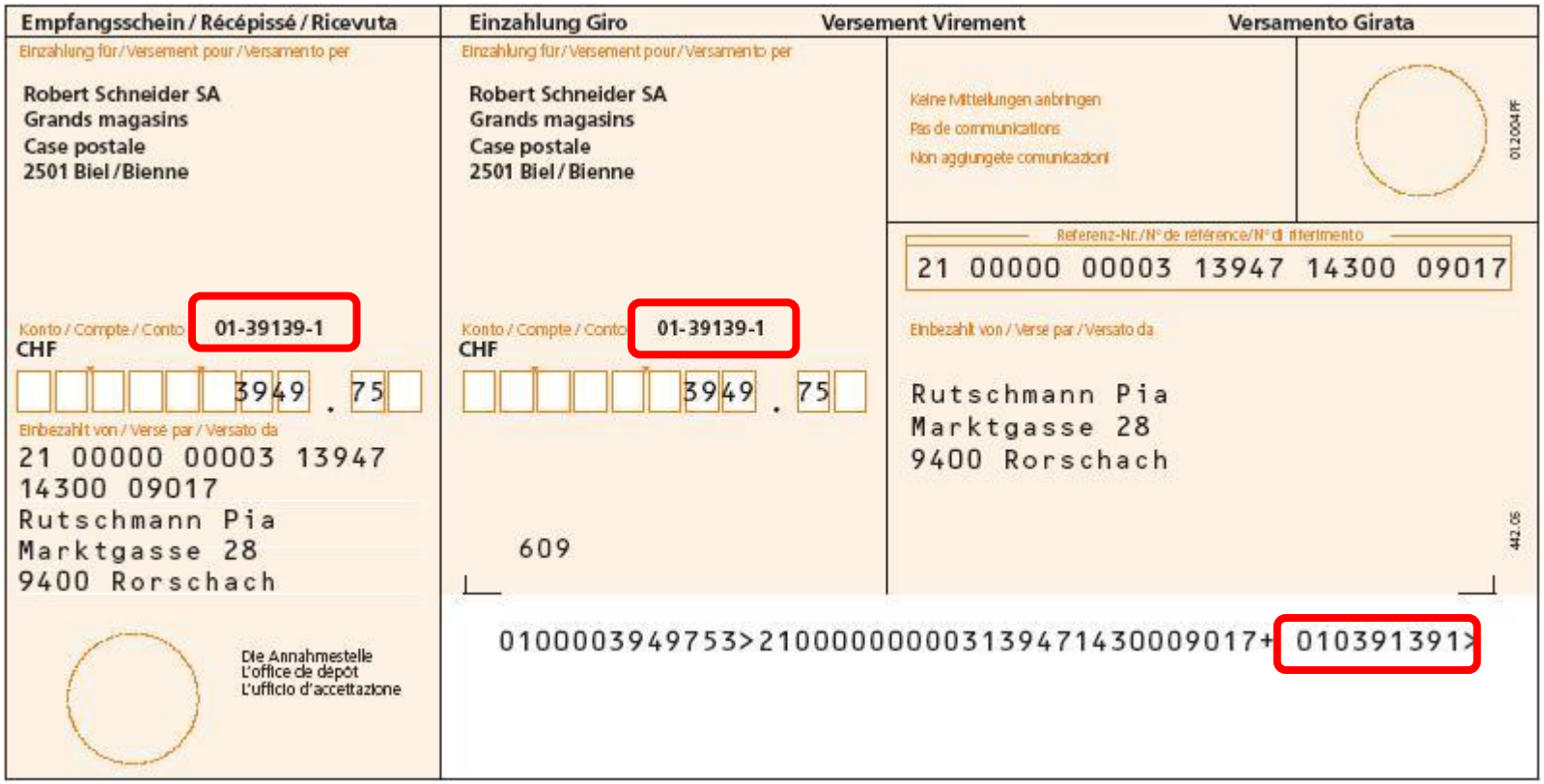


Il y a trois clés, toutes calculés avec l'algorithme modulo 10 récursif :

- La clé du montant (**3** – clé obtenue avec les 11 chiffres précédents),
- Clé du numéro de réf. (**7** – clé obtenue avec les 26 chiffres précédents),
- Clé du numéro de CCP (**1** – clé obtenue avec les 9 chiffres précédents).

Exemple : codage du CCP

Appliquons l'algorithme Modulo 10 récursif pour vérifier le numéro de compte 01–39139–1



Exemple : codage du CCP

- D'abord, notons que le CCP est codé sur 2 + 7 chiffres + le chiffre clé, or 01-39139-**1** est composé de 2 + 6 chiffres,
- Il manque un 0 : 01-039139-**1**,
- Il faut donc vérifier si la séquence 01039139 retourne bien **1** comme clé de chiffrement !

Clé du CCP 01-(0)39139-?

$T =$

0	9	4	6	8	2	7	1	3	5

i	r	x_i	$v = (x_i + r) \bmod 10$	$r = T[v]$	**Clé** $10 - r \bmod 10$
0	0	0	$(0 + 0) \bmod 10 = 0$	0	0
1	0	1	$(1 + 0) \bmod 10 = 1$	9	1
2	9	0	$(0 + 9) \bmod 10 = 9$	5	5
3	5	3	$(3 + 5) \bmod 10 = 8$	3	7
4	3	9	$(9 + 3) \bmod 10 = 2$	4	6
5	4	1	$(1 + 4) \bmod 10 = 5$	2	8
6	2	3	$(3 + 2) \bmod 10 = 5$	2	8
7	2	9	$(9 + 2) \bmod 10 = 1$	9	**1**

STOP : retourner **1**

Algorithme de chiffrement RSA

Par Ronald **R**ivest, Adi **S**hamir et Leonard **A**dleman (1977).

C'est une méthode de cryptage *ASYMÉTRIQUE*, contrairement au Nombre de César qui lui, est symétrique…

Quelle différence ?

Le nombre de César est basé sur une seule clé secrète (le nombre lui-même) qui, s'il est connu, permet de déchiffrer le message.

En tant que méthode asymétrique, RSA possède une clé privée ET une clé publique !

Principe asymétrique

1) Julie génère une clé publique $[n, e]$ et une clé privée $[d]$.

2) Paul écrit un message en clair (non-crypté) à Julie,

3) Le texte est converti en un nombre M (chaque caractère est remplacé par le code ASCII, Unicode, ….)
NOTE: il faut que $0 \leq M < n$, donc si $M \geq n$, on décomposera Men plusieurs nombres $M_i \in [0, n[$.

4) Paul récupère la clé publique de Julie, composée d'une paire (e, n) et calcule $\mu = M^e mod\ n$,

5) Julie reçoit $\mu = M^e mod\ n$ et calcule $\mu^d = M^{e \times d} = M\ mod\ n$ pour déchiffrer les messages.

Génération des clés

Julie génère une clé publique $[n, e]$ et une clé privée $[d]$.

- Choix de deux nombres premiers p et q (très grands!),

- Calcul de $n = p \times q$: n est publique,

Ex: $p = 5$ et $q = 11$, alors $n = 55$

- Calcul de $\varphi(n) = (p-1) \times (q-1)$: $\varphi(n)$est privé

Ex: $p = 5$ et $q = 11$, alors $\varphi(n) = 40$

- Choix d'un exposant e tel que $\mathrm{pgcd}\big(e, \varphi(n)\big) = 1$,

Ex: $e = 7$ (qui est premier avec $\varphi(n) = 40$).

Génération des clés [suite]

- Comme $\mathrm{pgcd}\big(e, \varphi(n)\big) = 1$, par l'algorithme d'Euclide étendu on obtient les coefficients de Bézout pour

$$d \times e + b \times \varphi(n) = 1$$

Ou autrement dit $d \times e = 1 \; mod \; \varphi(n)$!

Ex: $\varphi(n) = 40$ et $\mathrm{e} = 7$ => $3 \times 40 - 17 \times 7 = 1$
Donc $\mathrm{d} = -17 \; mod \; 40 = 23.$

NOTE: si $\mathrm{d} < 0$, on prend $\mathrm{d} = \mathrm{d} \bmod \varphi(n)$.

Dans ce cas, la clé publique est $[n, e] = [55, 7]$ et la clé privée est $d = 23.$

Chiffrement du message

Paul écrit un message qu'il convertit en nombre $M = 13$ puis applique la clé publique de Julie $[n, e] = [55, 7]$

Paul calcule alors $\mu = m^e \; mod \; n = 13^7 \; mod \; 55 = 7$ via ***l'algorithme d'exponentiation rapide*** :

$13^1 \; mod \; 55 = 13^1 \; mod \; 55 = 13$
$13^2 \; mod \; 55 = (13^1 \; mod \; 55) \times (13^1 \; mod \; 55) = 169 \; mod \; 55 = 4$
$13^4 \; mod \; 55 = 4 \times 4 \; mod \; 16 \; mod \; 55 = 16$
$13^7 \; mod \; 55 = (13^4 \times 13^2 \times 13^1) \; mod \; 55 = (16 \times 4 \times 13) \; mod \; 55$
$= 832 \; mod \; 55 = 7.$

Le message envoyé à Julie est donc $\mu = 7$.

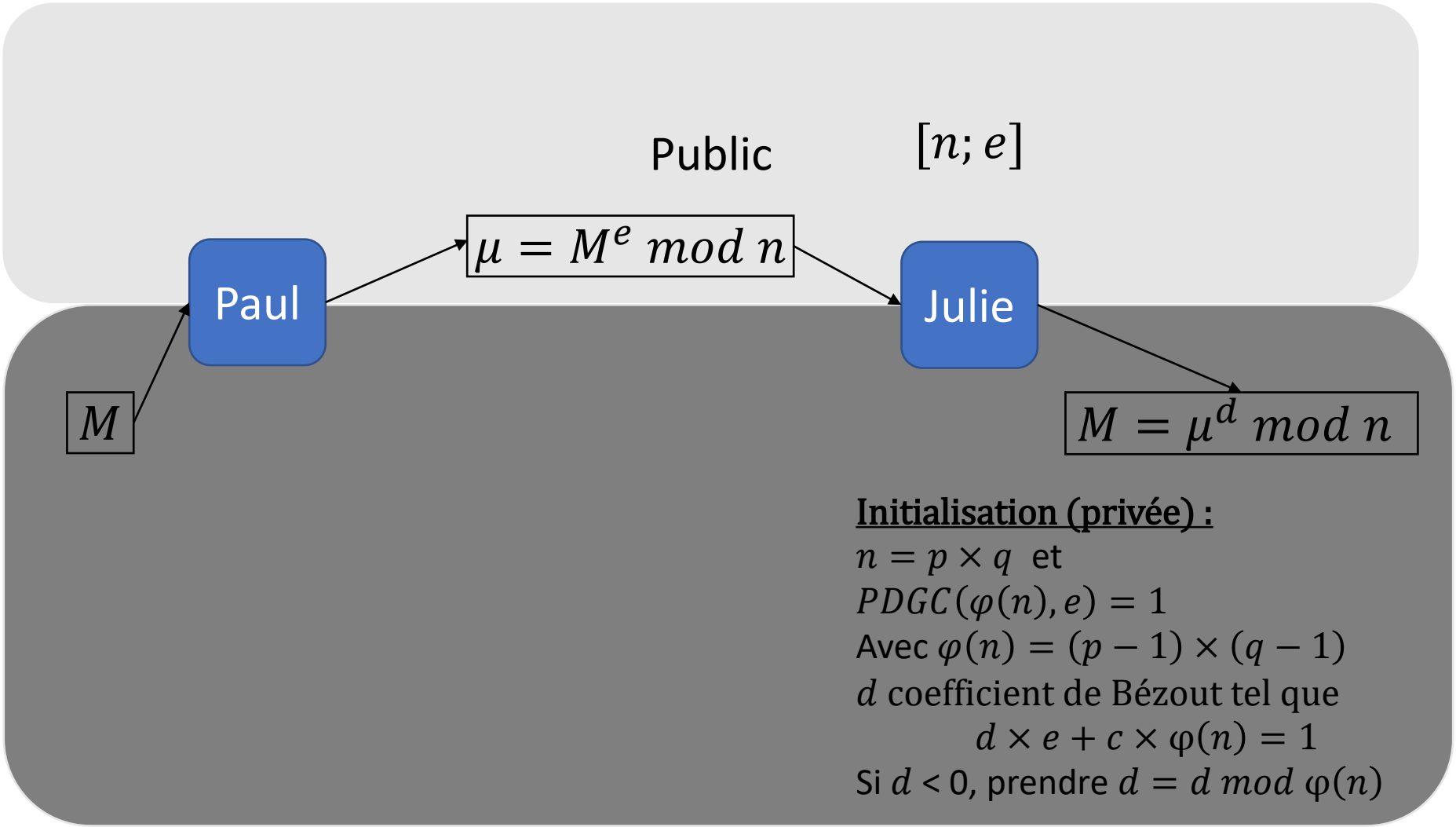
Déchiffrement du message

Julie reçoit le message $\mu = 7$. Elle va alors calculer (via l'algorithme d'exponentiation rapide)

$$M = \mu^d \, mod \, n$$

$7^1 \, mod \, 55 = 7$
$7^2 \, mod \, 55 = (49 \, mod \, 55) = 49$
$7^4 \, mod \, 55 = 49 \times 49 \, mod \, 55 = 2401 \, mod \, 55 = 36$
$7^8 \, mod \, 55 = 36 \times 36 \, mod \, 55 = 1296 \, mod \, 55 = 31$
$7^{16} \, mod \, 55 = 31 \times 31 \, mod \, 55 = 961 \, mod \, 55 = 26$
$7^{23} \, mod \, 55 = (7^{16} \times 7^4 \times 7^2 \times 7^1) mod \, (26 \times 36 \times 49 \times 7) \, mod \, 55$
$= (936 \, mod \, 55) \times (343 \, mod \, 55) = (1 \times 13) \, mod \, 55$
$= 13.$

RSA – vue d'ensemble



RSA - formalisation

Soit p, q deux premiers avec $p \neq q$ et

- $n = p \times q,$
- $\varphi(n) = (p-1) \times (q-1),$
- e tel que $PGCD\big(e, \varphi(n)\big) = 1,$
- d tel que $d \times e = 1\, mod\, \varphi(n)$

Alors pour tout $0 \leq M < n$ on a

Si $\boldsymbol{\mu = M^e\ mod\ n}$ alors $\boldsymbol{M = \mu^d\ mod\ n.}$

Pourquoi cela fonctionne

Rappelons-nous que $\mu = M^e \; mod \; n$, donc

$\mu^d = M^{e \times d} \; mod \; n.$

Par construction, nous savons que e est un coefficient de Bézout :

$$PGCD[e, \varphi(n)] = 1 = d \times e + b \times \varphi(n) \Rightarrow d \times e = 1 + b \times \varphi(n)$$

Donc

$$\mu^d \equiv_n M^{e \times d} = M^{1 + b \times \varphi(n)} \equiv_n M \times \left(M^{\varphi(n)}\right)^b.$$

Or, souvenez-vous du résultat disant que si M et n sont premiers entre eux, alors $M^{\varphi(n)} \equiv_n 1$. Or, $n = p \times q$ et $< n = p \times q$. Donc M et n sont premiers entre eux dès lors que M n'est pas multiple de p ou q.

Donc

$$1 \equiv_n M^{\varphi(n)} \equiv_n \left(M^{\varphi(n)}\right)^b.$$

Il suffit alors de multiplier des deux côtés par M pour obtenir

$$M \equiv_n M \times \left(M^{\varphi(n)}\right)^b \equiv_n \mu^d.$$

RSA – Complexité de décodage

En 1999, des chercheurs on décodé le RSA-155 (RSA avec nombre codé sur 155 chiffres décimaux, soit 512 bits).

Total : 8’000 ans de calculs à 1 Megaflops (1 million d’opérations par secondes).

Résolu en 3 mois de calcul avec 300 ordinateurs PC dédiés.

Aujourd’hui, les RSA-1024 et 2048 sont souvent utilisés. Les techniques brutes sont inefficaces, mais il est possible de cracker la clé grâce à des mesures de variations électriques sur un PC (nécessite un accès physique) !

RSA – Complexité de décodage

Supposons que p et q soient de l'ordre de 10^{100} (ce qui es le cas pour le RSA-1024).

Alors pour effectuer la décomposition en nombres premiers de $p \times q$ (d'ordre 10^{200}) il faut au pire des cas $\sqrt{p \times q} = \sqrt{10^{200}} = 10^{100}$ calculs.

Imaginons que nous disposons d'une puissance totale de 10^{30} flops (c'est une estimation grossière de la capacité de calcul totale de TOUS les ordinateurs sur terre combinés).

Il faudrait alors 10^{70} secondes pour résoudre la décomposition, soit plus de 10^{63} années de calcul !!!

Comment générer p et q ?

La génération est basée sur une approche probabiliste (donc les deux nombres premiers sont «probablement» premiers)

1. Générer un nombre aléatoire de la longueur désirée, disons n
2. Appliquer le Test de Miller-Rabin pour vérifier si a est premier,
   OUI => choisir n comme premier
   NON => prendre $n = n + 1$ et recommencer 2.

Test de Miller-Rabin

ATTENTION: il ne s'agit pas d'un test EXHAUSTIF, mais PROBABILISTE

Si le test échoue, on sait que le nombre n n'est PAS premier.
S'il réussit, on dira que n est «probablement» premier.

Test de Miller-Rabin

Entrées:

- n un entier impair > 3 (le nombre à tester)
- k un paramètre déterminant la précision du test (nombre de fois)

Sortie:

Faux si n est composé, Vrai si n est probablement premier

a est	Résultat de Miller-Rabin
Premier	Le résultat est toujours VRAI
Composé	Le résultat est FAUX avec probabilité $\geq 1 - 4^{-k}$

En sommes, Miller-Rabin peut avoir des faux POSITIFS (nombre probablement premier qui est, en réalité, composé), mais pas de faux NEGATIF !

Test de Miller-Rabin

Soit $n \geq 3$ un nombre entier impair et k un nombre de tests défini

Calculer s et d tels que

$$(n-1) = 2^s \times d$$

(n étant impair, $n-1$ est un multiple de 2)

Pour $t = 1, \ldots, k$ faire

choisir aléatoirement $x \in [2, n-2]$ et $y = x^d \ mod \ n$

Si $y \neq 1$ et $y \neq n-1$

poser $r = 1$

tant que $r \leq s-1$ faire

$y = y^2 \ mod \ n$

SI $y = n-1$ => passer à $t+1$

poser $r = r+1$

fin

SI $r = s$ et $y \neq 1$ => STOP, n n'est pas premier

SINON passer à $t+1$

fin

Fin => n est probablement premier

Test de Miller-Rabin - Exemple

Question: $n = 221$ est-il un nombre premier ?

- $n - 1 = 220 = 2 \times 110 = 2^2 \times 55$ $(s = 2, d = 55)$
- $x = 174 \in [2, 220]$ (pris aléatoirement) [Test $k = 1$]
  - $r = 1$:

    $y = x^d \ mod \ n = 174^{55} \ mod \ 221 = 47 \notin \{1,220\}$

    STOP: n n'est PAS premier

Test de Miller-Rabin - Explications

1. Par Fermat, si n est premier alors $a^{n-1} \equiv_n 1$ pour tout $a < n$,
2. Comme n est impair, on peut écrire $n - 1 = 2^s \times d$ (car $n - 1$ est pair),
3. Donc, en remplaçant on a que $\left(a^d\right)^{2^s} \equiv_n 1$ si n est premier,
4. En prenant la racine carrée ci-dessus, par distributivité du modulo, on aura que

$$\sqrt{(a^d)^{2^s}} = \pm\left(a^d\right)^{2^{s-1}},$$

et par distributivité du modulo, nous aurons donc que

$$\left(a^d\right)^{2^{s-1}} \equiv_n \pm 1.$$

Ce résultat s'applique pour toutes les racines successives, donc autrement dit, si n est premier, toutes les racines sont congruentes à ± 1. Or rappelons que $-1 \; mod \; n = n - 1$, donc on a forcément soit que $a^d \equiv_n \pm 1$ ou, si $a^d \; mod \; n \notin \{-1,1\}$, en mettant au carré successivement, nous tomberons forcément sur un $i \in [1, s-1]$ tel que $(a^d)^{2^i} \equiv_n \pm 1$. Si ce n'est pas le cas, il n'est pas possible que $a^{n-1} \equiv_n 1$, donc n ne peut pas être premier !

Le teste de Miller-Rabin cherche donc un a pour lequel nous n'obtenons pas $a^{n-1} \equiv_n 1$ de manière aléatoire.

Exercice RSA

Appliquez le chiffrement et déchiffrement du RSA avec

- $p = 5$ et $q = 7$
- choisissez $e = 5$ (vérifiez que c'est un choix valide)
- (d = 5)
- Envoyez le message $M = 10$ (donc $10^5 \ mod \ 35 = 5$)