

# SOLUZIONI

CRITTOGRAFIA: raccolta di esercizi d'esame (RSA, DH).

→ vedi testo

## Esercizio 1

- Spiegare in cosa consiste il cifrario RSA e dimostrarne la correttezza.
- Darne un esempio di applicazione impiegando parametri numerici molto piccoli per cifrare il messaggio costituito dalle due cifre meno significative del proprio numero di matricola.



## Esercizio 2

Posto che si scopra un algoritmo polinomiale per calcolare la funzione di Eulero, spiegare in termini matematici quale influenza la scoperta avrebbe sul cifrario RSA.

↳ si potrebbe calcolare in tempo polinomiale la chiave privata  $d = e^{-1} \pmod{\phi(n)}$ . Il cifrario sarebbe compromesso e non più utilizzabile

## Esercizio 3

Per la costruzione di una coppia di chiavi RSA si sceglie il numero  $n$  come prodotto di due primi  $p$  e  $q$  considerando le seguenti possibilità:

- $p = \Theta(n^{1/2}), q = \Theta(n^{1/2})$ .
- $p = O(n^{1/3}), q = O(n^{2/3})$ .
- $p = \Theta(n^{1/3}), q = \Theta(n^{1/3})$ .
- $p = O(\log n), q = O(n/\log n)$ .

Per ciascuna di queste possibilità spiegare con precisione se la scelta è corretta e consigliabile.

1) NO ( $p$  e  $q$  troppo vicini) 2) OK,  $p$  e  $q$  sono sufficientemente grandi e distanti tra loro.  
3) NO  $p, q$  non è  $\Theta(n)$  4)  $p$  è troppo piccolo, l'attacco bruteforce avrebbe costo polinomiale

## Esercizio 4

Si consideri un cifrario RSA con  $p = 7, q = 11, e = 13$ .

- Determinare il valore della chiave privata  $d$ .
- Qual è la dimensione dei blocchi per la cifratura?
- Cifrare 100011001010.

$$\begin{aligned} m_1 &= 35, & c_1 &= 35^{13} \pmod{77} = 63 \\ m_2 &= 10, & c_2 &= 10^{13} \pmod{77} = 10 \end{aligned}$$

$$C = \underbrace{11111}_{c_1} \underbrace{001010}_{c_2}$$

## Esercizio 5

Si consideri il cifrario RSA con chiave pubblica  $n = 55, e = 7$ .

- Cifrare il messaggio  $m = 10$ .
- Forzare il cifrario trovando  $p, q, d$ .
- Decifrare il crittogramma  $c = 35$ .

Da qui in poi, si vedono le pagine successive

## Esercizio 6

Siano  $x, y, n$  tre interi positivi arbitrari, con  $x < n, y < n$ . Poniamo che si scopra un algoritmo di algebra modulare di complessità  $O(d^2)$  per calcolare, se esiste, il logaritmo discreto di  $y$  in base  $x$  modulo  $n$ , con  $d = \Theta(n)$  oppure  $d = \Theta(\log n)$ .

Spiegare in termini matematici, per i due suddetti valori di  $d$ , quale influenza la scoperta avrebbe sull'algoritmo DH (Diffie-Hellman) per lo scambio segreto di chiavi.

## Esercizio 7

Si consideri il protocollo basato sull'algoritmo DH con  $g = 3$  e  $p = 353$ , e siano  $x = 97$  e  $y = 233$ .

Calcolare  $X, Y$  e la chiave  $k$ .

[Sol.  $k[\text{session}] = 160$ ]

$$X = g^{97} \pmod{353}$$

### Esercizio 8

Due utenti A, B vogliono costruire una chiave segreta di sessione impiegando il protocollo basato sull'algoritmo DH. A tale scopo concordano su una coppia pubblica di interi  $\langle p, g \rangle$ , con  $p = 11$  ( $p$  è piccolo per costruire il nostro esempio), e  $g = 6$ .

1. **Dimostrare** che la coppia  $\langle 11, 6 \rangle$  è adatta per il protocollo DH.
2. Posto che A e B scelgano come numeri “casuali” segreti  $x, y$  la terza e la quarta cifra del numero di matricola del candidato, **creare** la chiave di sessione **indicando i calcoli eseguiti da A e B.**

### Esercizio 9

Considerando il cifrario RSA:

1. Discutere se è possibile scegliere un valore pari per il parametro  $e$ .
2. Siano  $e$  ed  $e'$  due valori scelti per la chiave pubblica tali che  $e'$  è ottenuto da  $e$  cambiando un bit da 0 a 1. Dimostrare che  $\text{MCD}(e, e') = 1$ .

### Esercizio 10

Nonostante il cifrario RSA sia considerato un cifrario sicuro, alcune sue implementazioni possono rendere insicura la cifratura. Si consideri ad esempio la cifratura di un messaggio  $m$  di 64 bit con una chiave pubblica RSA  $\langle e, n \rangle$ , dove  $e = 3$  e  $n$  è un numero di 512 bit.

- **Spiegare** perché la cifratura di  $m$  è completamente insicura.
- **Decifrare** il crittogramma  $c = 33076161$  nel caso in cui  $n = 100082119$ .

### Esercizio 11

Si supponga che Eve intercetti un crittogramma  $c = m^e \pmod{n}$  diretto ad Alice. Si supponga inoltre che Alice sia disposta a decifrare per Eve qualsiasi crittogramma  $c'$ , a patto che  $c'$  sia diverso da  $c$ .

**Descrivere** come Eve possa decifrare  $m$  in tempo polinomiale, richiedendo ad Alice la decifrazione del crittogramma  $c' = c x^e$ , dove  $x < n$  è un intero casuale, co-primo con  $n$ .

### Esercizio 12

Alice vuole mandare un messaggio cifrato a Bob usando il cifrario RSA, ma non conosce la sua chiave pubblica. Quindi invia un email a Bob chiedendogli la chiave. Bob risponde inviando  $K[\text{pub}] = \langle e, n \rangle$ .

Eve intercetta il messaggio, sostituisce  $e$  con un nuovo intero  $e'$  coprimo con  $e$ , e invia la chiave modificata  $K'[\text{pub}] = \langle e', n \rangle$  ad Alice.

Alice usa  $K'[\text{pub}]$  per cifrare il messaggio  $m$ , e invia il crittogramma  $c' = m^{e'} \pmod{n}$  a Bob. Dato che  $m$  è stato cifrato con la chiave sbagliata, Bob non può decifrare e quindi rimanda la sua chiave pubblica ad Alice, chiedendole di inviare nuovamente il messaggio cifrato. A questo punto Alice invia il crittogramma corretto  $c = m^e \pmod{n}$ .

**Mostrare** come Eve (che ha spiato lo scambio di messaggi e conosce  $e, e', c, c'$ ) possa risalire al messaggio in chiaro  $m$ .

[**Suggerimento:** sfruttare il fatto che  $e$  ed  $e'$  sono coprimi.]

## ESERCIZIO 5

$$n = 55, e = 7$$

$$1) \quad m = 10 \quad c = 10^7 \bmod 55 = 10^{1+2+4} \bmod 55$$

$$10^2 \bmod 55 = 45$$

$$10^4 \bmod 55 = (45)^2 \bmod 55 = 45$$

$$\Rightarrow c = (10 * 10^2 * 10^4) \bmod 55 = (10 * 45 * 45) \bmod 55 = 10$$

$$2) \quad p = 5, q = 11$$

$$\phi(n) = (p-1)(q-1) = 40$$

$$d = 7^{-1} \bmod 40$$

$$EE(7, 40) \rightarrow \langle 1, -17, \dots \rangle$$

$$EE(40, 7) \rightarrow \langle 1, 3, -2 - 3 * \lfloor \frac{40}{7} \rfloor \rangle = \langle 1, 3, -17 \rangle$$

$$EE(7, 5) \rightarrow \langle 1, -2, 1 + 2 * \lfloor \frac{7}{5} \rfloor \rangle = \langle 1, -2, 3 \rangle$$

$$EE(5, 2) \rightarrow \langle 1, 1, -\lfloor \frac{5}{2} \rfloor \rangle = \langle 1, 1, -2 \rangle$$

$$EE(2, 1) \rightarrow \langle 1, 0, 1 \rangle$$

$$EE(1, 0) \rightarrow \langle 1, 1, 0 \rangle$$

$$\Rightarrow d = -17 \bmod 40 = 23$$

3)

$$c = 35$$

$$m = 35^{23} \bmod 55 = 35^{1+2+4+16} \bmod 55$$

$$35^2 \bmod 55 = 15$$

$$35^4 \bmod 55 = 15^2 \bmod 55 = 5$$

$$35^8 \bmod 55 = 5^2 \bmod 55 = 25$$

$$35^{16} \bmod 55 = 25^2 \bmod 55 = 20$$

$$m = (35 * 15 * 5 * 20) \bmod 55 = 30$$

## ESERCIZIO 6

Se  $d = \Theta(n)$ , l'algoritmo ha complessità esponenziale nella dimensione dell'input (è polinomiale solo nel valore di  $n$ )

nessuna inflessione su DH, da si prevedere in sicurezza

Se  $d = \Theta(\log n)$ , l'algoritmo è effettivamente polinomiale.  
Il protocollo DH non è più utilizzabile

## ESERCIZIO 7

$$q=3 \quad p=353 \quad x=97 \quad y=233$$

$$X = q^x \mod p = 3^{97} \mod 353 = 3^{64+32+1} \mod 353 = 40$$

$$Y = q^y \mod p = 3^{233} \mod 353 = 3^{128+64+32+8+1} \mod 353 = 248$$

↑ (da calcolare con quadrature successive)

$$\downarrow \quad K[\text{session}] = 40^{233} \mod 353 = 248^{97} \mod 353 = 160$$

$$248^{97} \mod 353 = 248^{64+32+1} \mod 353$$

$$248^2 \mod 353 = 82$$

$$248^4 \mod 353 = 82^2 \mod 353 = 17$$

$$248^8 \mod 353 = 17^2 \mod 353 = 289$$

$$248^{16} \mod 353 = 289^2 \mod 353 = 213$$

$$248^{32} \mod 353 = 213^2 \mod 353 = 185$$

$$248^{64} \mod 353 = 185^2 \mod 353 = 337$$

$$\Rightarrow 248^{97} \mod 353 = (337 * 185 * 248) \mod 353 = 160$$

## ESERCIZIO 8

1. Si deve dimostrare che 6 è un generatore di  $\mathbb{Z}_{11}^*$

$k$	1	2	3	4	5	6	7	8	9	10
$6^k \bmod 11$	6	3	7	9	10	5	8	4	2	1

✓

$$2. \text{ MAT} = 654321$$

$x \swarrow \searrow y$

$$X = 6^4 \bmod 11 = 9$$

$$Y = 6^3 \bmod 11 = 7$$

A calcola  $Y^x \bmod 11 = 7^4 \bmod 11 = 3$  (lezione) = 3  
 B calcola  $X^y \bmod 11 = 9^3 \bmod 11 = 3$

## ESERCIZIO 9

1) No, perché  $\phi(n)$  è sempre pari e risulterebbe  $\text{MCD}(\phi(n), e) = 2$ .

2) Sia  $e = e + 2^i$  ( $i$  i-esimo bit di  $e$  è 0)

¶ t t.c.  $t \mid e$  sappiamo che  $t \neq 2$  (vedi punto 1)

dunque  $t$  non può dividere  $2^i$

$\Rightarrow \nexists t : t \mid e, t \nmid e' \Rightarrow \text{MCD}(e, e') = 1$

## ESERCIZIO 10

1) m: numero di 64 bit      n: numero di 512 bit

per  $e=3$ ,  $m^e$  è un numero di  $3 \times 64 = 192$  bit

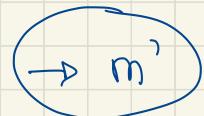
Dato che  $192 < 512$  non intuirebbe la riduzione in modulo, e si può decifrare facilmente estraendo la radice

del crittoalgoritmo

2)  $m = (33076161)^{1/3} = 321$

### ESERCIZIO 11

Eve chiede la decifrazione di  $c' = c \cdot x^e \pmod{n}$   
con x t.c.  $\text{MCD}(x, n) = 1$ .



Osserveremo che:

$$\begin{aligned} m' &= (c \cdot x^e)^d \pmod{n} = (c^d \pmod{n}) \cdot (x^{ed} \pmod{n}) \pmod{n} = \\ &= m \cdot x^{ed} \pmod{n} = m \cdot x^{1+r\phi(n)} \pmod{n} = \\ &= m \cdot x \cdot x^{r\phi(n)} \pmod{n} = mx \pmod{n} \end{aligned}$$

$\downarrow$   
si è usato il teorema di Euler  
( $x, n$  sono coprimi)

Eve può dunque decifrare c in questo modo:

- Sia  $m' = (c')^d \pmod{n}$  (decifrazione di  $c'$ )
- Eve calcola  $x^{-1} \pmod{n}$  (esistenza e unicità dell'inverso sono garantite dalla coprimità di  $n$  e  $x$ )
- Ottenere  $m$ :

$$m = (x^{-1} \cdot m') \pmod{n}$$

$$\begin{aligned} \text{Infatti: } (x^{-1} \cdot m') \pmod{n} &= x^{-1} \cdot mx \pmod{n} = mx x^{-1} \pmod{n} \\ &= m \pmod{n} \stackrel{\downarrow}{=} m \\ &\quad m < n \end{aligned}$$

## Esercizio 12

e ed  $e'$  sono coprimi. Applicando l'Algoritmo di Euclideo Esteso, si possono calcolare in tempo polinomiale  $r$  ed  $s$  t.c.

$$e \cdot r + e' \cdot s = \text{MCD}(e, e') = 1$$

$$\begin{aligned} \Rightarrow m &= m^{e^r + e's} \pmod{n} = (m^{e^r} \pmod{n}) \cdot (m^{e's} \pmod{n}) \pmod{n} \\ &= c^r (c')^s \pmod{n} \end{aligned}$$

Supponiamo  $r < 0$  e  $s > 0$ .

$$\Rightarrow m = (c^{-1})^{-r} (c')^s \pmod{n}$$

Eve calcola l'inverso di  $c$ , calcola  $(c^{-1})^{-r}$  ( $-r$  è positivo), calcola  $(c')^s$ , e ottiene  $m$  dal loro prodotto ridotto al modulo.

Tutti i passaggi richiedono tempo polinomiale.

L'inverso di  $c$  modulo  $n$  esiste ed è unico se  $c$  ed  $n$  sono coprimi. Se non lo fossero Eve potrebbe allora fattoriare  $n$  e franturare il cifrario.