# Crittografia a chiave pubblica - Un invito a RSA

Francesco Pappalardi

15 Novembre, 2001

# I due diversi tipi di Crittografia

- **Chiave pubblica.**
  - RSA;
  - Diffie–Hellmann;
  - Zainetti;
  - NTRU.
- **Chiave privata (o simmetrica).**
  - Lucifer;
  - DES;
  - AES.



$$RSA\text{–}2048 = 25195908475657893494027183240048398571429282126204$$
$$03202777713783604366202070759555626401852588078440691829064124 9$$
$$51508218929855914917618450280848912007284499268739280728777673 5$$
$$97141834727026189637501497182469116507761337985909570009733045 9$$
$$74880842840179742910064245869181719511874612151517265463228221 6$$
$$86998754918242243363725908514186546204357679842338718477444792 0$$
$$73993423658482382428119816381501067481045166037730605620161967 6$$
$$25613384414360383390441495263443219011465754445417842402092461 6$$
$$51572335077870774981712577246796292638635637328991215483143816 7$$
$$89988504044536402352738195137863656439121201039712282212072035 7$$

$RSA$–2048 è un numero con 617 cifre decimali

http://www.rsa.com/rsalabs/challenges/factoring/numbers.html/



$$RSA\text{–}2048 = p \cdot q, \quad p, q \approx 10^{308}$$

**PROBLEMA:** *Calcolare $p$ e $q$*

PREMIO: 200.000 \$ ($\sim$ 190.000€)!!

**Teorema.** Se $a \in \mathbb{N}$ $\exists!$ $p_1 < p_2 < \cdots < p_k$ *primi*

t.c. $a = p_1^{\alpha_1} \cdots p_k^{\alpha_k}$

**Purtroppo:** RSAlabs ritiene che per fattorizzare in un anno:

| numero | computers | memoria |
|---|---|---|
| $RSA$–1620 | $1.6 \times 10^{15}$ | 120 Tb |
| $RSA$–1024 | 342,000,000 | 170 Gb |
| $RSA$–760 | 215,000 | 4Gb. |



http://www.rsa.com/rsalabs/challenges/factoring/numbers.html

| Challenge Number | Prize ($US) | Status |
|---|---|---|
| RSA-576 | $10,000 | Not Factored |
| RSA-640 | $20,000 | Not Factored |
| RSA-704 | $30,000 | Not Factored |
| RSA-768 | $50,000 | Not Factored |
| RSA-896 | $75,000 | Not Factored |
| RSA-1024 | $100,000 | Not Factored |
| RSA-1536 | $150,000 | Not Factored |
| RSA-2048 | $200,000 | Not Factored |



## **Storia dell'** "*Arte del Fattorizzare*"

- 1880 Landry & Le Lasseur:

$$2^{2^6} + 1 = 274177 \times 67280421310721$$

- 1919 Pierre e Eugène Carissan (Macchina per fattorizzare)
- 1970 Morrison & Brillhart

$$2^{2^7} + 1 = 59649589127497217 \times 5704689200685129054721$$

- 1980 Crivello quadratico (QS) (Pomerance)



# Antica Macchina per fattorizzazione di Carissan

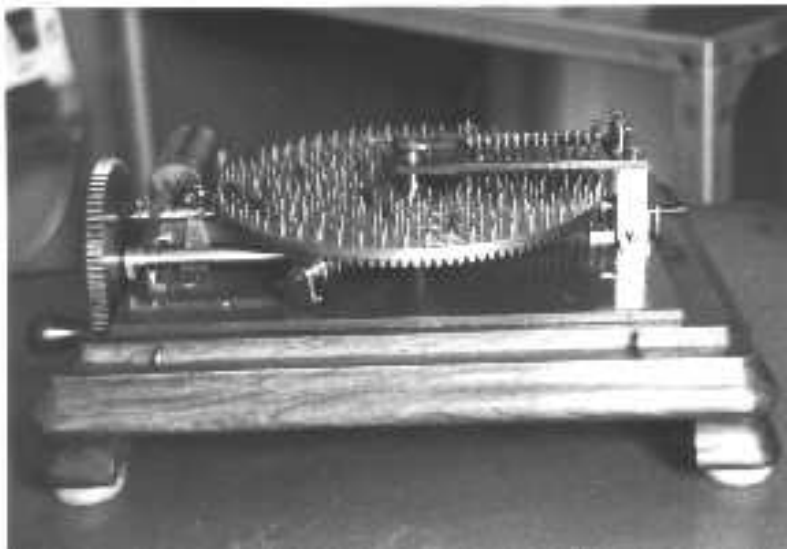


Figure 1: Conservatoire Nationale des Arts et Métiers in Paris.



Figure 2: Tenente Eugène Carissan.

$$225058681 = 229 \times 982789 \qquad 3 \text{ minuti}$$

$$3450315521 = 1409 \times 2418769 \qquad 2 \text{ minuti}$$

$$3570537526921 = 841249 \times 4244329 \qquad 18 \text{ minuti}$$



# Fattorizzare ai giorni nostri

1. Crivello Quadratico (QS): (8 mesi, 600 volontari, 20 paesi) D.Atkins, M. Graff, A. Lenstra, P. Leyland

$$RSA-129 = 114381625757888867669235779976146612010218296721242362562561842935706$$
$$935245733897830597123563958705058989075147599290026879543541 =$$
$$= 3490529510847650949147849619903898133417764638493387843990820577 \times$$
$$327691329932667095499619881908344614131776429679929425397982885 33$$

2. Crivello del campo numerico (NFS): (2 Feb 1999) 160 Sun workstations, 4 mesi.

$$RSA-155 = 109417386415705274218097073220403576120037329454492059909138421314763499842$$
$$889347847179972578912673324976257528997818337970765372440271467435315933543338 97 =$$
$$= 102639592829741105772054196573991675900716567808038066803341933521790711307779 \times$$
$$106603488380168454820927220360012878679207958575989291522270608237193062808643$$

3. Fattorizzazione con curve ellittiche: introdotta da H. Lenstra. Adatta a trovare fattori primi con 50 cifre (piccoli).

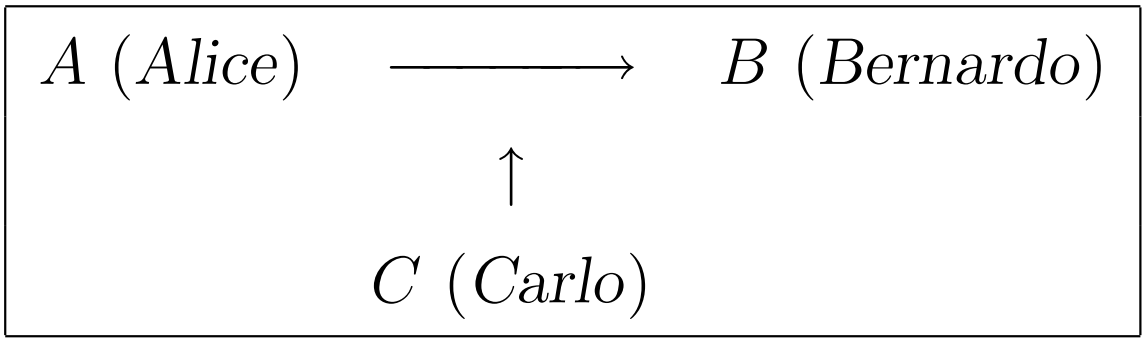
Hanno "*tempi di esecuzione sub-esponenziale*"



# Il crittosistema RSA

1978 R. L. Rivest, A. Shamir e L. Adleman
(Brevetto scaduto nel 1999)

**Problema:** *Alice vuole spedire il messaggio $\mathcal{P}$ a Bernardo e non vuole farlo leggere a Carlo.*




1. Generazione della chiave. Deve farla Bernardo.
2. Cifratura. Deve farla Alice.
3. Decifratura. Deve farla Bernardo.
4. Attacco al sistema. Vorrebbe farlo Carlo.



# Bernardo genera la chiave.

- Sceglie *in modo casuale* $p$ e $q$ primi $\quad (p, q \approx 10^{100})$;
- Calcola $\quad M = p \times q$, $\varphi(M) = (p-1) \times (q-1)$;
- Sceglie $e$ intero t.c.

$$0 \leq e \leq \varphi(M), \quad \text{e} \quad \gcd(e, \varphi(M)) = 1;$$

N.B. Si potrebbe anche prendere $e = 3$ e $p \equiv q \equiv 2 \bmod 3$.

Gli esperti suggeriscono $e = 2^{16} + 1$.

- Calcola l'inverso aritmetico $d$ di $e$ modulo $\varphi(M)$
  (i.e. $d \in \mathbb{N}$ (unico $\leq \varphi(M)$) t.c. $e \times d \equiv 1 \pmod{\varphi(M)}$;
- Pubblica $(M, e)$ **chiave pubblica** e conserva **chiave segreta** $d$.

**Problema:** *Come fa Bernardo a fare tutto ciò?*- Ci torneremo



# Alice cifra.

Si rappresenta il messaggio $\mathcal{P}$ come un elemento di $\mathbb{Z}/M\mathbb{Z}$.

(per esempio) $\boxed{A \leftrightarrow 1 \quad B \leftrightarrow 2 \quad C \leftrightarrow 3 \quad \ldots \quad Z \leftrightarrow 26 \quad AA \leftrightarrow 27 \ldots;}$

$$\texttt{PESCARA} \leftrightarrow 16 \cdot 26^6 + 5 \cdot 26^5 + 18 \cdot 26^4 + 3 \cdot 26^3 + 26^2 + 17 \cdot 26 + 1 = 5010338711.$$

N.B. È bene che i testi non siano troppo corti. Altrimenti si fa il *padding*.

$$\boxed{\mathcal{C} = E(\mathcal{P}) = \mathcal{P}^e \pmod{M}}$$

Esempio: $p = 9049465727$, $q = 8789181607$, $M = 79537397720925283289$,
$e = 2^{16} + 1 = 65537$, $\mathcal{P} = \texttt{PESCARA}$:

$$E(\texttt{PESCARA}) = 5010338711^{65537} \pmod{79537397720925283289}$$

$$= 93781898406377767 50 = \texttt{TYFWDKYEQFCGT}$$



## Bernardo Decifra.

$$\mathcal{P} = D(\mathcal{C}) = \mathcal{C}^d \pmod{M}$$

**N.B.** Bernardo decifra perchè è l'unico che conosce $d$.

**Il Piccolo Teorema di Fermat.** Se $a, m \in \mathbb{N}$, $\gcd(a, m) = 1$,

$$a^{\varphi(m)} \equiv 1 \pmod{m}.$$

Se $n_1 \equiv n_2 \bmod \varphi(m)$ allora $a^{n_1} \equiv a^{n_2} \bmod m$.

Quindi $(ed \equiv 1 \bmod \varphi(M))$

$$D(E(\mathcal{P})) = \mathcal{P}^{ed} \equiv \mathcal{P} \bmod M$$

Esempio(cont.):$d = 65537^{-1} \bmod \varphi(9049465727 \cdot 8789181607) = 57173914060643780153$

$D(\texttt{TYFWDKYEQFCGT}) =$

$93781898406377767 50^{57173914060643780153} (\mathrm{mod}\, 79537397720925283289) = \texttt{PESCARA}$



# L'algoritmo dei quadrati successivi

**Problema:** Come si fa a calcolare $a^b \bmod c$?

$$93781898406377776750^{571739140606437801 53} (\mathrm{mod}\, 79537397720925283289)$$

- Espansione binaria di $b = \sum_{j=0}^{[\log_2 b]} \epsilon_j 2^j$:

$$571739140606437801 53 = 1100011001011100101000101111101010111100110110001001000110001110 01;$$

- Calcolare ricorsivamente. $a^{2^j} \bmod c, j = 1, \ldots, [\log_2 b]$

$$a^{2^j} \bmod c = \left(a^{2^{j-1}} \bmod c\right)^2 \bmod c.$$

- Moltiplicare gli $a^{2^j} \bmod c$ con la $\epsilon_j = 1$;

$$a^b \bmod c = \left( \prod_{j=0, \epsilon_j = 1}^{[\log_2 b]} a^{2^j} \bmod c \right) \bmod c.$$



$$\#\{\textbf{oper. in } \mathbb{Z}/c\mathbb{Z} \textbf{ per calc. } a^b \bmod c\} \leq 2\log_2 b$$

`TYFWDKYEQFCGT` si decifra con 131 operazioni in

$$\mathbb{Z}/795373977209252832 89\mathbb{Z}.$$

AQS - Pseudo codice: $e_c(a,b) = a^b \bmod c$

| | | | | | |
|---|---|---|---|---|---|
| $e_c(a,b,c)$ | = | `if` | $b=1$ | `then` | $a \bmod c$ |
| | | `if` | $2 \mid b$ | `then` | $e_c(a, \frac{b}{2})^2 \bmod c$ |
| | | `else` | | | $a * e_c(a, \frac{b-1}{2})^2 \bmod c$ |

Per cifrare con $e = 2^{16}+1$ bastano 17 operazioni in $\mathbb{Z}/M\mathbb{Z}$.



# Generazione della chiave RSA

**Problema.** Produrre un primo in modo casuale $p \approx 10^{100}$.

| Algoritmo probabilistico (tipo Las Vegas). |
|---|
| 1. Let $p = \textsc{Random}(10^{100})$; |
| 2. If $\textsc{isprime}(p)$=1 then $\textsc{Output}$=$p$ else goto 1. |

**Sotto problemi:**

*A*. Quante iterazioni sono necessarie?

(i.e. come sono distribuiti i numeri primi?)

*B*. Come si verifica se $p$ è primo?

(i.e. come si calcola $\textsc{isprime}(p)$?) $\rightsquigarrow$ Test di primalità

*Falsa leggenda metropolitana: Verificare la primalità è equivalente a fattorizzare.*



# A. Distribuzione dei numeri primi:

$$\pi(x) = \#\{p \leq x \text{ t. c. } p \text{ è primo}\}.$$

**Teorema** (Hadamard - de la vallee Pussen - 1897)

$$\pi(x) \sim \frac{x}{\log x}.$$

Versione quantitativa:

**Teorema** (Rosser - Schoenfeld) se $x \geq 67$

$$\frac{x}{\log x - 1/2} < \pi(x) < \frac{x}{\log x - 3/2}.$$

Quindi

$$0.0043523959267 < Prob\ (\textsc{Random}(10^{100}) = \texttt{primo}\ < 0.004371422086$$



Se $P_k$ è la probabilità che tra $k$ numeri casuali $\leq 10^{100}$ ce ne sia uno primo, allora

$$P_k = 1 - \left(1 - \frac{\pi(10^{100})}{10^{100}}\right)^k$$

Quindi

$$0.663942 < P_{250} < 0.66554440$$



Per fare più in fretta so possono considerare solo numeri casuali dispari e non divisibili né per 3 né per 5 se:

$$\Psi(x, 30) = \#\{n \leq x \text{ t.c. } \gcd(n, 30) = 1\}$$

allora

$$\frac{4}{15}x - 4 < \Psi(x, 30) < \frac{4}{15}x + 4$$

Dunque se $P_k$ è la probabilità che tra $k$ numeri casuali $\leq 10^{100}$ coprimi con 30 ce ne sia uno primo, allora

$$P'_k = 1 - \left(1 - \frac{\pi(10^{100})}{\Psi(10^{100}, 30)}\right)^k$$

e

$$0.98365832 < P'_{250} < 0.98395199$$



# *B.* Test di primalità.

## Pseudo Primi e Pseudo Primi Forti.

**Teorema.** (Eulero) Se $p$ è primo, $p \nmid a \in \mathbb{N}$

$$a^{p-1} \equiv 1 \bmod p.$$

**Test di NON-primalità:**

$$M \in \mathbb{Z}, \quad 2^{M-1} \not\equiv 1 \bmod M \Rightarrow M \text{composto!}$$

Esempio: $2^{RSA\text{–}2024-1} \not\equiv 1 \bmod RSA\text{–}2024$.

Quindi $RSA$–2024 è composto.

Il Teorema di Eulero non si inverte. Infatti

$$2^{93960} \equiv 1 \pmod{93961} \quad \text{però} \quad 93961 = 7 \times 31 \times 433.$$



**Definizione.** $m \in \mathbb{N}$ dispari e composto si dice *pseudo primo* in base $a$ se

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

Se $m$ è pseudo primo rispetto a qualsiasi base $a$ allora di dice numero di *Carmichael*.

ESEMPIO $561 = 3 \times 11 \times 17$ è un numero di Carmichael.

**Teorema.** Alford, Granville & Pomerance (1995)

Esistono infiniti numeri di Carmichael

*Idea da buttar via?* NO.



# Numeri di Carmichael

Ecco tutti i 43 numeri di Carmichael fino a $10^6$:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 561 | 46657 | 252601 | 530881 | 1105 | 52633 | 278545 | 552721 |
| 1729 | 62745 | 294409 | 656601 | 2465 | 63973 | 314821 | 658801 |
| 2821 | 75361 | 334153 | 670033 | 6601 | 101101 | 340561 | 748657 |
| 8911 | 115921 | 399001 | 825265 | 10585 | 126217 | 410041 | 838201 |
| 15841 | 162401 | 449065 | 852841 | 29341 | 172081 | 488881 | 997633 |
| 41041 | 188461 | 512461 | | | | | |

**FATTI SUI NUMERI DI CARMICHAEL.**

1. $m$ di Carmichael $\Rightarrow$ $m$ privo di fattori quadratici (sfq);
2. $m$ dispari, sfq è di Carmichael $\Leftrightarrow \forall p|m, \quad p-1|m-1$;
3. $m$ è di Carmichael, $\Rightarrow$ $m$ è il prodotto di almeno tre primi;



# Pseudo primi forti

D'ora innanzi $m \equiv 3 \bmod 4$. (Solo per semplificare le notazioni)

**Definizione.** $m \in \mathbb{N}$, $m \equiv 3 \bmod 4$, composto si dice *pseudo primo forte* (PSPF) in base $a$ se

$$a^{(m-1)/2} \equiv \pm 1 \pmod{m}.$$

**N.B.** Se $p > 2$ primo $\Rightarrow$ $a^{(p-1)/2} \equiv \pm 1 \pmod{p}$.

Sia $\boxed{\mathcal{S} = \{a \in \mathbb{Z}/m\mathbb{Z} \text{ t.c. } \gcd(m,a) = 1, a^{(m-1)/2} \equiv \pm 1 \pmod{m}\}}$.

$\Rightarrow$

1. $\mathcal{S} \subseteq (\mathbb{Z}/m\mathbb{Z})^*$ sottogruppo;
2. Se $m$ è composto $\Rightarrow$ sottogruppo proprio;
3. Se $m$ è composto $\Rightarrow$ $\#\mathcal{S} \leq \frac{\varphi(m)}{4}$;
4. Se $m$ è composto $\Rightarrow$ $Prob(m\text{PSPF in base } a) \leq 0,25$.



# Test di Primalità di Miller–Rabin

Sia $m \equiv 3 \bmod 4$.

| ALGORITMO MILLER RABIN CON $k$ ITERAZIONI |
|---|
| $N = (m-1)/2$; |
| for $j = 0$ to $k$ do $a =$Random$(m)$; |
| if $a^N \not\equiv \pm 1 \bmod m$ then OUPUT=($m$ composto): END |
| endfor; OUTPUT=($m$ primo) |

Test probabilistico montecarlo:

$$Prob(\text{Miller Rabin dice } m \texttt{ primo e } m \text{ è composto}) \lessapprox \frac{1}{4^k}.$$

Nel mondo reale il software applica il test di Miller Rabin con $k = 10$.



# Perchè RSA è sicuro?

- È chiaro che se Corrado è capace a fattorizzare $M$, allora è in grado di calolare $\varphi(M)$ e poi anche $d$ e quindi a decifrare i messaggi;
- Per Corrado calcolare $\varphi(M)$ è equivalente a fattorizzare $M$. Infatti

$$p, q = \frac{M - \varphi(M) + 1 \pm \sqrt{(M - \varphi(M) + 1)^2 - 4M}}{2}.$$

- **Ipotesi RSA** L'unico modo per calcolare efficientemente

$$x^{1/e} \bmod M, \quad \forall x \in \mathbb{Z}/M\mathbb{Z}$$

(cioè decifrare i messaggi) è fattorizzare $M$.

In altre parole

*i due problemi sono polinomialmente equivalenti.*



# Certificazione di primalità

**Teorema.** Se $m$ è composto e vale **GRH**, allora $\exists a \leq 2\log^2 m$ t.c. $m$ non è pseudo primo forte in base $a$.

**Conseguenze:** Miller–Rabin si può "*de–randomizzare*".
(i.e. per mostrare che $m$ dispari è primo basta controllare che:

$$a^{(m-1)/2} \equiv \pm 1 \bmod m \quad \forall a \leq 2\log^2 m.$$

Altri metodi per certificare la primalità:

**Teorema [Pocklington].** $m$ dispari, $F|m-1$, $F > \sqrt{m}$, $a \in (\mathbb{Z}/m\mathbb{Z})^*$ t.c.

1. $a^{m-1} \equiv 1 \bmod m$;
2. $\forall q|F$, $q$ primo, $\gcd(a^{\frac{m-1}{q}} - 1, m) = 1$

Allora $m$ è **primo**.

**Esempio.** $2^6 \cdot 3^{33} \cdot 5^2 \cdot 7^{58} \cdot 11^{59} + 1$. è un primo (certificabile).