Il Criptosistema RSA

Crittografia a Chiave Pubblica

Prof. Fedeli Massimo

IIS Fermi Sacconi Cpia
Ascoli Piceno

3 gennaio 2026

Obiettivo della Crittografia

Escogitare metodi per cifrare e decifrare messaggi riservati, garantendo sicurezza da occhi indiscreti.

Esempio storico: Cifrario di Cesare

- Scambio lettere dell'alfabeto secondo una permutazione prefissata
- Esempio: spostamento di 3 posizioni ($A \to D$, $B \to E$, $Z \to C$)
- Metodo semplice ma facilmente violabile

Limiti della Crittografia Classica

Problemi del Cifrario di Cesare:

1. Chi sa cifrare sa anche decifrare (stessa difficoltà computazionale)
2. Necessario scambio preventivo della chiave segreta
3. Limitato a ordini di battaglia (nascondere dalle spie)
4. Interlocutori fidati (Labieno, Marco Antonio)

Esigenze della Crittografia Moderna

Oggi la rete permette comunicazioni tra chiunque: serve evitare che l'ampia diffusione del sistema ne pregiudichi la segretezza.

Principi della Crittografia a Chiave Pubblica

Requisiti fondamentali:

1. Ogni utente dispone di due chiavi:
   - **Chiave pubblica**: per cifrare messaggi destinati all'utente
   - **Chiave privata**: per decifrare (riservata al solo proprietario)
2. Cifrare è lecito a chiunque, ma solo il destinatario può decifrare
3. Le operazioni di cifrare e decifrare NON devono essere computazionalmente equivalenti

Crittografia a Chiave Pubblica

Sistema crittografico che utilizza coppie di chiavi asimmetriche per garantire sicurezza nelle comunicazioni.

Il Sistema RSA (1977)

Proposto da: Rivest, Shamir, Adleman nel 1977

Base matematica:

- Preliminari di Aritmetica
- Teorema di Eulero è il principio fondamentale
- Semplice da implementare

Caratteristiche

RSA è il metodo più noto nell'ambito della crittografia a chiave pubblica e si basa su proprietà dei numeri primi e dell'aritmetica modulare.

Rappresentazione dell'Alfabeto

Codifica lettere $\rightarrow$ numeri (0-26):

-	A	B	C	D	E	F	G	H	I	J	K	L
0	1	2	3	4	5	6	7	8	9	10	11	12

M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25	26

Esempio: "*CIAO*" diviene 3 9 1 15

(- indica lo spazio vuoto tra due parole)

La Funzione di Eulero

Definizione

Per ogni numero naturale $n \geq 1$, la funzione di Eulero $\varphi(n)$ conta quanti numeri interi tra 1 e n sono **coprimi** con n (cioè hanno MCD con n uguale a 1).

Esempi:

- $\varphi(6) = 2$ perché solo 1 e 5 sono coprimi con 6
- $\varphi(10) = 4$ perché 1, 3, 7, 9 sono coprimi con 10
- $\varphi(12) = 4$ perché 1, 5, 7, 11 sono coprimi con 12

Caso speciale: numeri primi

Se p è un numero primo, allora $\varphi(p) = p - 1$
perché tutti i numeri da 1 a $p - 1$ sono coprimi con p.

Proprietà della Funzione di Eulero

Proprietà fondamentali:

1. **Per numeri primi:** Se p è primo
$$\varphi(p) = p - 1$$
2. **Per potenze di primi:** Se p è primo e $k \geq 1$
$$\varphi(p^k) = p^k - p^{k-1} = p^{k-1}(p-1)$$
Non ci chiediamo perché :-)
3. **Funzione moltiplicativa:** Se $\gcd(m, n) = 1$
$$\varphi(m \cdot n) = \varphi(m) \cdot \varphi(n)$$

Applicazione importante per RSA: Se $q = p_1 \cdot p_2$ con p_1, p_2 primi distinti:

$$\varphi(q) = \varphi(p_1) \cdot \varphi(p_2) = (p_1 - 1)(p_2 - 1)$$

Il Teorema di Eulero

Teorema di Eulero

Se a e n sono interi coprimi (cioè $\gcd(a, n) = 1$), allora:

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

Esempi:

- $n = 10$: $\varphi(10) = 4$, quindi per $a = 3$ (coprimo con 10):

$$3^4 = 81 \equiv 1 \pmod{10}$$

- $n = 7$ (primo): $\varphi(7) = 6$, quindi per $a = 2$:

$$2^6 = 64 \equiv 1 \pmod{7}$$

Caso speciale: Piccolo Teorema di Fermat

Se p è primo e $\gcd(a, p) = 1$: $\quad a^{p-1} \equiv 1 \pmod{p}$

Come il Teorema di Eulero Rende Possibile RSA

L'idea chiave:
Se $s \cdot t \equiv 1 \pmod{\varphi(q)}$, allora esiste un intero k tale che:

$$s \cdot t = 1 + k \cdot \varphi(q)$$

Per un messaggio a coprimo con q:

$$\begin{aligned}(a^s)^t &= a^{s \cdot t} \\ &= a^{1+k \cdot \varphi(q)} \\ &= a \cdot (a^{\varphi(q)})^k \\ &\equiv a \cdot 1^k \pmod{q} \quad \text{(per il Teorema di Eulero)} \\ &\equiv a \pmod{q}\end{aligned}$$

Conclusione

Cifrando con s e decifrando con t, recuperiamo esattamente il messaggio originale a!

Cifrario di Cesare: Formulazione Matematica

Operazione di codifica (modulo 27):

- Lettere corrispondono a classi di congruenza modulo 27
- Cifrare con spostamento di 3 posizioni

Funzioni di Cifratura e Decifratura

Cifrare: $a \mapsto b \cdot a + c \pmod{27}$ per ogni a
Decifrare: $a \mapsto b^{-1} \cdot (a - c) \pmod{27}$
dove b^{-1} è l'inverso di b modulo 27

Nota: Molti analoghi procedimenti di codifica e decodifica sono identici su questa base: basta fissare due interi b, c modulo 27 con b invertibile modulo 27.

Funzionamento RSA: Generazione delle Chiavi

L'utente A costruisce le sue chiavi:

1. Sceglie un numero naturale q sufficientemente grande
2. Sceglie un numero primo con tutti i numeri a con $1 \leq a \leq 26$
3. Per il Teorema di Eulero, per ogni intero a compreso tra 1 e 26:

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

4. Sceglie due naturali s, t inversi modulo $\varphi(q)$:

$$s \cdot t \equiv 1 \pmod{\varphi(q)}$$

Quindi:

- q, s costituiscono la **chiave pubblica** di A
- t è la sua **chiave segreta**

Funzionamento RSA: Cifratura e Decifratura

B vuole cifrare una corrispondenza rivolta ad A:

1. B eleva ogni lettera a del messaggio alla s modulo q e trasmette ad A:

$$a^s \pmod{q}$$

2. A decifra elevando quanto ricevuto (cioè a^s) alla chiave segreta t modulo q e sfruttando:

$$(a^s)^t \equiv a \pmod{q}$$

Validità

Si noti che anche $a = 0$ soddisfa banalmente:

$$(a^s)^t \equiv a \pmod{q}$$

Esempio Pratico con RSA

Dati: $q = 101$ (primo), dunque $\varphi(101) = 100$

Si ha: $3 \cdot 67 \equiv 201 \equiv 1 \pmod{100}$

Scelta: $s = 3, t = 67$

A rende pubblici $q = 101$ e $s = 3$, mantiene segreto $t = 67$

B vuole scrivere "*CIAO*" (cioè 3 9 1 15) ad A:
Eleva i numeri alla 3 modulo 101:

$$3^3 \equiv 27 \pmod{101}$$
$$9^3 \equiv 47 \pmod{101}$$
$$1^3 \equiv 1 \pmod{101}$$
$$15^3 \equiv 42 \pmod{101}$$

Scrive: 27 47 1 42

Decifratura dell'Esempio

A decifra elevando alla 67:

$$\begin{aligned} 27^{67} &\equiv 3 \pmod{101} \\ 47^{67} &\equiv 9 \pmod{101} \\ 1^{67} &\equiv 1 \pmod{101} \\ 42^{67} &\equiv 15 \pmod{101} \end{aligned}$$

Recupera: 3 9 1 15

cioè "*CIAO*"

Nota

Il calcolo di potenze modulo q può essere svolto tramite metodi ragionevolmente rapidi, ma la sicurezza di RSA rispetto ad eventuali attacchi di un pirata C dipende dalla scelta di $q = 101$,

Sicurezza del Sistema RSA

Per garantire la sicurezza:

- Utilizzare primi $p_1 \neq p_2$ "**titanici**" (enormemente grandi)
- Porre $q = p_1 \cdot p_2$

Fattorizzazione

Il pirata C conosce q e s, ma deve comunque recuperare t per infrangere il sistema. Per questo ha verosimilmente bisogno di sapere $\varphi(q) = (p_1 - 1) \cdot (p_2 - 1)$.

Problema computazionale:

- Decomporre q nei suoi fattori primi p_1, p_2
- All'attuale stato della conoscenza, anche usando i migliori algoritmi ed i più potenti calcolatori, tempi proibitivamente lunghi
- Addirittura superiori a quanto trascorso dalla nascita dell'universo fino ad oggi!

La sicurezza di RSA è garantita proprio dalla difficoltà di decomporre q e recuperare i suoi fattori primi.

Conclusioni

Vantaggi del Sistema RSA

- Nessuno (se non A e chi conosce t) può facilmente decifrare
- q, s possono facilmente cifrare conoscendo solo la chiave pubblica
- Tutti possono facilmente cifrare
- La sicurezza si basa su un problema matematico complesso

Citazione (H. Lenstra, 1986)

"Supponiamo di avere due primi $p_1 \neq p_2$ di almeno 100 cifre. Supponiamo che p_1, p_2 finiscano in un pagliaio e che ci resti solo $q = p_1 \cdot p_2$. Deve essere avvertito come una sconfitta della scienza l'ammettere che il metodo più sensato che possiamo oggi seguire per trovare p_1 e p_2 è quello di cercare nel pagliaio".