Introduzione alla crittografia. Diffie-Hellman e RSA

Daniele Giovannini

Torino 2011, Crittografia a chiave pubblica: oltre RSA

Università degli Studi di Trento, Lab di Matematica Industriale e Crittografia

13 maggio 2011

1.INTRODUZIONE

Introduzione

CRITTOGRAFIA = scienza/arte delle scritture segrete

Obiettivo:

consentire a due utenti di comunicare su un canale potenzialmente insicuro, senza permettere a una terza persona di comprendere il contenuto dei messaggi

Alice

Bob

Alcune definizioni...

testo in chiaro

Sistema di cifratura:

- $\mathcal{M}$ = insieme di messaggi;
- $\mathcal{C}$ = insieme di testi cifrati;
- $\mathcal{K}$ = insieme di chiavi;
- ad ogni chiave K sono associate due funzioni: una che cifra (e_K) e una che decifra (d_K).

Due categorie di cifrari

- A CHIAVE PRIVATA (o SIMMETRICA)
- A CHIAVE PUBBLICA (o ASIMMETRICA)

Due categorie di cifrari

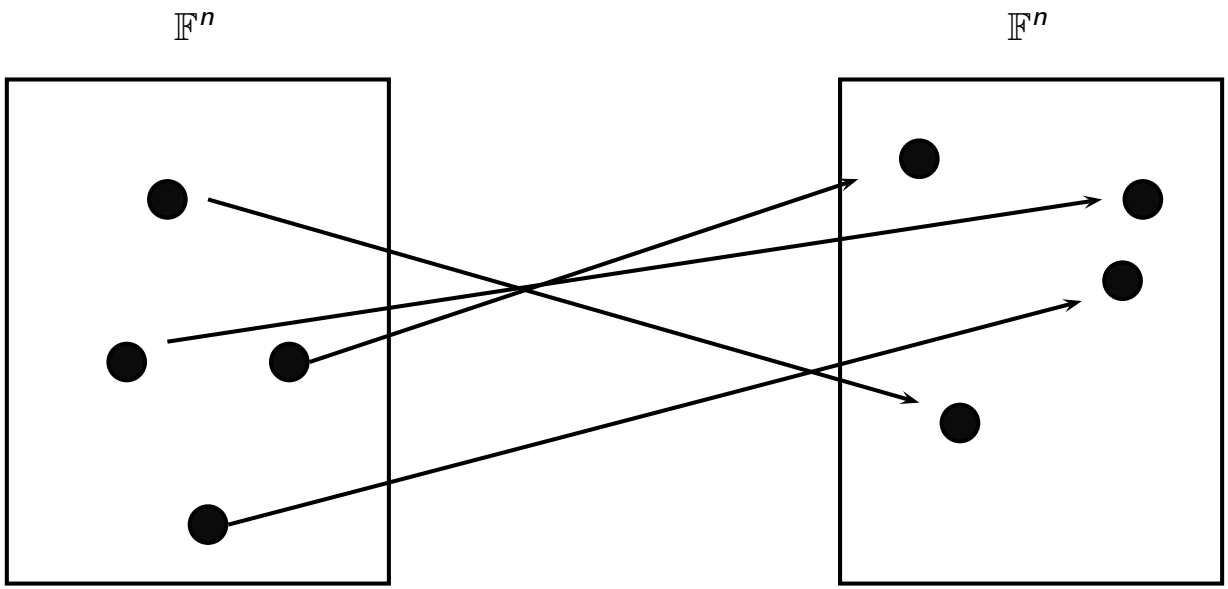
Cifrari a CHIAVE PRIVATA

- *un'unica* chiave privata K, sia per l'operazione di cifratura che per quella di decifratura
- solo  e  conoscono K ($\Rightarrow$ e_K e d_K)
- è necessario che prima Alice e Bob scelgano K, utilizzando un canale sicuro
- ci sono 2 tipi di cifrari a chiave simmetrica: i *Block Ciphers*, generalmente più sicuri ma pesanti, e gli *Stream Ciphers*, meno pesanti ma più insicuri

Cifrari a CHIAVE PRIVATA

L'insieme dei messaggi da trasmettere $\mathcal{M} = \mathbb{F}^n$, dove $\mathbb{F} = \{0, 1\}$
Possiamo pensarlo come l'insieme delle stringhe binarie lunghe n.

Una funzione di crittazione è una mappa $f : \mathcal{M} \to \mathcal{M}$ (bigettiva) che trasforma plaintext in ciphertext, in modo che a ciascun plaintext corrisponda un solo ciphertext e viceversa.



Cifrari a CHIAVE PRIVATA

Un cifrario a blocchi, φ, è un insieme di funzioni di crittazione

$$\{ \varphi_K \}_{K \in \mathcal{K}} \subset Sym(\mathcal{M})$$

dove ogni chiave K nello spazio delle chiavi $\mathcal{K}$ definisce l'azione di una funzione di crittazione.

ATTENZIONE: sappiamo calcolare

$$x \longrightarrow \varphi_K(x)$$

ma non "sappiamo" φ_K.

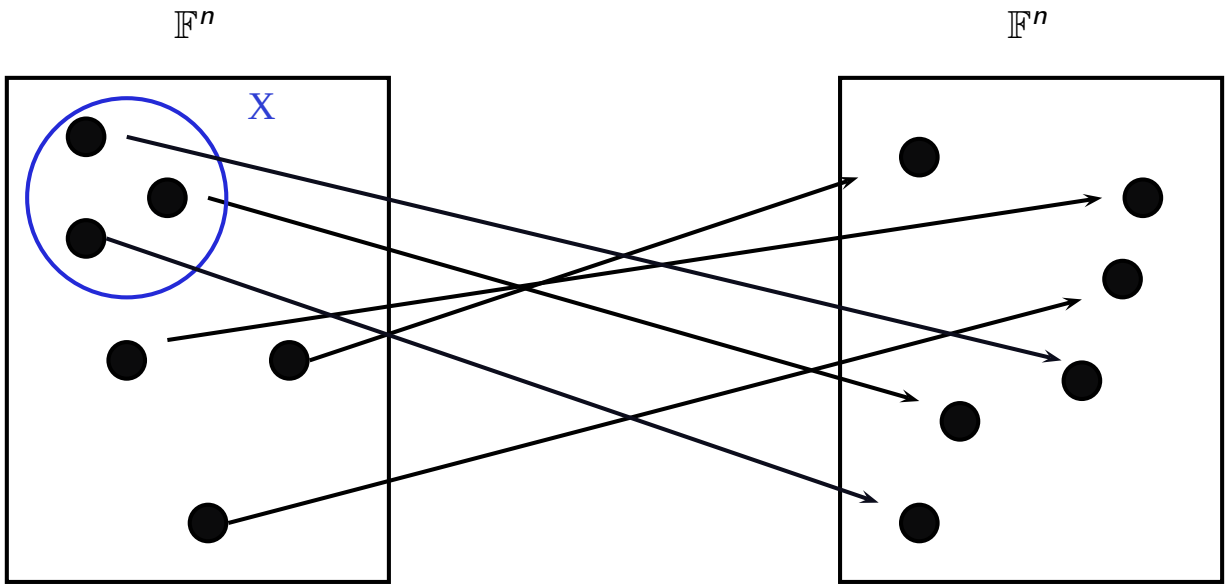
Cifrari a CHIAVE PRIVATA

Solitamente $\mathcal{K} = \mathbb{F}^k$.

Un buon cifrario non deve permettere di capire φ_K da

$$\{ (x, \varphi_K(x) \}_{x \in X}$$

per un piccolo $X \subset \mathcal{M}$. Tantomeno deve permettere di capire K.



Cifrari a CHIAVE PRIVATA

In particolare, l'azione del cifrario deve apparire completamente casuale ad un osservatore esterno.

$$\begin{array}{ccc} 1 & \rightarrow & m \\ 2 & \rightarrow & ? \\ 3 & \rightarrow & ?? \\ \ldots & \ldots & \ldots \end{array}$$

Quindi l'unico modo per rompere un cifrario ideale è provare tutte le chiavi (brute force), che costa

$$2^k \textit{ cifrature elementari}.$$

La sicurezza di un cifrario qualunque si rapporta a quella di un cifrario ideale.
Useremo k per confronto
(per il caso ideale k=k, in generale $k \geq k$)

Cifrari a CHIAVE PRIVATA

Esempio

Prendiamo un cifrario a blocchi in cui $\mathcal{M} = \mathbb{F}_2^{50}$ *e* $\mathcal{K} = \mathbb{F}_2^{50}$. *Quindi* $k = n = 50$.
Ovvero:

$$k = 50$$

Sia $K \in \mathcal{K}$ *la chiave segreta della funzione di crittazione:* $\varphi_K : \mathcal{M} \longrightarrow \mathcal{M}$.

Se questa funzione è lineare, allora ci bastano 50 *crittazioni per poter ricostruire la funzione di crittazione.*
In questo caso:

$$k = \log_2 50 \approx 6$$

Due categorie di cifrari

Cifrari a CHIAVE PUBBLICA

- Alice  sceglie una chiave pubblica $K_{p,a}$
- Bob  sceglie una chiave pubblica $K_{p,b}$
- Associata a ciascuna chiave pubblica, ci sono due chiavi private $K_{s,a}$ e $K_{s,b}$
- con le chiavi pubbliche si cifra il messaggio, $e_{K_{p,a}}$ e $e_{K_{p,b}}$; con quelle private invece si decifra $d_{K_{s,a}}$ e $d_{K_{s,b}}$

Due categorie di cifrari

Cifrari a CHIAVE PUBBLICA (scambio di chiavi)

I cifrari a chiave pubblica non hanno bisogno di un canale sicuro, come in quelli simmetrici.

Quindi possono essere usati per scambiare o negoziare una chiave segreta K da usare nei cifrari a chiave privata.

Confronto fra Cifrari

Cifrari a Chiave Privata:

- richiedono un canale sicuro per la comunicazione della chiave
- sono computazionalmente più leggeri anche quando si tiene alto il livello di sicurezza

Cifrari a Chiave Pubblica:

- non serve un canale sicuro
- sono computazionalmente più pesanti in quanto devono fare uno sforzo maggiore di quelli simmetrici

In questa presentazione vedremo alcuni cifrari a chiave pubblica, la cui robustezza e sicurezza sfrutta le difficoltà di:

- calcolare il logaritmo discreto
- fattorizzare un numero intero

1. Crittografia con il LOGARITMO DISCRETO

Che cosa è il Logaritmo Discreto

Il logaritmo discreto è in algebra astratta il corrispettivo del logaritmo $\log x$ ($e^{\log x} = x$).

In maniera più rigorosa:

Definizione

Sia G un gruppo ciclico con N elementi. Sia $g \in G$ tale che ogni elemento $b \in G$ può essere scritto nella forma $b = g^h$ per un certo intero h.

Chiamiamo h il logaritmo discreto di b, ovvero $\log_g b$.

Che cosa è il Logaritmo Discreto

Dato un gruppo finito, anche non abeliano, ogni suo elemento g genera un gruppo ciclico finito G.
Dato G, calcolare il logaritmo discreto di un suo elemento rispetto a g non è una cosa semplice.

Questo è il motivo per cui è usato in Crittografia.

Un metodo per calcolarlo, sarebbe la ricerca esaustiva, che però richiede un tempo di calcolo lineare rispetto a N e quindi esponenziale rispetto a $n = \log_2 N$ (il numero di cifre di N).

Calcolare il Logaritmo Discreto

Alcuni metodi per il calcolo del logaritmo discreto su un gruppo qualunque:

- Baby-step Giant-step
- algoritmo ρ di Pollard
- algoritmo di Pohlig-Hellman
- ...

I primi due si applicano sempre ma hanno complessità esponenziale in n.

L' Algoritmo Diffie-Hellman (DH)

L'algoritmo è particolarmente adatto per scambiarsi una chiave segreta attraverso un canale non sicuro.

Alice  e Bob  vogliono negoziare una chiave segreta.

Innanzitutto Alice sceglie un primo N e un intero g tale che $1 < g < N$.

(N, g) è la chiave pubblica $K_{p,a}$.

Alice sceglie un numero segreto a e trasmette g^a a Bob:


$A=g^a \longrightarrow$

Bob sceglie un numero segreto b e trasmette g^b ad Alice:


$B=g^b \longrightarrow$

A questo punto Alice calcola $B^a = g^{a \cdot b}$;

Bob calcola $A^b = g^{a \cdot b}$.

Quindi hanno ottenuto la stessa chiave segreta $K_s = g^{a \cdot b}$.

L' Algoritmo Diffie-Hellman (DH)

Se si riesce a risolvere il problema del logaritmo discreto (DLP), è anche possibile attaccare questo crittosistema.
Infatti ad un attaccante basta calcolare il logaritmo di A o B per poi ricostruire la chiave segreta $g^{a \cdot b}$.

$$\text{Risolvere DLP} \longrightarrow \text{Attaccare DH}$$

Il viceversa non è noto:

$$\text{Attaccare DH} \not\longrightarrow \text{Risolvere DLP ?}$$

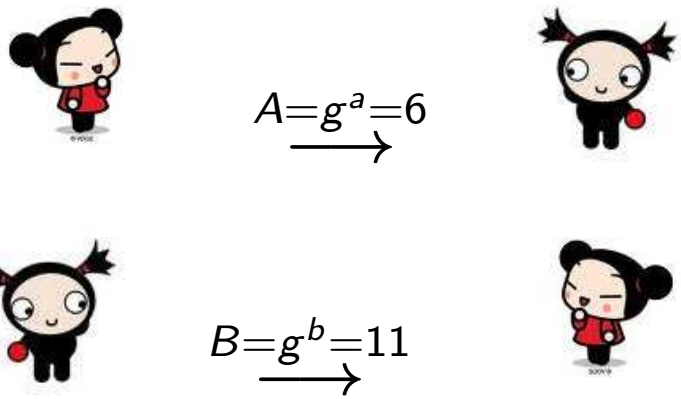
non sono noti al giorno d'oggi algoritmi che risolvono DH senza calcolare il logaritmo discreto.

L' Algoritmo Diffie-Hellman (DH)

Esempio

La chiave pubblica è data da $N = 13$ e $g = 2$. Questa è nota non solo ad Alice e a Bob, ma a qualunque terza parte in ascolto.

Alice sceglie $a = 5$ che mantiene segreto, Bob sceglie $b = 7$ che mantiene segreto.

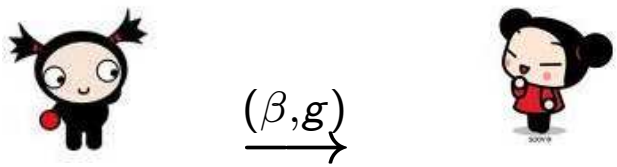


Alice calcola la chiave segreta $K_s = g^{a \cdot b} = B^a = 11^5 \equiv 7 \mod 13$.
Bob calcola la chiave segreta $K_s = g^{a \cdot b} = A^b = 6^7 \equiv 7 \mod 13$.

L' Algoritmo El Gamal

Questo è un altro crittosistema sempre basato sul logaritmo discreto.
Il messaggio m che Alice vuole trasmettere a Bob è un numero compreso tra 0 e N, con N primo noto.

Bob genera la chiave pubblica, prendendo g tale che $1 < g < N$ e un numero intero a segreto, $1 < a < N$. Calcola $\beta = g^a$ e trasmette la chiave pubblica (β, g) ad Alice.



L' Algoritmo El Gamal

Alice sceglie un intero segreto h, con $0 < h < N$ e critta m nel seguente modo:

$$g^h \equiv \gamma \mod N \qquad m\beta^h \equiv \delta \mod N$$

$\xrightarrow{(\gamma,\delta)}$

Per decifrare il messaggio, ora Bob calcola:

$$\gamma^{-a} \cdot \delta \equiv g^{-ah} \cdot m\beta^h \equiv g^{-ah} \cdot mg^{ah} \equiv m \mod N$$

L' Algoritmo El Gamal

Esempio

Riprendendo l'esempio di DH, scegliamo $N = 13$. *Bob sceglie* $g = 2$ *e* $a = 5$ *e trasmette la chiave pubblica ad Alice* $(\beta, g) = (g^a, g) = (6, 2)$.
Alice vuole trasmettere il messaggio $m = 9$.
Alice sceglie $h = 7$ *e calcola:*

$$(g^h, m\beta^h) = (\gamma, \delta) \equiv (11, 7) \mod N$$

$\xrightarrow{(11,7)}$

Bob decifra calcolando:

$$\gamma^{-a} \cdot \delta \equiv 11^{-5} \cdot 9 \cdot 6^7 \equiv 9 \mod 13$$

Crittosistemi basati sul Logaritmo Discreto

Abbiamo visto come funzionano DH e El Gamal sugli interi modulo un primo N. In realtà, per poter usare questi crittosistemi è sufficiente avere un qualsiasi gruppo finito e ciclico. Possiamo allora usare il logaritmo discreto in:

1. Crittografia su Campi Finiti $\mathbb{F}_q$
2. Crittografia con Curve Ellittiche (ECC)
3. Crittografia su Curve Iperellittiche (HCC)

ECC e HCC verranno trattati nelle successive slide.

Crittografia su Campi Finiti $\mathbb{F}_q$

Per costruire un crittosistema che sfrutta il DLP possiamo utilizzare un qualsiasi campo finito $\mathbb{F}_q$, dove $q = p^m$, p primo e m intero positivo.

Se togliamo lo 0 a $\mathbb{F}_q$ otteniamo un gruppo finito $\mathbb{F}_q^*$, che è ciclico grazie al *Teorema dell'elemento primitivo*.

Allora in questo caso la chiave pubblica dell'algoritmo è data da q e g che genera $\mathbb{F}_q^*$.

Crittografia su Campi Finiti $\mathbb{F}_q$

Per costruire un campo finito con p^m elementi, possiamo considerare l'insieme dei polinomi a coefficienti in $\mathbb{Z}_p = \{1,\ 2,\ 3,\ \ldots,\ p-1\}$ e prendere i resti della divisione per $f(x)$, polinomio irriducibile di grado m.

$$\mathbb{F}_q = \mathbb{Z}_p[x]/f(x) = \{h(x) \in \mathbb{Z}_p[x] \ \mid \ \deg h(x) < m\}$$

$(\mathbb{F}_q \setminus \{0\}, \cdot)$ è un gruppo moltiplicativo finito e ciclico su cui possiamo sfruttare le difficoltà di calcolo del logaritmo discreto.

Crittografia su Campi Finiti $\mathbb{F}_q$

I più veloci algoritmi per calcolare il logaritmo discreto su un campo finito sono quelli di tipo *index calculus*, che sono subesponenziali in n.

In particolare:

1. algoritmo di Coppersmith, se siamo su $\mathbb{F}_{2^m}$ con 2^m fino a 2^{613} ($n = m$)
2. NFS (number fields sieve), se siamo su $\mathbb{F}_p$, con p primo minore di 2^{530} ($n = \log_2(p)$)
3. FFS (function fields sieve), se siamo su un generico campo finito con p^m elementi, con p piccolo e p^m dell'ordine minore di 2^{673} ($n = m$)

2. L'ALGORITMO RSA

L'algoritmo RSA

L'RSA è un cifrario a chiave pubblica che permette di cifrare un messaggio sfruttando alcune proprietà elementari dei numeri primi.

Questo cifrario prende il nome dalle iniziali dei matematici che nel 1976 lo crearono: Rivest, Shamir e Adleman.

La loro idea fu quella di sfruttare la difficoltà di fattorizzare un numero intero. Di fatti la chiave pubblica è un numero N ottenuto moltiplicando due numeri primi molto grandi che restano segreti.

Descrizione di RSA

Per capire come funziona l'RSA abbiamo bisogno innanzitutto di due risultati dovuti a Eulero e Fermat.

Definizione (Funzione di Eulero φ)

La funzione di Eulero associa a un intero N il numero degli interi primi con N e minori di N (compreso 1).

$$\varphi(N) = |\{m \in \mathbb{N} \mid gcd(N, m) = 1,\ 1 \leq m < N\}|$$

Descrizione di RSA

Teorema (Teorema di Fermat-Eulero)

Dati due interi qualsiasi N e m, con $gcd(N, m) = 1$:

$$m^{\varphi(N)} \equiv 1 \mod (N)$$

Corollario (Piccolo Teorema di Fermat)

Se N è primo e m tale che $1 < m < N$:

$$m^{N-1} \equiv 1 \mod (N)$$

Descrizione di RSA

Possiamo così passare a descrivere il cifrario.

Alice  vuole trasmettere il messagio m a Bob  .

- Bob deve creare una chiave pubblica $K_{p,b} = (N, e)$ ed una privata $K_{s,b} = (N, d)$ stando attento a non divulgare d
- Bob trasmette (N, e) ad Alice

Descrizione di RSA

Come crea Bob la chiave pubblica e privata?

- Bob sceglie due numeri primi p e q, $p \neq q$, molto grandi e li moltiplica $N = p \cdot q$. Quindi $\varphi(N) = (p-1)(q-1)$
- Bob sceglie un numero e, coprimo con $\varphi(N)$ e $3 < e < \varphi(N)$
- Bob calcola d tale che
  $e \cdot d \equiv 1 \mod (p-1) \cdot (q-1)$
  d è l'inverso moltiplicativo di e
- solo Bob  conosce la chiave segreta (N, d) mentre (N, e) è pubblica

Descrizione di RSA

Come avviene la cifratura e decifratura?

- Alice calcola $c = m^e \mod N$, scegliendo m coprimo con N
- Alice trasmette c a Bob
- Bob riceve c e lo decritta calcolando:

$$c^d \equiv m \mod N$$

Infatti: $c^d \equiv m^{e \cdot d} \equiv m^{1+h \cdot \varphi(N)} \equiv m^1 \cdot \left(m^{\varphi(N)}\right)^h \equiv m \mod N$


Descrizione di RSA

La forza di questo algoritmo sta nel fatto che calcolare d, conoscendo la chiave (N, e), è un problema difficile.

Questo non esclude che ci sia un modo semplice di calcolarlo, ma al momento non si conosce.

Nella sezione relativa alla sicurezza di RSA approfondiremo quest'aspetto.

Attacchi ad RSA

Come si può ricavare la chiave privata da quella pubblica?

Apparentemente ci sono 3 alternative:

1. fattorizzare N
2. calcolare $\varphi(N)$
3. ricavare il valore d direttamente dalla chiave pubblica (N, e)

Si può facilmente dimostrare che questi tre metodi sono equivalenti:

$$(1) \Leftrightarrow (2) \Leftrightarrow (3)$$

Descrizione di RSA

Esempio

Bob sceglie $p = 3$ *e* $q = 11$. *Allora* $N = p \cdot q = 33$ *e* $\varphi(N) = 20$.
Bob sceglie $e = 7$ *che è coprimo con* 20 *e calcola* d *tale che* $d \cdot e \equiv 1 \mod \varphi(N)$, *ovvero* $d = 3$.

La chiave pubblica è $(33, 7)$, *quella privata* $(33, 3)$.
Se Alice vuole trasmettere il messaggio $m = 17$*:*

$$c = m^e \equiv 17^7 \equiv 8 \longrightarrow$$

A questo punto Bob calcola:

$$c^d \equiv (17^7)^3 \equiv 17^{21} \equiv 17^{1+20} \equiv 17 \equiv m \mod 33$$

Grazie per l'attenzione