Crittografia asimmetrica

RSA

1975 Diffie-Hellman-Merkle

- Tutti i sistemi di cifratura classici sono detti a chiave segreta ed utilizzano la stessa chiave sia per cifrare che per decifrare.
- Questo costituisce un problema non indifferente se pensiamo all'utilizzo della crittografia per la comunicazione a distanza, infatti le due parti devono riuscire in qualche modo a scambiarsi la chiave con la certezza che nessuno ne venga a conoscenza.
- La soluzione a questo tipo di problema fu proposta nel 1975 da Whitfield Diffie e Martin Hellman, col tributo di Ralph C. Merkle, che ebbero un'intuizione che rivoluzionò il mondo della crittografia.

Crittografia a chiave pubblica

- Diffie ed Hellman pensarono ad un sistema asimmetrico, basato su l'uso di due chiavi generate in modo che sia impossibile ricavarne una dall'altra. Le due chiavi vengono chiamate pubblica e privata: la prima serve per cifrare e la seconda per decifrare. Nell'esempio A deve comunicare con B.
- A cifra il messaggio con la chiave pubblica di B, che una volta ricevuto il messaggio può decifrarlo con la sua chiave privata.
- Ognuno possiede una coppia di chiavi, quella pubblica può essere tranquillamente distribuita perché consente solo di cifrare il messaggio, mentre quella privata deve essere conosciuta solo da una persona.
- Il problema è quello di trovare il modo di implementare matematicamente questo sistema, riuscire cioè a creare due chiavi per cui non sia possibile dedurre quella privata conoscendo quella pubblica.

Il meccanismo in azione

1977 RSA

- Un algoritmo a **chiave asimmetrica** è stato presentato nel 1977 da Ron **Rivest**, Adi **Shamir** e Leonard **Adleman** al **M**assachusetts **I**nstitute of **T**echnology.
- L'algoritmo è basato su particolari proprietà formali dei numeri primi.
- Non è sicuro da un punto di vista matematico teorico, in quanto esiste la possibilità che tramite la conoscenza della chiave pubblica si possa decrittare un messaggio, ma l'enorme mole di calcoli e l'enorme dispendio in termini di tempo necessario per trovare la soluzione, fa di questo algoritmo un sistema di affidabilità pressoché assoluta.
- Una variante del sistema RSA è utilizzato nel pacchetto di crittografia Pretty Good Privacy (PGP).
- L'algoritmo RSA costituisce la base dei sistemi crittografici su cui si fondano i sistemi di sicurezza informatici utilizzati sulla rete Internet per autentificare gli utenti.

RSA Funzionamento (semplificato)

- A deve spedire un messaggio segreto a B.
- B sceglie due numeri primi molto grandi (per esempio da 300 cifre) e li moltiplica con il suo computer (impiegando meno di un secondo).
- B invia il numero che ha ottenuto ad A. *Chiunque può vedere questo numero.*
- A usa questo numero per cifrare il messaggio
- A manda il messaggio cifrato a B, *chiunque può vederlo ma non decifrarlo*
- B riceve il messaggio e utilizzando i due fattori primi che solo lui conosce lo decifra.
- A e B hanno impiegato pochi secondi a cifrare e decifrare, ma chiunque avesse intercettato le loro comunicazioni impiegherebbe troppo tempo per scoprire i due fattori primi, con cui si può decifrare il messaggio.
- In realtà A e B si scambieranno con questo sistema una chiave segreta (che non occupa molto spazio), che poi useranno per comunicare tra loro usando un sistema a crittografia simmetrica, più semplice e veloce.

Primo passo: Generare le chiavi

Formal

1. Choose two primes p and q with $p \neq q$
2. Calculate their product: $N = p * q$
3. Calculate the value of Euler's totient function of N
   $$\varphi(N) = \varphi(p * q) = (p-1)(q-1)$$
4. Choose a number e between 1 and $N-1$ which is coprime to $\varphi(N)$
5. Find another number d where
   $$d * e \equiv 1 \; mod \; \varphi(N)$$

Example

1. Suppose we select $p = 13$ and $q = 7$
2. Thus: $N = 13 * 7 = 91$
3. $\varphi(91) = \varphi(13 * 7) = (13-1)(7-1) = 72$
4. Suppose we choose $e = 5$, because: $gcd(5, 72) = 1$
5. We will select $d = 29$ as thus: $d * e = 145 = 2 * 72 + 1 \equiv 1 \; mod \; 72$

(e, N) **is the *public RSA key.***
(d, N) **is the *private key.***

Passo 2: Crittare il messaggio

- First we must convert the letters into numbers to be able to use them in our calculations.

For example you can use the following substitution:

A	B	C	D	…	Z
01	02	03	04	…	26

Formal

To encrypt a message we have to calculate

$$C \equiv K^e \; mod \; N$$

Here K is the converted message and C is the encoded text, the ciphertext. The numbers e and N are taken from the public RSA key.

Example

We shall continue our example by encoding the word "SECRET":

Letters	S	E	C	R	E	T
Numbers	19	05	03	18	05	20

Now we take the first letter S = 19 and encrypt it by using the public key: $(5, 91)$

$K^e = 19^5 = 19 * (19^2)^2 = 19 * (361)^2$
$\equiv 19 * (88)^2 \equiv 19 * 9 = 171 = 80 \; mod \; 91$

Following this pattern, "SECRET" is encrypted as follows:

80	31	61	44	31	76

Passo 3: Decrittare il messaggio

- The receiver gets the message now in its encrypted form only.

Formal

To decipher the original message the receiver needs to calculate the following:

$$K \equiv C^d \; mod \; N$$

Here K will produce the plaintext. The values d and N are saved in the receiver's private key (d, N).

Example

The encrypted message is as follows:

30	31	61	44	31	76

According to the formula given to left, he or she can decipher by using his or her private key $(29, 91)$:

$$C^d = 30^{29} = \cdots \equiv 19 \; \; mod \; 91$$

The complete plaintext is obtained by calculating accordingly for each value.

Nnumbers	19	05	03	18	05	20
Letters	S	E	C	R	E	T

Sicurezza di RSA

- Per quanto riguarda l'algoritmo RSA l'attacco a forza bruta (ovvero ottenere i due numeri primi usati per creare la chiave pubblica), è una procedura lentissima.
- L'attacco più veloce è durato 5 mesi utilizzando 80 processori da 2,2GHz
- Questi dati consentono di dire che l'algoritmo è sufficientemente sicuro.