Cryptologie & Sécurité

Master 1 informatique

Université Claude Bernard Lyon 1

Fabien Laguillaumie

fabien.laguillaumie@ens-lyon.fr
http://perso.ens-lyon.fr/fabien.laguillaumie

Cryptologie & Sécurité

Master 1 informatique

Université Claude Bernard Lyon 1

Fabien Laguillaumie

fabien.laguillaumie@ens-lyon.fr
http://perso.ens-lyon.fr/fabien.laguillaumie

Introduction

Cryptologie :

- Cryptographie :
  - conception de systèmes cryptographiques
  - étude (preuve) de leur sécurité
  - implantation efficace et amélioration des performances
- Cryptanalyse :
  - mise en défaut des systèmes cryptographiques
  - attaque des problèmes algorithmiques sous-jacents
  - observation des “canaux auxiliaires”

Introduction

Cryptologie :

- Cryptographie :
  - conception de systèmes cryptographiques
  - étude (preuve) de leur sécurité
  - implantation efficace et amélioration des performances $\longleftarrow$
- Cryptanalyse :
  - mise en défaut des systèmes cryptographiques
  - attaque des problèmes algorithmiques sous-jacents $\longleftarrow$
  - observation des “canaux auxiliaires”

Algorithmique de base

Un algorithme est une procédure prenant en entrée un ensemble de valeurs et qui donne en sortie un ensemble de valeurs après une suite finie d'instructions élémentaires.

La complexité d'un algorithme est mesurée par

- le nombres d'opérations qu'il réalise
- la quantité de mémoire dont il a besoin

en fonction de la *taille* de son entrée.

Croissance des fonctions

Trois “grands” types de fonction :

- $c \log(n)$ pour $c \in \mathbb{R}$
- cn^a pour $c \in \mathbb{R}$ et $a \in \mathbb{N}$
- $ce^n = c \exp(n)$ pour $c \in \mathbb{R}$

Croissance des fonctions

Trois “grands” types de fonction :

- $c \log(n)$ pour $c \in \mathbb{R}$ croissance lente
- cn^a pour $c \in \mathbb{R}$ et $a \in \mathbb{N}$ croissance rapide
- $ce^n = c \exp(n)$ pour $c \in \mathbb{R}$ croissance extrèmement rapide

Croissance des fonctions

n	10	100	1000
$\log(n)$	1	2	3
n^2	100	10000	10^6
n^{15}	10^{15}	10^{30}	10^{45}
$\exp(n)$	22026	2.7×10^{43}	1.97×10^{434}

Croissance des fonctions

$$f(n) = \Theta(g(n)) \iff \exists c_1, c_2, n_0 \in \mathbb{R} : 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$$
$$\text{pour } n \geq n_0$$

$$f(n) = O(g(n)) \iff \exists c, n_0 \in \mathbb{R} : 0 \leq f(n) \leq cg(n) \text{ pour } n \geq n_0$$

$$f(n) = \Omega(g(n)) \iff \exists c, n_0 \in \mathbb{R} : 0 \leq cg(n) \leq f(n) \text{ pour } n \geq n_0$$

$$f(n) = o(g(n)) \iff \forall\ c \in \mathbb{R}\ \exists n_0 \in \mathbb{N} : 0 \leq f(n) < cg(n) \text{ pour } n \geq n_0$$
$$\lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

Croissance des fonctions

$f(n) = O(g(n)) \iff \exists c, n_0 \in \mathbb{R} : 0 \leq f(n) \leq cg(n)$ pour $n \geq n_0$

Exemples :

- $4n^2 + n + 1 = O(n^2)$
- $10\log(n) + 5\log^3(n) + 7n + 3n^2 + 6n^3 = O(n^3)$

Introduction

Quelques ordres de grandeur

- sécurité : $\geq 2^{80}$
- nombre d'atomes dans l'univers : $10^{80} \sim 2^{265}$
- taille d'un module RSA : 1024 bits $\sim 2^{1024} \sim 10^{310}$
- taille d'une clé AES : 256 bits
- Core 2 Quad (Penryn) - 3,2 GHz : 2×24200 MIPS [1]
  1 000 000 $\sim 2^{20}$
  2^{35} opérations en 1 seconde
  Recherche exhaustive sur 2^{80} : $2^{80}/2^{35} = 2^{45}$ secondes

1. le nombre de millions d'instructions complétées par le microprocesseur en une seconde

Introduction

Quelques ordres de grandeur

- sécurité : $\geq 2^{80}$
- nombre d'atomes dans l'univers : $10^{80} \sim 2^{265}$
- taille d'un module RSA : 1024 bits $\sim 2^{1024} \sim 10^{310}$
- taille d'une clé AES : 256 bits
- Core 2 Quad (Penryn) - 3,2 GHz : 2×24200 MIPS [1]
  1 000 000 $\sim 2^{20}$
  2^{35} opérations en 1 seconde
  Recherche exhaustive sur 2^{80} : $2^{80}/2^{35} = 2^{45}$ secondes
  $\rightsquigarrow$ 1114925 années

http://fr.wikipedia.org/wiki/Ordre_de_grandeur_(nombres)

1. le nombre de millions d'instructions complétées par le microprocesseur en une seconde

Introduction

Quelques grandeurs

B. Schneier. Cryptographie appliquée.

Probabilité de mourir foudroyé (par jour)	1 chance sur 9 milliards (2^{33})
Probabilité de gagner le gros lot à la loterie américaine	1 chance sur 4 000 000 (2^{22})
Probabilité de gagner le gros lot à la loterie américaine et de mourir le même jour	1 chance sur 2^{61}
Probabilité d'être tué dans un accident automobile (aux États-Unis sur toute une vie)	1 chance sur 88 (2^{7})
Âge de la Terre	10^{9} années (2^{30})
Âge de l'Univers	10^{10} années (2^{34})
Nombre d'atomes constituant l'Univers	10^{77} (2^{265})

Un nombre de 1024 bit :

5858564308428828017644637396873011442410741924446401526444489392722401
2630691789482633773954538722685686046254635246206232275735158054157931
106062291505299908970881005023860120906954381649574977117336617312655
2467966227874466635888109429506335487371428797711478405925439695590447
76689821928151495754348605493

Groupe
Anneau
Corps
(finis)

Groupe

Un groupe $(\mathbb{G}, \cdot)$ est un ensemble muni d'une opération $\cdot$ satisfaisant :

1. associativité : $\forall\ a, b, c \in \mathbb{G} : \quad (a \cdot b) \cdot c = a \cdot (b \cdot c)$
2. élément neutre : $\exists 1 \in \mathbb{G}, \forall\ a \in \mathbb{G} : \quad a \cdot 1 = 1 \cdot a = a$
3. inverse : $\forall\ a \in \mathbb{G}, \exists a^{-1} \in \mathbb{G} : a \cdot a^{-1} = a^{-1} \cdot a = 1$

Anneau

Un anneau $(\mathbb{R}, +, \cdot)$ est un ensemble muni de deux opérations $+$ et $\cdot$ tel que :

1. $(\mathbb{R}, +)$ est un groupe commutatif
2. $(\mathbb{R}, \cdot)$ est un ensemble pour lequel $\cdot$ est associative

Corps

Un corps $(\mathbb{F}, +, \cdot)$ est un ensemble muni de deux opérations $+$ et $\cdot$ tel que :

1. $(\mathbb{F}, +)$ est un groupe commutatif d'élément neutre 0
2. $(\mathbb{F} \setminus \{0\}, \cdot)$ est un groupe commutatif d'élément neutre 1
3. $(\mathbb{F}, +, \cdot)$ satisfait une loi distributive :
   $\forall\, a, b, c \in \mathbb{F}, \quad a \cdot (b + c) = a \cdot b + a \cdot c$

Arithmétique

Arithmétique élémentaire

$\mathbb{N} = \{0, 1, 2, 3, \ldots, \}$
$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$

- $a, b \in \mathbb{Z}$:
$$a \mid b \Longleftrightarrow \exists\ c \in \mathbb{Z}\ : b = ac$$
- Quelques propriétés :
  1. $a \mid a$
  2. si $a \mid b$ et $b \mid c$ alors

Arithmétique élémentaire

$\mathbb{N} = \{0, 1, 2, 3, \ldots, \}$
$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$

- $a, b \in \mathbb{Z}$:
$$a \mid b \Longleftrightarrow \exists\ c \in \mathbb{Z}\ : b = ac$$
- Quelques propriétés :
  1. $a \mid a$
  2. si $a \mid b$ et $b \mid c$ alors $a \mid c$
  3. si $a \mid b$ et $a \mid c$ alors

Arithmétique élémentaire

$\mathbb{N} = \{0, 1, 2, 3, \ldots, \}$
$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$

- $a, b \in \mathbb{Z}$:
$$a \mid b \Longleftrightarrow \exists\ c \in \mathbb{Z}\ : b = ac$$
- Quelques propriétés :
  1. $a \mid a$
  2. si $a \mid b$ et $b \mid c$ alors $a \mid c$
  3. si $a \mid b$ et $a \mid c$ alors $a \mid (bx + cy)\ \forall\ x, y \in \mathbb{Z}$
  4. si $a \mid b$ et $b \mid a$ alors

Arithmétique élémentaire

$\mathbb{N} = \{0, 1, 2, 3, \ldots, \}$
$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$

- $a, b \in \mathbb{Z}$:

$$a \mid b \Longleftrightarrow \exists\ c \in \mathbb{Z}\ : b = ac$$

- Quelques propriétés :
  1. $a \mid a$
  2. si $a \mid b$ et $b \mid c$ alors $a \mid c$
  3. si $a \mid b$ et $a \mid c$ alors $a \mid (bx + cy)\ \forall\ x, y \in \mathbb{Z}$
  4. si $a \mid b$ et $b \mid a$ alors $a = \pm b$

- Division euclidienne : $a, b \in \mathbb{Z},\ b \geq 1$

$$\exists!(q, r) \in \mathbb{N}^2 : a = bq + r \text{ avec } 0 \leq r < b$$

Arithmétique élémentaire

Définition (pgcd)

Le pgcd de a et b est le plus grand entier qui divise à la fois a et b.

- $d = \mathrm{pgcd}(a, b)$:
  $d \mid a$, $d \mid b$ et si $c \mid a$ et $c \mid b$ alors

Définition

a et b sont premiers entre eux si $pgcd(a, b) = 1$

Définition (Nombre premier)

Un entier $p \geq 2$ *est premier si ses seuls diviseurs positifs sont* 1 *et* p.

- $p \mid ab \Longrightarrow p \mid a$ ou $p \mid b$
- il y a une infinité de nombres premiers
- y en a-t-il beaucoup ?

Arithmétique élémentaire

Définition (pgcd)

Le pgcd de a et b est le plus grand entier qui divise à la fois a et b.

- $d = \mathrm{pgcd}(a, b)$:
  $d \mid a$, $d \mid b$ et si $c \mid a$ et $c \mid b$ alors $c \mid d$.

Définition

a et b sont premiers entre eux si pgcd$(a, b) = 1$

Définition (Nombre premier)

Un entier $p \geq 2$ est premier si ses seuls diviseurs positifs sont 1 et p.

- $p \mid ab \implies p \mid a$ ou $p \mid b$
- il y a une infinité de nombres premiers
- y en a-t-il beaucoup ?
  $\pi(x) = \#\{p \text{ premier } \leq x\} \sim_{x \to \infty} x/\ln(x)$

Arithmétique élémentaire

Théorème (Théorème fondamental de l'arithmétique)

Tout entier $n \geq 2$ admet une factorisation unique (à l'ordre près des facteurs) en produit de puissances de nombres premiers :

$$n = p_1^{e_1} p_2^{e_2} \dots p_\ell^{e_\ell}$$

avec les p_i distincts et $e_i > 0$ entiers.

Arithmétique élémentaire

Théorème (Théorème fondamental de l'arithmétique)

Tout entier $n \geq 2$ admet une factorisation unique (à l'ordre près des facteurs) en produit de puissances de nombres premiers :

$$n = p_1^{e_1} p_2^{e_2} \dots p_\ell^{e_\ell}$$

avec les p_i distincts et $e_i > 0$ entiers.

(Preuve non-constructive...)

$\mathbb{Z}/n\mathbb{Z}$

$$\mathbb{Z}/n\mathbb{Z} = \{0, 1, 2, \ldots, n-1\}$$

avec $+$ et $\times$ " mod n"

$\mathbb{Z}/4\mathbb{Z}$:

$\times$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

$\mathbb{Z}/5\mathbb{Z}$:

$\times$	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

$\mathbb{Z}/n\mathbb{Z}$

- $(\mathbb{Z}/n\mathbb{Z}, +)$ est un groupe

$\mathbb{Z}/n\mathbb{Z}$

- $(\mathbb{Z}/n\mathbb{Z}, +)$ est un groupe
- $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ est un anneau

$\mathbb{Z}/n\mathbb{Z}$

- $(\mathbb{Z}/n\mathbb{Z}, +)$ est un groupe
- $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ est un anneau

- $(\mathbb{Z}/p\mathbb{Z}, +, \times)$ est un *corps* si et seulement si p est premier

$\mathbb{Z}/n\mathbb{Z}$

- $(\mathbb{Z}/n\mathbb{Z}, +)$ est un groupe
- $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ est un anneau

- $(\mathbb{Z}/p\mathbb{Z}, +, \times)$ est un *corps* si et seulement si p est premier

- $(\mathbb{Z}/n\mathbb{Z})^\star = \{x \in \mathbb{Z}/n\mathbb{Z} : \exists x^{-1} \text{ tel que } xx^{-1} \equiv 1 \pmod{n}\}$

$\mathbb{Z}/n\mathbb{Z}$

- $(\mathbb{Z}/n\mathbb{Z}, +)$ est un groupe
- $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ est un anneau

- $(\mathbb{Z}/p\mathbb{Z}, +, \times)$ est un *corps* si et seulement si p est premier

- $(\mathbb{Z}/n\mathbb{Z})^\star = \{x \in \mathbb{Z}/n\mathbb{Z} : \exists x^{-1} \text{ tel que } xx^{-1} \equiv 1 \pmod{n}\}$

  En particulier : $(\mathbb{Z}/p\mathbb{Z})^\star = \mathbb{Z}/p\mathbb{Z} \setminus \{0\}$.

  $(\mathbb{Z}/n\mathbb{Z})^\star$ est l'ensemble des éléments de $\mathbb{Z}/n\mathbb{Z}$ *premiers* à n (Bézout)

$\mathbb{Z}/n\mathbb{Z}$

- $(\mathbb{Z}/n\mathbb{Z}, +)$ est un groupe
- $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ est un anneau

- $(\mathbb{Z}/p\mathbb{Z}, +, \times)$ est un *corps* si et seulement si p est premier

- $(\mathbb{Z}/n\mathbb{Z})^\star = \{x \in \mathbb{Z}/n\mathbb{Z} : \exists x^{-1} \text{ tel que } xx^{-1} \equiv 1 \pmod{n}\}$

  En particulier : $(\mathbb{Z}/p\mathbb{Z})^\star = \mathbb{Z}/p\mathbb{Z} \setminus \{0\}$.

  $(\mathbb{Z}/n\mathbb{Z})^\star$ est l'ensemble des éléments de $\mathbb{Z}/n\mathbb{Z}$ *premiers* à n (Bézout)

- $((\mathbb{Z}/n\mathbb{Z})^\star, \times)$ est un groupe

Soit $n \geq 1$.

- $\varphi(n)$ représente le nombre d'entiers entre 1 et n qui sont premiers à n.

$$\varphi(n) = \#\{x \in [\![1, n]\!] : \mathrm{pgcd}(x, n) = 1\}$$

$$\#(\mathbb{Z}/n\mathbb{Z})^\star = \varphi(n)$$

- Propriétés :
  - p premier $\Longrightarrow \varphi(p) =$

Soit $n \geq 1$.

- $\varphi(n)$ représente le nombre d'entiers entre 1 et n qui sont premiers à n.

$$\varphi(n) = \#\{x \in [\![1, n]\!] : \mathrm{pgcd}(x, n) = 1\}$$

$$\#(\mathbb{Z}/n\mathbb{Z})^\star = \varphi(n)$$

- Propriétés :
  - p premier $\Longrightarrow \varphi(p) = p - 1$

Soit $n \geq 1$.

- $\varphi(n)$ représente le nombre d'entiers entre 1 et n qui sont premiers à n.

$$\varphi(n) = \#\{x \in [\![1, n]\!] : \text{pgcd}(x, n) = 1\}$$

$$\#(\mathbb{Z}/n\mathbb{Z})^\star = \varphi(n)$$

- Propriétés :
  - p premier $\Longrightarrow \varphi(p) = p - 1$
  - $\varphi(p^r) = (p-1)p^{r-1}$

Soit $n \geq 1$.

- $\varphi(n)$ représente le nombre d'entiers entre 1 et n qui sont premiers à n.

$$\varphi(n) = \#\{x \in [\![1, n]\!] : \mathrm{pgcd}(x, n) = 1\}$$

$$\#(\mathbb{Z}/n\mathbb{Z})^\star = \varphi(n)$$

- Propriétés :
  - p premier $\Longrightarrow \varphi(p) = p - 1$
  - $\varphi(p^r) = (p-1)p^{r-1}$
  - φ est multiplicative : si $\mathrm{pgcd}(m, n) = 1$

$$\varphi(mn) = \varphi(m)\varphi(n)$$

  - $n = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$

$$\varphi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\dots\left(1 - \frac{1}{p_k}\right)$$

Définition

Soit $(\mathbb{G}, \times)$ *un groupe d'ordre* n *(*$= \#\mathbb{G}$*). L'ordre d'un élément* $g \in \mathbb{G}$ *est le plus petit entier* r *tel que* $g^r = 1$.

Définition

Soit $(\mathbb{G}, \times)$ *un groupe d'ordre n* $(= \#\mathbb{G})$*. L'ordre d'un élément* $g \in \mathbb{G}$ *est le plus petit entier r tel que* $g^r = 1$*.*

- $\mathbb{Z}/21\mathbb{Z} = \{0, 1, 2, \ldots, 20\}$
- $\varphi(21) = \varphi(3 \times 7) = \varphi(3) \times \varphi(7) = 2 \times 6 = 12$
- $$(\mathbb{Z}/21\mathbb{Z})^\star = \{1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20\}$$

$(\mathbb{Z}/21\mathbb{Z})^\star$	$\{1$	2	4	5	8	10	11	13	16	17	19	$20\}$
ordre		6	3	6	2	6	6	2	3	6	6	2

$\mathbb{Z}/n\mathbb{Z}$

Exemples

$\mathbb{Z}/21\mathbb{Z}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
carré	1	4	9	16	4	15	7	1	18	16	16	18	1	7	15	4	16	9	4	1

Les carrés de $\mathbb{Z}/21\mathbb{Z} = \{1, 4, 7, 9, 15, 16, 18\}$

- $\mathbb{Z}/13\mathbb{Z}$ - $(\mathbb{Z}/13\mathbb{Z})^\star = \mathbb{Z}/13\mathbb{Z} \setminus \{0\}$ - $\varphi(13) = 12$
-

$(\mathbb{Z}/13\mathbb{Z})^\star =$	1	2	3	4	5	6	7	8	9	10	11	12
ordre		12	3	6	4	12	12	4	3	6	12	2
carré	1	4	9	3	12	10	10	12	3	9	4	1

- $\langle 6 \rangle = \{1, 6, 10, 8, 9, 2, 12, 7, 3, 5, 4, 11\}$

$$10 = 6^3 \pmod{13}$$

Soit $m_1, \dots, m_r$ des entiers premiers entre eux 2 à 2, et $a_1, \dots, a_r$ des entiers.

$$\begin{cases} x & = & a_1 \pmod{m_1} \\ x & = & a_2 \pmod{m_2} \\ & \vdots & \\ x & = & a_r \pmod{m_r} \end{cases}$$

Le théorème des restes chinois affirme qu'il existe une unique solution modulo $\prod_{i=1}^{r} m_i =: M$, donc que l'application π suivante est une bijection.

$$\begin{array}{rccc} \pi : & \mathbb{Z}/M\mathbb{Z} & \longrightarrow & \mathbb{Z}/m_1\mathbb{Z} \times \cdots \times \mathbb{Z}/m_r\mathbb{Z} \\ & x & \longmapsto & (x \mod m_1, \dots, x \mod m_r) \end{array}$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
-

$\pi(0) = (0,0)$	$\pi(5) =$	$\pi(10) =$
$\pi(1) =$	$\pi(6) =$	$\pi(11) =$
$\pi(2) =$	$\pi(7) =$	$\pi(12) =$
$\pi(3) =$	$\pi(8) =$	$\pi(13) =$
$\pi(4) =$	$\pi(9) =$	$\pi(14) =$

- En déduire la solution du système

$$\begin{cases} x & = & 2 \mod 3 \\ x & = & 3 \mod 5 \end{cases}$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
-

$\pi(0) = (0,0)$	$\pi(5) =$	$\pi(10) =$
$\pi(1) = (1,1)$	$\pi(6) =$	$\pi(11) =$
$\pi(2) =$	$\pi(7) =$	$\pi(12) =$
$\pi(3) =$	$\pi(8) =$	$\pi(13) =$
$\pi(4) =$	$\pi(9) =$	$\pi(14) =$

- En déduire la solution du système

$$\begin{cases} x & = & 2 \mod 3 \\ x & = & 3 \mod 5 \end{cases}$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
-
$$\begin{array}{|l|l|l|}
\pi(0) = (0,0) & \pi(5) = \qquad\qquad & \pi(10) = \qquad\qquad \\
\pi(1) = (1,1) & \pi(6) = & \pi(11) = \\
\pi(2) = (2,2) & \pi(7) = & \pi(12) = \\
\pi(3) = & \pi(8) = & \pi(13) = \\
\pi(4) = & \pi(9) = & \pi(14) =
\end{array}$$

- En déduire la solution du système

$$\left\{ \begin{array}{lcll} x & = & 2 & \mod 3 \\ x & = & 3 & \mod 5 \end{array} \right.$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
-

$\pi(0) = (0,0)$	$\pi(5) =$	$\pi(10) =$
$\pi(1) = (1,1)$	$\pi(6) =$	$\pi(11) =$
$\pi(2) = (2,2)$	$\pi(7) =$	$\pi(12) =$
$\pi(3) = (0,3)$	$\pi(8) =$	$\pi(13) =$
$\pi(4) =$	$\pi(9) =$	$\pi(14) =$

- En déduire la solution du système

$$\begin{cases} x &= 2 \mod 3 \\ x &= 3 \mod 5 \end{cases}$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
- $$\begin{array}{|l|l|l|} \pi(0) = (0,0) & \pi(5) = \qquad\qquad & \pi(10) = \qquad\qquad \\ \pi(1) = (1,1) & \pi(6) = & \pi(11) = \\ \pi(2) = (2,2) & \pi(7) = & \pi(12) = \\ \pi(3) = (0,3) & \pi(8) = & \pi(13) = \\ \pi(4) = (1,4) & \pi(9) = & \pi(14) = \end{array}$$
- En déduire la solution du système

$$\left\{ \begin{array}{lcll} x & = & 2 & \mod 3 \\ x & = & 3 & \mod 5 \end{array} \right.$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
-

$\pi(0) = (0,0)$	$\pi(5) = (2,0)$	$\pi(10) =$
$\pi(1) = (1,1)$	$\pi(6) =$	$\pi(11) =$
$\pi(2) = (2,2)$	$\pi(7) =$	$\pi(12) =$
$\pi(3) = (0,3)$	$\pi(8) =$	$\pi(13) =$
$\pi(4) = (1,4)$	$\pi(9) =$	$\pi(14) =$

- En déduire la solution du système

$$\begin{cases} x & = & 2 & \mod 3 \\ x & = & 3 & \mod 5 \end{cases}$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
- $$\begin{array}{|c|c|c|} \pi(0) = (0,0) & \pi(5) = (2,0) & \pi(10) = (1,0) \\ \pi(1) = (1,1) & \pi(6) = (0,1) & \pi(11) = (2,1) \\ \pi(2) = (2,2) & \pi(7) = (1,2) & \pi(12) = (0,2) \\ \pi(3) = (0,3) & \pi(8) = (2,3) & \pi(13) = (1,3) \\ \pi(4) = (1,4) & \pi(9) = (0,4) & \pi(14) = (2,4) \end{array}$$
- En déduire la solution du système

$$\left\{ \begin{array}{lcll} x & = & 2 & \mod 3 \\ x & = & 3 & \mod 5 \end{array} \right.$$

$\mathbb{Z}/n\mathbb{Z}$

Théorème des restes chinois

- $\mathbb{Z}/15\mathbb{Z} \longrightarrow \mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/5\mathbb{Z}$
-
$$\begin{array}{|c|c|c|}
\pi(0) = (0,0) & \pi(5) = (2,0) & \pi(10) = (1,0) \\
\pi(1) = (1,1) & \pi(6) = (0,1) & \pi(11) = (2,1) \\
\pi(2) = (2,2) & \pi(7) = (1,2) & \pi(12) = (0,2) \\
\pi(3) = (0,3) & \pi(8) = (2,3) & \pi(13) = (1,3) \\
\pi(4) = (1,4) & \pi(9) = (0,4) & \pi(14) = (2,4)
\end{array}$$

- En déduire la solution du système

$$\left\{\begin{array}{lcll} x & = & 2 & \mod 3 \\ x & = & 3 & \mod 5 \end{array}\right.$$

$$\begin{array}{rccl}\pi^{-1}: & \mathbb{Z}/m_1\mathbb{Z}\times\cdots\times\mathbb{Z}/m_r\mathbb{Z} & \longrightarrow & \mathbb{Z}/M\mathbb{Z} \\ & (a_1,\ldots,a_r) & \longmapsto & \sum_{i=1}^r a_i M_i y_i \pmod{M}\end{array}$$

avec

$$\begin{cases} M_i = M/m_i \\ y_i = M_i^{-1} \pmod{m_i}\end{cases}$$

- $\mathrm{CRT}((a_1,\ldots,a_r),(m_1,\ldots,m_r),M)$

```
result = 0
for i from 1 to r do{
    Mi=M/m[i]
    yi= ModInv(Mi,m[i])
    result = result + a[i]*Mi*yi mod M
}
return result
```

Le chiffrement RSA

Le chiffrement RSA

A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. R. Rivest, A. Shamir, L. Adleman. Communications of the ACM, Vol. 21 (2), pp. 120–126 (1978)

RSA

- **Génération des clés :**

  Soit $k \in \mathbb{N}$ le *paramètre de sécurité*

  - Construire 2 nombres premiers p et q tels que $2^{\lfloor k/2 \rfloor - 1} \leq p, q \leq 2^{\lfloor k/2 \rfloor} - 1$
  - $N = p \times q$
  - Choisir $e \in (\mathbb{Z}/\varphi(N)\mathbb{Z})^\star$ et calculer d tel que

$$ed \equiv 1 \pmod{\varphi(N)}.$$

clé publique	(N, e)
clé privée	(d, p, q)

RSA

- **Génération des clés :**

  Soit $k \in \mathbb{N}$ le *paramètre de sécurité*

  - Construire 2 nombres premiers p et q tels que $2^{\lfloor k/2 \rfloor - 1} \leq p, q \leq 2^{\lfloor k/2 \rfloor} - 1$ Primalité

  - $N = p \times q$ Multiplication

  - Choisir $e \in (\mathbb{Z}/\varphi(N)\mathbb{Z})^\star$ et calculer d tel que

    $$ed \equiv 1 \pmod{\varphi(N)}.$$

    Euclide étendu

clé publique	(N, e)
clé privée	(d, p, q)

RSA

- **Chiffrement :**

  Un message m est un élément de $\mathbb{Z}/N\mathbb{Z}$.

  - Alice obtient (N_B, e_B).

RSA

- **Chiffrement :**

  Un message m est un élément de $\mathbb{Z}/N\mathbb{Z}$.

  - Alice obtient (N_B, e_B).
  - $c = m^{e_B} \pmod{N_B}$.

RSA

- **Chiffrement :**

  Un message m est un élément de $\mathbb{Z}/N\mathbb{Z}$.

  - Alice obtient (N_B, e_B).
  - $c = m^{e_B} \pmod{N_B}$.

- **Déchiffrement :**

  - Bob utilise sa clé secrète (d_B, p_B, q_B).

RSA

- **Chiffrement :**

  Un message m est un élément de $\mathbb{Z}/N\mathbb{Z}$.

  - Alice obtient (N_B, e_B).
  - $c = m^{e_B} \pmod{N_B}$.

- **Déchiffrement :**

  - Bob utilise sa clé secrète (d_B, p_B, q_B).
  - $c^{d_B} \pmod{N_B} = m$.

RSA

- **Chiffrement :**

  Un message m est un élément de $\mathbb{Z}/N\mathbb{Z}$.

  - Alice obtient (N_B, e_B).
  - $c = m^{e_B} \pmod{N_B}$. Exponentiation Modulaire

- **Déchiffrement :**

  - Bob utilise sa clé secrète (d_B, p_B, q_B).
  - $c^{d_B} \pmod{N_B} = m$. Exponentiation Modulaire

RSA

Certificat X.509

```
Certificate:
   Data:
       Version: 1 (0x0)
       Serial Number: 7829 (0x1e95)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
       Validity
           Not Before: Jul  9 16:04:02 1998 GMT
           Not After : Jul  9 16:04:02 1999 GMT
       Subject: C=US, ST=Maryland, L=Pasadena, O=Brent Baccala,
                OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:b4:31:98:0a:c4:bc:62:c1:88:aa:dc:b0:c8:bb:
                   33:35:19:d5:0c:64:b9:3d:41:b2:96:fc:f3:31:e1:
                   66:36:d0:8e:56:12:44:ba:75:eb:e8:1c:9c:5b:66:
                   70:33:52:14:c9:ec:4f:91:51:70:39:de:53:85:17:
                   16:94:6e:ee:f4:d5:6f:d5:ca:b3:47:5e:1b:0c:7b:
                   c5:cc:2b:6b:c1:90:c3:16:31:0d:bf:7a:c7:47:77:
                   8f:a0:21:c7:4c:d0:16:65:00:c1:0f:d7:b8:80:e3:
                   d2:75:6b:c1:ea:9e:5c:5c:ea:7d:c1:a1:10:bc:b8:
                   e8:35:1c:9e:27:52:7e:41:8f
               Exponent: 65537 (0x10001)
```

RSA

Pourquoi ça marche ?

Théorème (Lagrange)

Si $(\mathbb{G}, \cdot)$ *est un groupe fini de cardinal* n*, alors* $a^n = 1$, $\forall\ a \in \mathbb{G}$

RSA

Pourquoi ça marche ?

Théorème (Lagrange)

Si $(\mathbb{G}, \cdot)$ *est un groupe fini de cardinal* n, *alors* $a^n = 1, \forall\ a \in \mathbb{G}$

Corollaire (Fermat)

p *un entier premier,* $a \in \mathbb{Z}$

$$pgcd(a, p) = 1 \Longrightarrow a^{p-1} = 1 \mod p$$

RSA

Pourquoi ça marche ?

Théorème (Lagrange)

Si $(\mathbb{G}, \cdot)$ *est un groupe fini de cardinal* n*, alors* $a^n = 1, \forall\ a \in \mathbb{G}$

Corollaire (Fermat)

p *un entier premier,* $a \in \mathbb{Z}$

$$pgcd(a, p) = 1 \Longrightarrow a^{p-1} = 1 \mod p$$

Corollaire (Euler)

$a, N \in \mathbb{Z}$

$$pgcd(a, N) = 1 \Longrightarrow a^{\varphi(N)} = 1 \mod N$$

RSA

- Pour déchiffrer $c \in \mathbb{Z}/N\mathbb{Z}$

$$m \equiv c^{d} \pmod{N}$$

En effet :

$$\begin{aligned} c^{d} \mod N &\equiv m^{ed} \pmod{N} \\ &\equiv m^{1+k\varphi(N)} \pmod{N} \\ &\equiv m \times (m^{\varphi(N)})^{k} \pmod{N} \\ &\equiv m \end{aligned}$$

$$ed \equiv 1 \pmod{\varphi(N)} \iff \exists k \in \mathbb{Z} : ed = 1 + k\varphi(N)$$

Arithmétique algorithmique

Arithmétique algorithmique

Les opérations de bases

- $\mathrm{Add}_2(a, b)$ $\quad (a = \sum_{i=0}^{n} a_i 2^i, b = \sum_{i=0}^{n} b_i 2^i)$

- $\mathrm{Add}_2(a, b)$ $\qquad (a = \sum_{i=0}^{n} a_i 2^i, b = \sum_{i=0}^{n} b_i 2^i)$

```
R[0] = 0
for i from 0 to n
    c[i] = a[i] + b[i] + R[i]
    if c[i] >= 2
        c[i] = c[i]-2
        R[i+1] = 1
c[n+1] = R[n+1]
return c = sum(i=0,n+1,c[i]2^i)
```

Complexité :

- $\text{Add}_2(a, b)$ $\quad (a = \sum_{i=0}^{n} a_i 2^i, b = \sum_{i=0}^{n} b_i 2^i)$

```
R[0] = 0
for i from 0 to n
    c[i] = a[i] + b[i] + R[i]
    if c[i] >= 2
        c[i] = c[i]-2
        R[i+1] = 1
c[n+1] = R[n+1]
return c = sum(i=0,n+1,c[i]2^i)
```

Complexité : $O(n)$

- $\text{Mul}_2(a, b)$ $\qquad (a = \sum_{i=0}^{n} a_i 2^i, b = \sum_{i=0}^{n} b_i 2^i)$

- $\mathrm{Mul}_2(a, b)$ $\quad (a = \sum_{i=0}^{n} a_i 2^i, b = \sum_{i=0}^{n} b_i 2^i)$

```
for i from 0 to n
    d[i] = a[i] * 2^i *  b

return c = sum(i=0,n+1,d[i])
```

Complexité :

Arithmétique algorithmique

Les opérations de bases

- $\mathrm{Mul}_2(a, b)$ $\quad$ $(a = \sum_{i=0}^{n} a_i 2^i, b = \sum_{i=0}^{n} b_i 2^i)$

```
for i from 0 to n
    d[i] = a[i] * 2^i *  b

return c = sum(i=0,n+1,d[i])
```

Complexité : $O(n^2)$

Arithmétique algorithmique

Les opérations de bases

- pgcd(a, b) $\qquad a \geq b > 0$

  On définit deux suites d'entiers $r_0, r_1, \ldots, r_{\ell+1}$ et $q_1, q_2, \ldots, q_\ell$

$$\begin{cases} r_0 & = & a \\ r_1 & = & b \end{cases}$$

$$\begin{array}{rcll} r_0 & = & r_1 q_1 + r_2 & (0 < r_2 < r_1) \\ & \vdots & & \\ r_{i-1} & = & r_i q_i + r_{i+1} & (0 < r_{i+1} < r_i) \\ & \vdots & & \\ r_{\ell-2} & = & r_{\ell-1} q_{\ell-1} + r_\ell & (0 < r_\ell < r_{\ell-1}) \\ r_{\ell-1} & = & r_\ell q_\ell & (r_{\ell+1} = 0) \end{array}$$

On a $r_\ell = \text{pgcd}(a, b)$.

- pgcd(a, b) $a \geq b > 0$

```
while b != 0
        t = b
        b = a mod b
        a = t
return a
```

Le nombre d'itérations de l'algorithme d'Euclide ℓ vérifie

$$\ell \leq \log b / \log \phi + 1 \tag{1}$$

où $\phi := (1+\sqrt{5})/2 \approx 1,62$ est le nombre d'or.

1. ϕ vérifie l'équation $\phi^2 - \phi - 1 = 0$
2. Pour $i = 2, \ldots, \ell - 1$,

$$r_{\ell-i} \geq r_{\ell-(i-1)} + r_{\ell-(i-2)}$$

3. Pour $i = 0, 1, \ldots, \ell - 1$

$$r_{\ell-i} \geq \phi^i.$$

4. En posant $i = \ell - 1$ dans la relation précédente, on a bien 1

- La complexité de la division euclidienne de x par y est $O\big(l(x) \times l(q_{x,y})\big)$ où $q_{x,y}$ est le quotient de x par y

- La complexité de l'algorithme d'Euclide est donc

$$O\left(\sum_{i=0}^{\ell-1}\Big(l(r_i) \times l(q_{i+1})\Big)\right)$$

  et finalement

- La complexité de la division euclidienne de x par y est $O\big(l(x) \times l(q_{x,y})\big)$ où $q_{x,y}$ est le quotient de x par y

- La complexité de l'algorithme d'Euclide est donc

$$O\left(\sum_{i=0}^{\ell-1}\Big(l(r_i) \times l(q_{i+1})\Big)\right)$$

et finalement

$$O(n^2)$$

Un chiffrement RSA :

5858564308428828017644637396873011442410741924446401526444489392722401263069178948263377395453872268568604625463524620623227573515805415793110606229150529990897088100502386012090695438164957497711733366173126552467966227874466635888109429506335487371428797711478405925439695590447766898219281514957543486 0493

$\widehat{\ }$

485026823229373001701980504783180351727171263592647327291705346026678385111020231892741216651887081869078636504254158873283040454382716615907535509571781887982061084704513239484344430179548060122539493726827253564903863276122674090816397373454613901866554200167072423792857741532013940404023739096690479 7167

mod

179769313486231590772930519078902473361797697894230657273430081157732675805500963132708477322407536021120113879871393357658789768814416622492849081450667704519456266638519269269623133293482336582955895481722648601861246662929361448136770863662699658175987684543951819223123953830177176642598667396016540539057

Arithmétique algorithmique

Exponentiation modulaire

- $2^{16} = 2 \times 2 \times 2 \times \cdots \times 2 = 655536$

  $\#$ multiplications :

Arithmétique algorithmique

Exponentiation modulaire

- $2^{16} = 2 \times 2 \times 2 \times \cdots \times 2 = 655536$

  $\#$ multiplications :15

Arithmétique algorithmique

Exponentiation modulaire

- $2^{16} = 2 \times 2 \times 2 \times \cdots \times 2 = 655536$

  $\#$ multiplications :15

- $2^2 = 4$
  $4^2 = 16$
  $16^2 = 256$
  $256^2 = 65536$

  $\#$ multiplications :

Arithmétique algorithmique

Exponentiation modulaire

- $2^{16} = 2 \times 2 \times 2 \times \cdots \times 2 = 655536$

  $\#$ multiplications :15

- $2^2 = 4$
  $4^2 = 16$
  $16^2 = 256$
  $256^2 = 65536$

  $\#$ multiplications : 4

Arithmétique algorithmique

Exponentiation modulaire

$m^{11} \pmod{N}$?

- Méthode naïve :

$$m \times m \times m \times m \times m \times m \times m \times m \times m \times m \times m$$

\# multiplications : 10

Arithmétique algorithmique

Exponentiation modulaire

$m^{11} \pmod N$?

- Méthode naïve :

$$m \times m \times m \times m \times m \times m \times m \times m \times m \times m \times m$$

  $\#$ multiplications : 10

- Méthode "square and multiply" :
  $11 =$

$m^{11} \pmod N$?

- Méthode naïve :

$$m \times m \times m \times m \times m \times m \times m \times m \times m \times m \times m$$

  $\#$ multiplications : 10

- Méthode “square and multiply” :
  $11 = 2 \times 5 + 1$

Arithmétique algorithmique

Exponentiation modulaire

$m^{11} \pmod N$?

- Méthode naïve :

$$m \times m \times m \times m \times m \times m \times m \times m \times m \times m \times m$$

  $\#$ multiplications : 10

- Méthode "square and multiply" :
  $11 = 2 \times (2 \times 2 + 1) + 1$

$m^{11} \pmod N$?

- Méthode naïve :

$$m \times m \times m \times m \times m \times m \times m \times m \times m \times m \times m$$

  $\#$ multiplications : 10

- Méthode "square and multiply" :
  $11 = 2 \times (2 \times 2 + 1) + 1$ $\qquad$ $11_2 = 1011$

$m^{11} \pmod{N}$?

- Méthode naïve :

$$m \times m \times m \times m \times m \times m \times m \times m \times m \times m \times m$$

  $\#$ multiplications : 10

- Méthode "square and multiply" :
  $11 = 2 \times (2 \times 2 + 1) + 1$ $\qquad 11_2 = 1011$

$$\begin{aligned} m^{11} &= m^{(2\times(2\times2+1)+1)} \\ &= (m^2)^{(2\times2+1)} \times m \\ &= (m^2)^{2\times2} \times m^2 \times m \\ &= (((m^2)^2)^2) \times m^2 \times m \end{aligned}$$

  $\#$ multiplications : 6

Arithmétique algorithmique

Exponentiation modulaire

- ExpModN(m, e, N)

```
x=m
for i from 1 to e-1
    x = x*m mod N
return x
```

Complexité :

- ExpModN(m, e, N)

```
x=m
for i from 1 to e-1
    x = x*m mod N
return x
```

Complexité : $O(e \times \log(N)^2)$

$$
\begin{aligned}
e &= e_0 + e_1 2 + e_2 2^2 + e_3 2^3 \cdots + e_{t-1} 2^{t-1} + 2^t \\
&= e_0 + 2(e_1 + e_2 2^+ e_3 2^2 \cdots + e_{t-1} 2^{t-2} + 2^{t-1}) \\
&= e_0 + 2\big(e_1 + 2(e_2 + e_3 2 + \ldots e_{t-1} 2^{t-3} + 2^{t-2})\big) \\
&\vdots \\
&= e_0 + 2\left(e_1 + 2\Big(e_2 + 2\big(e_3 + \ldots 2(e_{t-1} + 2)\big)\Big)\right)
\end{aligned}
$$

$$
\begin{aligned}
m^e &= m^{e_0 + 2\left(e_1 + 2\left(e_2 + 2\left(e_3 + \ldots 2(e_{t-1} + 2)\right)\right)\right)} \\
&= m^{e_0}\left(m^{e_1}\left(m^{e_2}\left(\ldots(m^{e_{t-1}}(m)^2)^2 \ldots\right)^2\right)^2\right)^2
\end{aligned}
$$

- ExpBinMod(m, e, N)

```
b=m
for i from t-1 to 0
      b = b² mod N
      if (e[i] == 1) then
               b=b*m mod N
return b
```

Complexité :

- ExpBinMod(m, e, N)

```
b=m
for i from t-1 to 0
      b = b² mod N
      if (e[i] == 1) then
               b=b*m mod N
return b
```

Complexité : $O(\log(e) \times \log(N)^2)$

Quel peut être le but de l'attaquant ?

- Retrouver la clé secrète — Cassage total

De quoi dispose un attaquant ?

- La clé publique

Introduction

Comment attaquer RSA ?

Quel peut être le but de l'attaquant ?

- Retrouver la clé secrète — Cassage total
- Retrouver un message à partir de son chiffré — Sens unique

De quoi dispose un attaquant ?

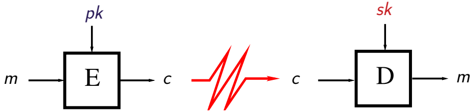
- La clé publique
- La clé publique et un chiffré

Introduction

Comment attaquer RSA ?

Remarque :

- Casser le sens unique $\leq$ casser totalement le système

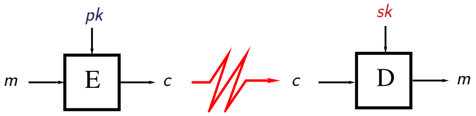


Remarque :

- Casser le sens unique $\leq$ casser totalement le système

Cassage total

Étant donné (N, e), retrouver (d, p, q)



Remarque :

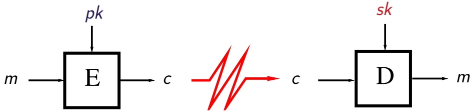
- Casser le sens unique $\leq$ casser totalement le système

Cassage total

Étant donné (N, e), retrouver (d, p, q)

Sens unique

Étant donné c et (N, e), retrouver m tel que $c = m^e \pmod N$



Introduction

Comment attaquer RSA ?

Remarque :

- Casser le sens unique $\leq$ casser totalement le système

Cassage total

Étant donné (N, e), retrouver (d, p, q)

Factorisation

Sens unique

Étant donné c et (N, e), retrouver m tel que $c = m^e \pmod{N}$

Problème RSA

Introduction

Factorisation

- $$f : \begin{array}{ccc} \mathcal{P}_k \times \mathcal{P}_k & \longrightarrow & \{N = p \times q, \text{ avec } p, q \in \mathcal{P}_k\} \\ (p, q) & \longmapsto & p \times q \end{array}$$

  f est une fonction à sens unique

- $$g : \begin{array}{ccc} \mathbb{Z}/N\mathbb{Z} & \longrightarrow & \mathbb{Z}/N\mathbb{Z} \\ x & \longmapsto & x^e \pmod N \end{array}$$

  g est une fonction à sens unique *à trappe*

Introduction

Factorisation

- $$f : \begin{array}{rcl} \mathcal{P}_k \times \mathcal{P}_k & \longrightarrow & \{N = p \times q, \text{ avec } p, q \in \mathcal{P}_k\} \\ (p, q) & \longmapsto & p \times q \end{array}$$

  f est une fonction à sens unique

- $$g : \begin{array}{rcl} \mathbb{Z}/N\mathbb{Z} & \longrightarrow & \mathbb{Z}/N\mathbb{Z} \\ x & \longmapsto & x^e \pmod N \end{array}$$

  g est une fonction à sens unique *à trappe*

On va maintenant s'intéresser à f.

Factorisation