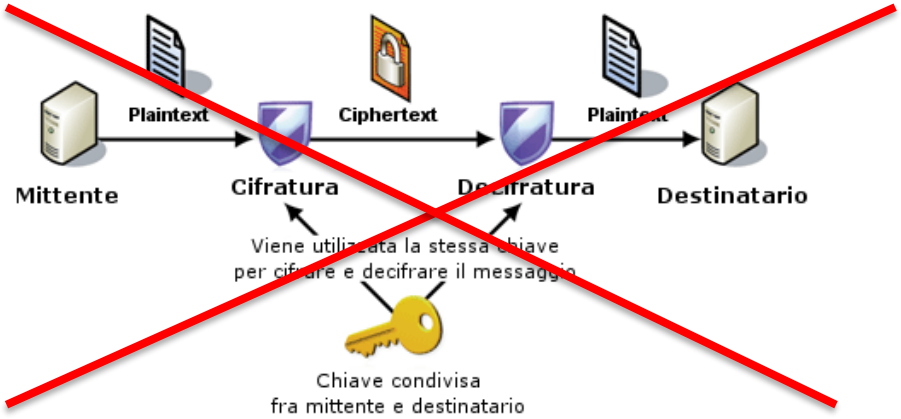


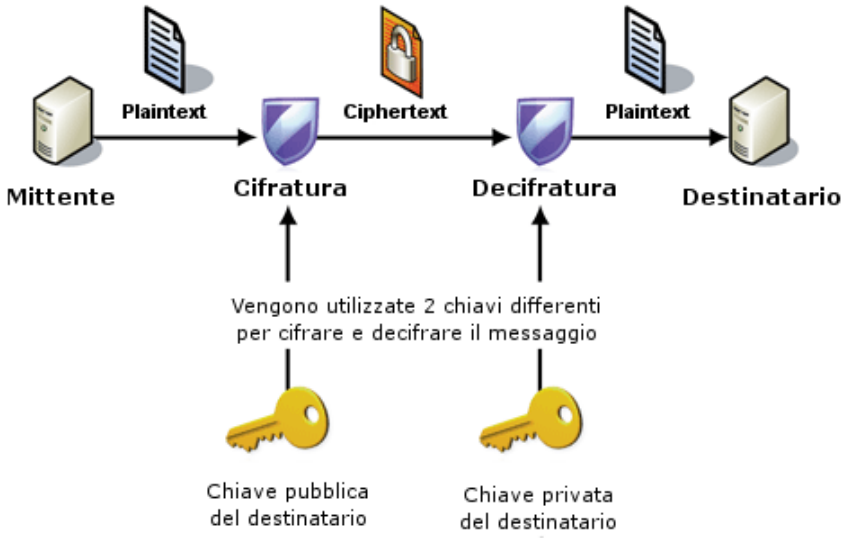


# crittografia moderna

A. Ferrari

# 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
- la chiave segreta costituisce un ***problema*** per l'utilizzo della crittografia per la comunicazione a distanza
  - 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

A. Ferrari

- sistema ***asimmetrico***
  - basato su l'uso di ***due chiavi***
  - generate in modo che sia ***impossibile*** ricavarne una dall'altra
- ***chiave* pubblica**
  - per cifrare
- ***chiave privata***
  - per decifrare
- mittente ***cifra*** il messaggio con la ***chiave pubblica*** del destinatario
- destinatario ***decifra*** il messaggio con la ***chiave privata*** segreta
- ogni persona possiede una ***coppia*** di chiavi
  - quella pubblica è di pubblico dominio
  - quella privata deve essere conosciuta solo dal possessore
- problema
  - implementare riuscire a creare due chiavi per cui **non sia possibile dedurre quella privata conoscendo quella pubblica**

# cifrare e decifrare



# Il meccanismo in azione



# 1977 RSA



- algoritmo a **chiave asimmetrica** descritto nel 1977 da Ron ***R***ivest, Adi ***S***hamir e Leonard ***A***dleman al **M**assachusetts **I**nstitute of **T**echnology
- basato su proprietà formali dei numeri primi
- ***non è sicuro da un punto di vista matematico teorico***
  - esiste la ***possibilità*** di ***risalire*** alla chiave privata dalla chiave pubblica
  - usando numeri molto grandi l'***enorme*** mole di ***calcoli*** e l'enorme ***tempo*** necessario per trovare la soluzione rende l'algoritmo un sistema di affidabilità pressoché assoluta
- una variante del sistema RSA è utilizzato nel pacchetto di crittografia Pretty Good Privacy (***PGP***)
- RSA è alla base dei sistemi crittografici su cui si fondano i sistemi di sicurezza informatici utilizzati inInternet per autentificare gli utenti

# RSA funzionamento (semplificato)

A. Ferrari

- 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*

# $Z_n$ con n primo

A. Ferrari

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| f(x) = 5x mod n | 0 | 5 | 10 | 4 | 9 | 3 | 8 | 2 | 7 | 1 | 6 |
| $f^{-1}(x)$ = 9f(x) mod n | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

- f(x) "mescola" l'insieme dei valori
- $f^{-1}(x)$ "riordina" l'insieme dei valori
- le due funzioni sono moltiplicazioni modulo n
- 5 è intesa come Ke
- 9 è il reciproco di 5 modulo 11 è intesa come Kd

# funzione di Eulero

A. Ferrari

- la funzione φ di Eulero è una funzione definita, per ogni intero positivo n, come il numero degli interi compresi tra 1 e n che sono coprimi con n
- esempio: φ(8)=4
  - i numeri coprimi di 8 sono quattro: 1, 3, 5, 7
  - dato un numero primo p φ(p)=p-1
- la funzione φ di Eulero è moltiplicativa
  - per ogni coppia di interi a e b tali che MCD(a, b)=1 si ha:
  - φ(ab)=φ(a) φ(b)
- se p e q sono numeri primi e N = p·q
  - $\varphi(N) = \varphi(p \cdot q) = \varphi(p) \cdot \varphi(q) = (p-1) \cdot (q-1)$

# algoritmo RSA

- si decidono due numeri primi molto grandi $p$ e $q$
  - attualmente vengono utilizzati numeri con circa 300 cifre
- si calcola il prodotto $N = p \cdot q$
- *si calcola* $\varphi(N) = \varphi(p \cdot q) = \varphi(p) \cdot \varphi(q) = (p - 1) \cdot (q - 1)$
- si calcola la *chiave pubblica* cercando un numero $e$ coprimo e minore di $(p-1) \cdot (q-1)$
- la chiave privata $d$ è l'inverso di $e$ *modulo* $(p-1) \cdot (q-1)$
  - $d \cdot e \equiv 1 \ mod \ \varphi(N)$

# esempio

- p=13 e q=7
- $N = 13 \cdot 7 = 91$
- $\varphi(N) = \varphi(13 \cdot 7) = (13-1) \cdot (7-1) = 72$
- supponiamo $e = 5$ $\quad MCD(5,72) = 1$ (coprimi)
- allora troviamo $d = 29$
  - $d \cdot e = 145 = 1 \bmod 72$

- S = messaggio sorgente (convertito in numero)
- C = messaggio crittato
- $C = S^e \ mod \ N$
- provare a crittare un testo convertendo in codici ASCII i caratteri utilizzando le chiavi definite precedentemente

# decrittare

- S = messaggio sorgente (convertito in numero)
- C = messaggio crittato
- D = messaggio decrittato
- $D = C^d \ mod \ N$
- provare a decrittare il risultato della crittazione precedente

- per quanto riguarda l'algoritmo RSA l'attacco a ***forza bruta*** (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
- oggi vengono utilizzati numeri primi molto più grandi
- questi dati consentono di dire che l'algoritmo è sufficientemente sicuro

# CrypTool

- CrypTool è un software libero e open source di e-learning per Microsoft Windows che illustra i concetti fondamentali della crittografia in via pratica.
- scritto in C++, è disponibile in inglese, in tedesco, in spagnolo e in polacco
- la versione scritta in Java, che prende il nome di JCrypTool, è disponibile da agosto 2007.
- https://www.cryptool.org/en/