7 Il Crittosistema RSA

7.1 Elementi di Teoria dei Numeri

Prima di definire il Crittosistema RSA, richiamiamo alcuni fatti elementari di Teoria dei Numeri.

Teorema 7.1. ***(Teorema Cinese dei Resti)***
Siano a_i, m_i, $i = 1, \dots, r$, interi con $gcd(m_i, m_j) = 1$ per ogni $i, j = 1, \dots, r$, $i \neq j$, e si consideri il seguente sistema congruenziale:

$$\mathcal{S}: \begin{cases} x \equiv a_1 \ mod \ m_1 \\ \vdots \\ x \equiv a_r \ mod \ m_r. \end{cases}$$

Allora risulta che:
(1) *il sistema $\mathcal{S}$ ammette soluzione;*
(2) *se x_1 e x_2 sono due soluzioni di $\mathcal{S}$, allora $x_1 \equiv x_2 \ mod \ M$, dove $M = \prod_{i=1}^{r} m_i$.*

Dimostrazione.
(1) Sia $M_i := \frac{M}{m_i}$ per ogni $i = 1, \dots, r$, allora si ha che $gcd(M_i, m_i) = 1$, poiché $gcd(m_i, m_j) = 1$ per ogni $i, j = 1, \dots, r$, $i \neq j$. Quindi esiste N_i tale che $M_i N_i \equiv 1 \ mod \ m_i$. Sia

$$x_0 := \sum_{h=1}^{r} a_h M_h N_h.$$

Risulta

$$m_i \mid M_h \text{ per } h \neq i \Rightarrow m_i \mid \sum_{h=1, h\neq i}^{r} a_h M_h N_h,$$

e quindi

$$x_0 \equiv a_i M_i N_i \ mod \ m_i.$$

Siccome $M_i N_i \equiv 1 \ mod \ m_i$, si ha $x_0 \equiv a_i \ mod \ m_i$ per ogni $i = 1, \dots, r$.
(2) Osserviamo che se x_1 e x_2 sono due soluzioni di $\mathcal{S}$, allora $x_1 \equiv x_2 \ mod \ m_i$ per ogni $i = 1, \dots, r$, e quindi, poiché $gcd(m_i, m_j) = 1$ per $i \neq j$, segue che $x_1 \equiv x_2 \ mod \ M$.

□

Corollario 7.2. *Il sistema $\mathcal{S}$ ammette un'unica soluzione compresa tra 0 ed $M - 1$.*

Dimostrazione. Dal **Teorema Cinese dei Resti**, le soluzioni di $\mathcal{S}$ sono tutti e soli gli elementi della classe di resto modulo M individuata da x_0. Poiché tale classe ha un unico rappresentante compreso tra 0 ed $M - 1$, segue l'asserto.

□

Definizione 7.3. Sia n un intero, allora

$$\varphi(n) = |\{1 \leq x \leq n : \gcd(x,n) = 1\}|$$

è la **"Phi" di Eulero** di n.

Sono fatti noti i seguenti:

Proposizione 7.4. Valgono i seguenti fatti:

(1) Se $p \in \mathbb{P}$ e $\alpha \in \mathbb{N}$, allora $\varphi(p^\alpha) = p^{\alpha-1}(p-1)$.

(2) Se $\gcd(n,m) = 1$, allora $\varphi(n \cdot m) = \varphi(n) \cdot \varphi(m)$.

(3) Se $n = \prod_{i=1}^{k} p_i^{\alpha_i}$, allora $\varphi(n) = \prod_{i=1}^{k} p_i^{\alpha_i - 1}(p-1)$.

Dimostrazione.

(1) Per provare l'asserto **(1)** è sufficiente notare che i numeri naturali divisibili per p e compresi tra 1 e p^α sono tutti e soli quelli della forma xp con $x = 1, \ldots, p^{\alpha-1}$, che sono in totale $p^{\alpha-1}$.

(2) Per provare l'asserto **(2)**, si considerino i seguenti insiemi:

$$\begin{aligned} X(mn) &= \{b = 1, \ldots, mn-1 : gcd(b, mn) = 1\} \\ X(m) &= \{x = 1, \ldots, m-1 : gcd(x, m) = 1\} \\ X(n) &= \{y = 1, \ldots, n-1 : gcd(y, n) = 1\} \end{aligned}$$

e si consideri, inoltre, l'applicazione

$$f : X(mn) \longrightarrow X(m) \times X(n), b \longmapsto (b_1, b_2),$$

dove $b_1 \equiv b \ mod \ m$ e $b_2 \equiv b \ mod \ n$ con $b_1 = 1, \ldots, m-1$ e $b_2 = 1, \ldots, n-1$. Ovviamente risulta $gcd(b_1, m) = gcd(b_2, n) = 1$. Per il **Teorema Cinese dei Resti**, il sistema

$$\begin{cases} b \ mod \ m = b_1 \\ b \ mod \ n = b_2 \end{cases}$$

essendo $gcd(m,n) = 1$, ammette sempre un'unica soluzione compresa tra 0 e $mn-1$, e questo implica che f è una biiezione. Quindi

$$\varphi(nm) = |X(mn)| = |X(m)||X(n)| = \varphi(n)\varphi(m).$$

(3) Infine, sia $n = \prod_{i=1}^{k} p_i^{a_i}$ con p_h, p_j primi distinti per $h, j = 1, \ldots, k$, $h \neq j$. Da **(1)** e **(2)** segue che

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

Pertanto, anche l'asserto **(3)** è dimostrato.

□

Teorema 7.5. ***(Teorema di Eulero).***
Siano n, x interi tali che $\gcd(x, n) = 1$, allora $x^{\varphi(n)} \equiv 1 \ mod\ n$.

Dimostrazione. Siccome l'ordine di $\mathbb{Z}_n^*$ è $\varphi(n)$, segue dal **Teorema di Lagrange** che

$$x^{\varphi(n)} \equiv 1 \ mod\ n.$$

□

Corollario 7.6. ***(Piccolo Teorema di Fermat).***
Siano p un primo e x un intero tale che $p \nmid x$, allora $x^{p-1} \equiv 1 \ mod\ p$.

Dimostrazione. Segue dal **Teorema di Eulero**, osservando che $\varphi(p) = p - 1$.

□

7.2 Il Crittosistema RSA

Il Crittosistema RSA è un cifrario inventato da **Rivest**, **Adleman** e **Shamir** nel 1977. E' uno dei cifrari più usati ad oggi e fonda la sua sicurezza sulla difficoltà computazionale della fattorizzazione di interi costituiti da un numero elevato di cifre.

Figura 7.1: Ronald Linn Rivest (1947), Leonard Max Adleman (1945) e Adi Shamir (1952)

Il Crittosistema RSA è definito come segue:

Definizione 7.7. **(Crittosistema RSA)**
Sia $n = pq$, con p, q primi distinti, siano $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$ e sia

$$\mathcal{K} = \{(n, p, q, a, b) : ab \equiv 1 \ mod \ \varphi(n)\}.$$

Infine, se $K \in \mathcal{K}$ allora

$$\begin{aligned} e_K &: \mathbb{Z}_n \longrightarrow \mathbb{Z}_n, x \longmapsto x^b \ mod \ n \\ d_K &: \mathbb{Z}_n \longrightarrow \mathbb{Z}_n, y \longmapsto y^a \ mod \ n \end{aligned}$$

La coppia (n, b) è detta **chiave pubblica**, la terna (p, q, a) è detta **chiave privata**.

Proviamo che di fatto l'RSA è un crittosistema. Come vedremo, questo è vero per il **Teorema di Eulero**.

Teorema 7.8. $d_K \circ e_K = Id_{\mathbb{Z}_n}$.

Dimostrazione. Analizziamo i casi $x \in \mathbb{Z}_n^*$ e $x \in \mathbb{Z}_n - \mathbb{Z}_n^*$ separatamente.

- **Supponiamo che** $x \in \mathbb{Z}_n^*$.
  Poiché $ab \equiv 1 \ mod \ \varphi(n)$, allora $ab = 1 + t\varphi(n)$ con $t \in \mathbb{Z}$. Quindi
  $$(x^b)^a = x^{ab} \equiv \left(x^{\varphi(n)}\right)^t x \ mod \ n \equiv \quad 1^t x \ mod \ n \equiv x \ mod \ n$$
  e quindi $d_K(e_K(x)) = x$ per $x \in \mathbb{Z}_n^*$.
- **Supponiamo che** $x \in \mathbb{Z}_n - \mathbb{Z}_n^*$.
  Se $x = 0$, la tesi è banalmente vera. Quindi, supponiamo che $0 < x \leq n - 1$. Se $\gcd(x, n) = p$, allora $\gcd(x, q) = 1$ e quindi $x^{q-1} \equiv 1 \ mod \ q$ per il **Piccolo Teorema di Fermat**. Allora
  $$x^{\varphi(n)} \equiv 1 \ mod \ q$$
  essendo $\varphi(n) = (p-1)(q-1)$. Quindi,
  $$(x^b)^a = x^{ab} = \left(x^{\varphi(n)}\right)^t x \equiv 1^t x \ mod \ q \equiv x \ mod \ q.$$
  D'altra parte, è banalmente vero che
  $$(x^b)^a \equiv 0 \ mod \ p \equiv x \ mod \ p$$
  siccome $p \mid x$. Quindi
  $$\begin{cases} x^{ab} \equiv x \ mod \ q \\ x^{ab} \equiv x \ mod \ p \end{cases}$$
  Siccome $p \neq q$ segue che $(x^{ab}) \equiv x \ mod \ n$. Cioè, $d_K(e_K(x)) = x$ per $x \in \mathbb{Z}_n - \mathbb{Z}_n^*$. Il caso $\gcd(x, n) = q$ è dimostrato in maniera analoga.

□

Esempio 7.9. L'utente B sceglie i primi $p = 101$ e $q = 113$. Allora $n = 11413$ e

$$\varphi(n) = (101-1)(113-1) = 11200 = 2^6 5^2 7,$$

sceglie un intero b primo con $\varphi(n)$, ovvero non divisibile per $2, 5$ e 7, per esempio $b = 3533$ allora b è invertibile in $\mathbb{Z}_{11200}$ e $a = b^{-1} \ mod\ 11200$ è 6597. Quindi la chiave pubblica dell'utente B è $(11413, 3533)$, quella privata è $(101, 113, 6597)$. Supponiamo che l'utente A voglia trasmettere l'unità di messaggio in chiaro $x = 9726$. Allora calcola

$$y = 9726^{3533} \ mod\ 11413 = 5761$$

e lo trasmette a B che una volta ricevuto calcola

$$x = 5761^{6597} \ mod\ 11413 = 9726.$$

(i) La sicurezza dell'RSA consiste nell'**assunto** che la funzione $e_K(x) = x^b \ mod\ n$ sia **trapdoor**, **one-way**. Quindi, è computazionalmente difficile invertirla se non si hanno delle informazioni aggiuntive.

(ii) La **trapdoor** che permette al destinatario prestabilito di decifrare un testo cifrato è la conoscenza della fattorizzazione di $n = pq$. Infatti, esso calcola immediatamente $\varphi(n) = (p-1)(q-1)$ e determina b, l'inverso di a modulo $\varphi(n)$ utilizzando l'algoritmo di euclideo esteso.

7.2.1 Implementare l'RSA

Inseguito analizzeremo i seguenti aspetti del crittosistema RSA:

I. Costruzione del crittosistema.

II. Efficienza della cifratura e della decifratura.

III. Il problema della sicurezza.

L'efficienza della cifratura e della decifratura è data dal fatto che il calcolo di $e_K(x) = x^b \ mod\ n$ e $d_K(y) = y^a \ mod\ n$, noti b e a, hanno entrambi complessità $O(\log^3 n)$. Quindi, rimangono da analizzare gli aspetti **I.** e **III.**

La costruzione del crittosistema RSA avviene attraverso un algoritmo che determina i parametri dell'RSA e opera come segue:

1. Genesi di due primi distinti p, q.
2. Calcolo di $n = pq$ e di $\varphi(n) = (p-1)(q-1)$.
3. Scelta di un intero casuale $1 < b < \varphi(n)$ tale che $\gcd(b, \varphi(n)) = 1$.
4. Calcolo dell'inverso a di b modulo $\varphi(n)$.
5. La chiave pubblica è (n, b), quella segreta è (p, q, a).

- I primi p, q devono essere grandi, altrimenti un attacco ovvio all'RSA consiste nella fattorizzazione di n (è stato congetturato che che violare l'RSA è polinomialmente equivalente alla fattorizzazione di n). In generale, è sufficiente che p, q siano interi di almeno 512 bit. Pertanto, n ha almeno 1024 bit e al momento non esistono algoritmi efficienti per la fattorizzazione di tali interi. Tralasciando per il momento lo step **(1)** che verrà analizzato in seguito in dettaglio e che si prova avere complessità $O(\log^3 n)$.
- Il calcolo dello n e $\varphi(n)$ ha complessità totale $O(\log^2 n)$.
- Per gli step **(3)** e **(4)** si utilizza l'algoritmo euclideo che ha complessità $O(\log^2 n)$.

Pertanto, l'algoritmo che sta alla base della costruzione dell'RSA ha complessità $O(\log^3 n)$ che è una funzione polinomiale del numero dei bit in un unità di messaggio in chiaro (o cifrato).

Se da una parte il crittosistema RSA è dimostrabilmente sicuro, dall'altra, come tutti i crittosistemi a chiave pubblica, non è incondizionatamente sicuro.
Infatti, un avversario osservando un testo cifrato y ed avendo infinite risorse computazionali valuta $e_K(x)$ al variare di x in $\mathbb{Z}_n$ fino a quando non ottiene $e_K(x_0) = y$ e quindi decifra y con x_0.

Quindi, in seguito analizzeremo la sicurezza computazionale dell'RSA.