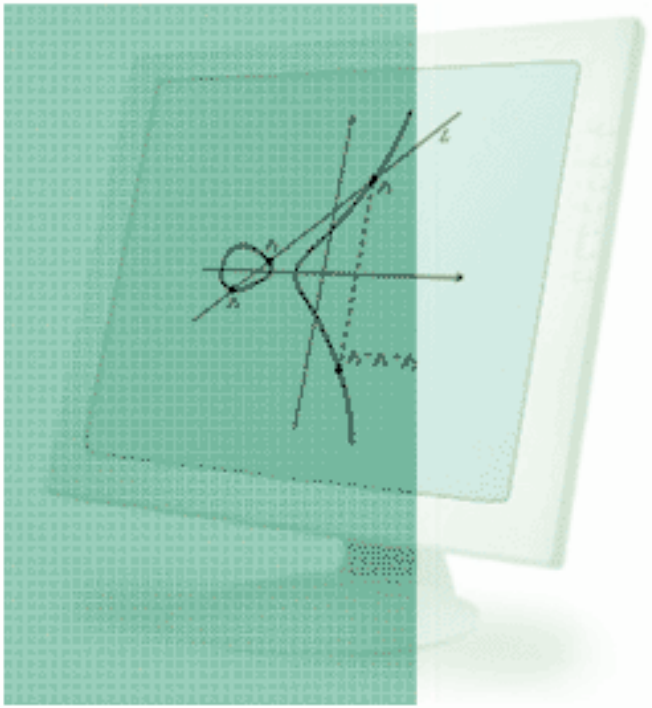
Uno sguardo alla crittografia moderna

di Enrico Zimuel (enrico@zimuel.it)

Liceo scientifico “Nicolò Copernico”
Brescia – 15 marzo 2004



Note sul copyright (copyfree):

Questa presentazione può essere utilizzata liberamente
a patto di citarne la fonte e non stravolgerne il contenuto.

Questa presentazione è stata creata con OpenOffice 1.1
www.openoffice.org
ed è disponibile su Internet all'indirizzo
http://www.zimuel.it/conferenze/liceocopernico.sxi
o in formato Acrobat Pdf all'indirizzo
http://www.zimuel.it/conferenze/liceocopernico.pdf

Sommario

- Che cos'è la crittografia?
- Gli ambiti d'utilizzo della crittografia: identificazione, autenticazione, riservatezza, integrità, anonimato
- La crittografia simmetrica ed il problema della trasmissione della chiave
- La crittografia asimmetrica o a chiave pubblica
- Le funzioni hash e la firma digitale
- Le frontiere matematiche della crittografia
- La crittografia quantistica

Che cos'è la crittografia?

- La **crittografia** (dal greco *kryptos*, nascosto, e *graphein*, scrivere) è la scienza che si occupa dello studio delle scritture "segrete".
- E' nata come **branca della matematica e dell'informatica** grazie all'utilizzo di tecniche di teoria dei numeri e di teoria dell'informazione.
- E' una disciplina antichissima, le cui origini risalgono forse alle prime forme di comunicazione dell'uomo, anche se si è sviluppata come scienza vera e propria solo dopo la seconda guerra mondiale .

La crittografia moderna

- Le basi teoriche della moderna crittografia, quella attualmente utilizzata, risalgono a circa 30 anni fa a partire dal 1969 con le prime ricerche di **James Ellis** del quartier generale governativo delle comunicazioni britanniche (GCHQ).
- Sviluppata ed affinata nel 1976 in America grazie al contributo di **Whitfield Diffie** e **Martin Hellman** con la nascita del termine crittografia a *chiave pubblica.*
- Nasce nel 1977 il cifrario a chiave pubblica **RSA** da tre ricercatori del MIT (Massachusetts Institute of Technology), **Ronald Rivest, Adi Shamir e Leonard Adelman** .
- Nel 1991 viene rilasciata la prima versione del software PGP (Pretty Good Privacy) di **Phil Zimmermann**, la crittografia diventa una realtà quotidiana.

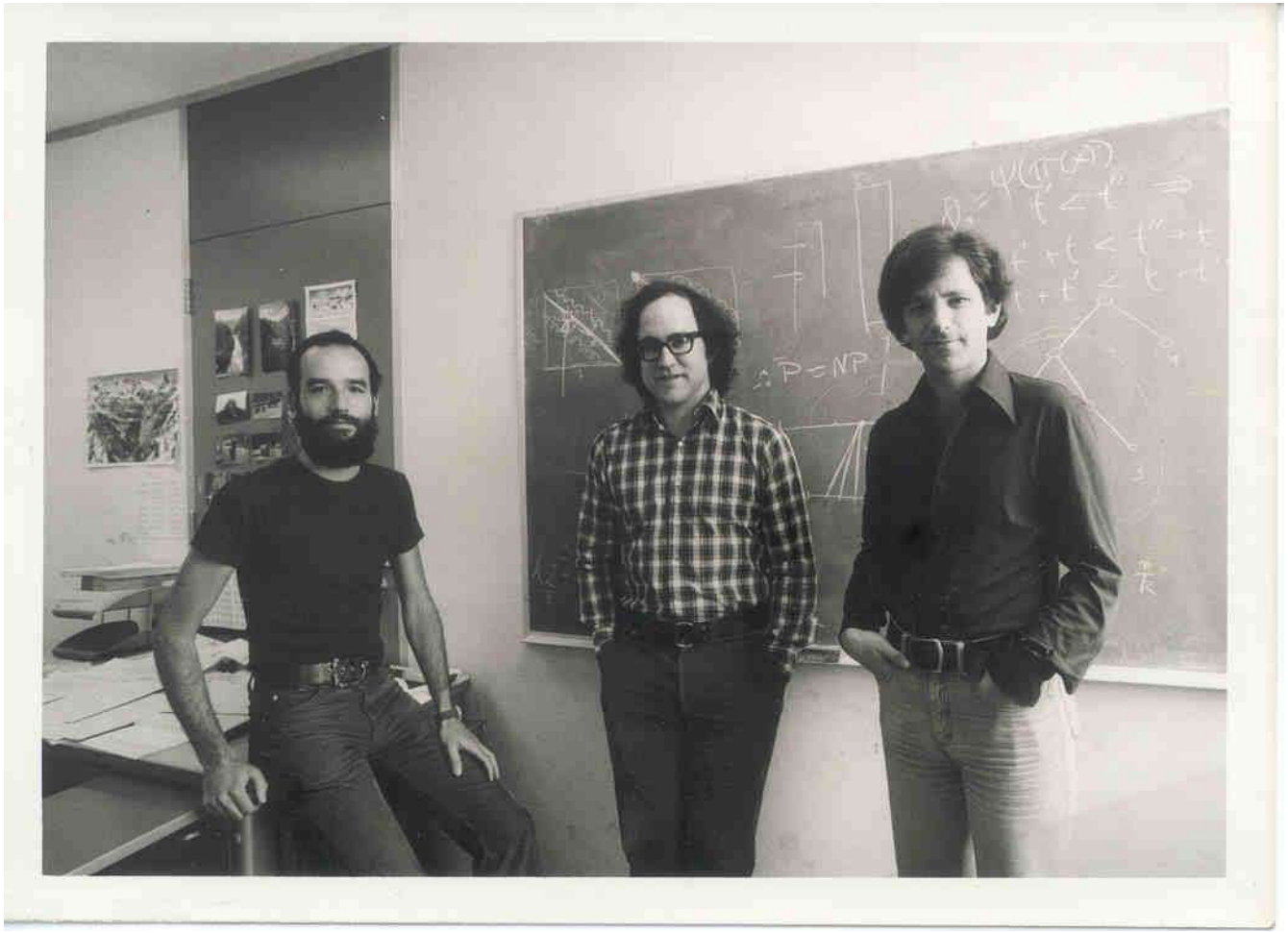


Foto di gruppo del 1977, da sinistra:
Adi Shamir, Ronald Rivest e Leonard Adelman

Gli ambiti d'utilizzo della crittografia

- La crittografia viene utilizzata principalmente per implementare le seguenti operazioni: *autenticazione*, *riservatezza*, *integrità*, *anonimato*.
- L'*autenticazione* è l'operazione che consente di assicurare l'identità di un utente in un processo di comunicazione.
- La *riservatezza* è l'operazione più conosciuta della crittografia perchè è quella che storicamente è nata per prima e che consiste nel proteggere le informazioni da occhi indiscreti.
- L'*integrità* è l'operazione che consente di certificare l'originalità di un messaggio o di un documento. In pratica si certifica che il messaggio non è stato modificato in nessun modo.
- L'*anonimato* è l'operazione che consente di non rendere rintracciabile una comunicazione, è una delle operazioni più complesse da realizzare.

Alcuni esempi d'utilizzo della crittografia

- Nei **bancomat** come sistema di protezione delle comunicazioni tra POS (Point Of Sale, punto di vendita) e banca.
- Nella telefonia mobile, ad esempio nel protocollo **GSM** tramite l'algoritmo A5/2* o nel procotollo **UMTS**, per la protezione delle comunicazioni vocali.
- Nelle comunicazioni satellitari per l'autenticazione e la protezione delle trasmissioni dati satellitari, ad esempio con lo standard **SECA2** impiegato dalla maggior parte delle Tv Digitali.
- Su Internet per la protezione del commercio elettronico e delle comunicazioni riservate (protocollo **SSL**).
- Nelle applicazioni di firma dei documenti digitali (**firma digitale**).

* l'algoritmo A5/2 è stato violato nel 2003 da un gruppo di ricercatori israeliani, per maggiori informazioni http://www.portel.it/news/news2.asp?news_id=8311

Le operazioni di cifratura e decifrazione

- Definiamo con **Msg** "l'insieme di tutti i messaggi" e con **Critto** "l'insieme di tutti i crittogrammi".
- ***Cifratura***: operazione con cui si trasforma un generico messaggio in chiaro **m** in un crittogramma **c** applicando una funzione **C**: Msg->Critto.
- ***Decifrazione***: operazione che permette di ricavare il messaggio in chiaro **m** a partire dal crittogramma **c** applicando una funzione **D**: Critto -> Msg.
- Matematicamente D(C(m))=m le funzioni C e D sono una inversa dell'altra e la funzione C deve essere iniettiva, ossia a messaggi diversi devono corrispondere crittogrammi diversi.

Che cos'è un cifrario?

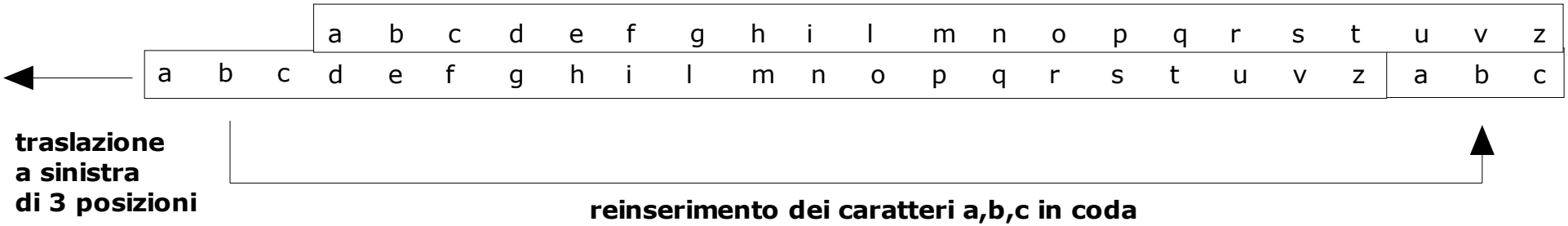
- Un cifrario è un sistema, di qualsiasi tipo, in grado di trasformare un testo in chiaro (messaggio) in un testo inintellegibile (testo cifrato o crittogramma).

- Per poter utilizzare un cifrario è necessario definire due operazioni: la cifratura del messaggio e la decifrazione del crittogramma.

Un primo esempio di cifrario: il cifrario di Cesare

- Consideriamo l'alfabeto italiano, costruiamo un cifrario che sostituisce ad ogni lettera di questo alfabeto la lettera che si trova 3 posizioni in avanti.

Cifrario di Cesare



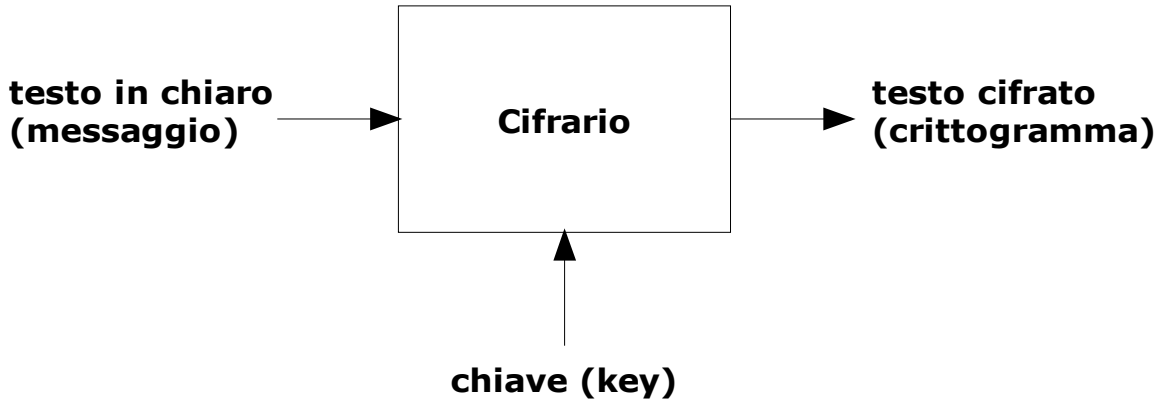
- Ad esempio il testo in chiaro “prova di trasmissione” viene cifrato nel crittogramma “surbd gn zudvpnvvnrqh”.

Crittoanalisi del cifrario di Cesare

- Il cifrario di Cesare, come la maggior parte dei cifrari storici basati tu trasposizioni e traslazioni, può essere facilmente violato utilizzando tecniche statistiche (crittoanalisi statistica).
- Si analizzano le frequenze relative dei caratteri nel testo cifrato e le si confrontano con quelle di una lingua conosciuta, ad esempio l'italiano.
- Le frequenze relative al testo cifrato "surbd gn zudvpnvvnrqh" risultano *s* (1/19), *u* (2/19), *r* (2/19), *b* (1/19), *d* (2/19), *g* (2/19), *n* (3/19), *z* (1/19), *v* (3/19), *p* (1/19), *h* (1/19).
- Si confrontano tali frequenze con quelle della lingua italiana: *a* (0,114), *e* (0,111), *i* (0,104), *o* (0,099), *t* (0,068), *r* (0,065),...
- Con queste informazioni ottengo una prima approssimazione del testo in chiaro "s**ro**b**a** g**i** z**ra**vp**i**vv**io**qh", procedo per tentativi ripetendo il procedimento.

La crittografia simmetrica

- Introduciamo un parametro chiamato **k** (key= chiave) all'interno delle funzioni di cifratura **C(m,k)** e decifrazione **D (c,k)**.
- Si parla di crittografia simmetrica perchè si utilizza la stessa chiave k per le operazioni di cifratura e decifrazione.
- La robustezza del cifrario dipende, a differenza di prima, solo dalla segretezza della chiave **k**.



Il principio di Kerckhoffs

- Risulterà strano ma uno dei principi fondamentali della crittografia, utilizzato ancora nei moderni sistemi crittografici è stato individuato nel lontano 1883 dal linguista franco-olandese August Kerckhoffs nel suo celebre articolo "La cryptographie militaire" apparso nel Journal des sciences militaires.
- Principio di Kerckhoffs: *"La sicurezza di un sistema crittografico è basata **esclusivamente** sulla conoscenza della chiave, in pratica si presuppone noto a priori l'algoritmo di cifratura e decifrazione."*
- Purtroppo alcuni sistemi crittografici proprietari moderni non rispettano questo essenziale principio di sicurezza.

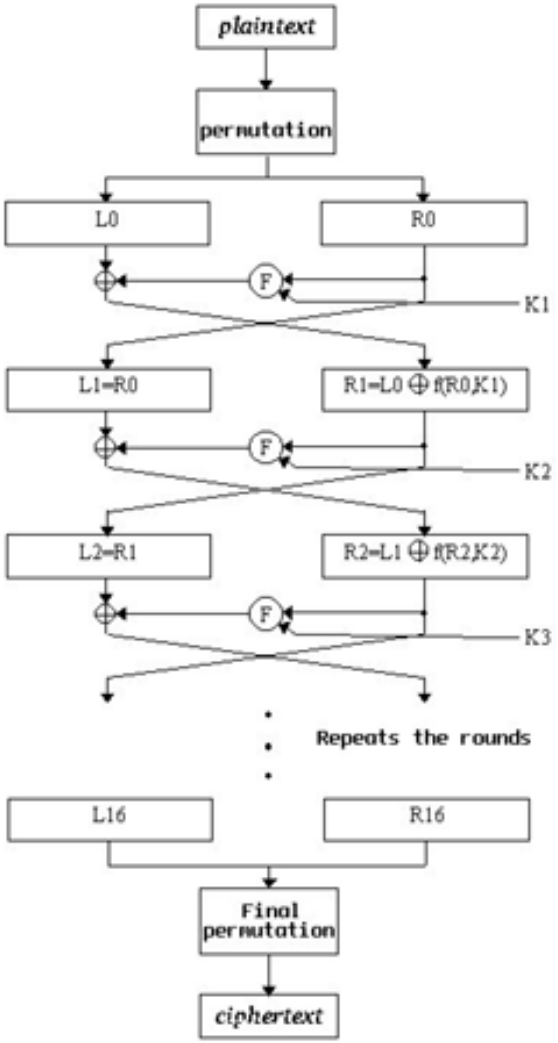
Il problema della trasmissione della chiave

- Volendo utilizzare un cifrario simmetrico per proteggere le informazioni tra due interlocutori come posso scambiare la chiave segreta? Devo utilizzare una "canale sicuro" di comunicazione.

- Ma tale "canale sicuro" esiste nella realtà?
- Per una comunicazione sicura tra n utenti si dovranno scambiare in tutto (n-1)*n/2 chiavi, ad esempio con 100 utenti occorreranno 4950 chiavi, il tutto per ogni comunicazione!

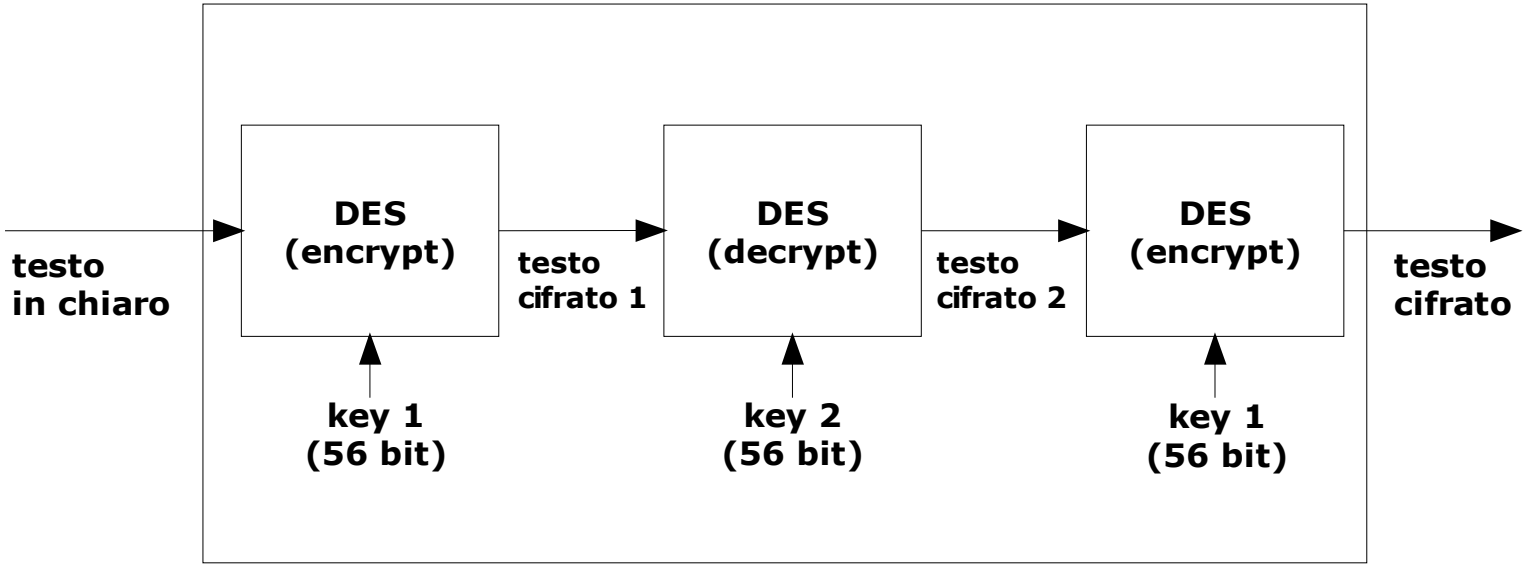
DES (Data Encryption Standard)

- Sviluppato dall'IBM nel 1970 diventato standard nel 1976.
- Utilizza chiavi di 56 bit, divide il testo in chiaro in blocchi di 64 bit, effettua delle permutazioni iniziali e finali ed un ciclo di 16 iterazioni di permutazioni e xor (Feistel network, tecniche di confusione e diffusione).
- Il 17 Luglio 1998, l'EFF (Electronic Frontier Foundation) costruisce un sistema dedicato in grado di violareill DES in meno di 3 giorni, tramite un attacco di tipo "brute-force".
- Morale della favola: non utilizzate sistemi di cifratura basati sul DES!



3DES

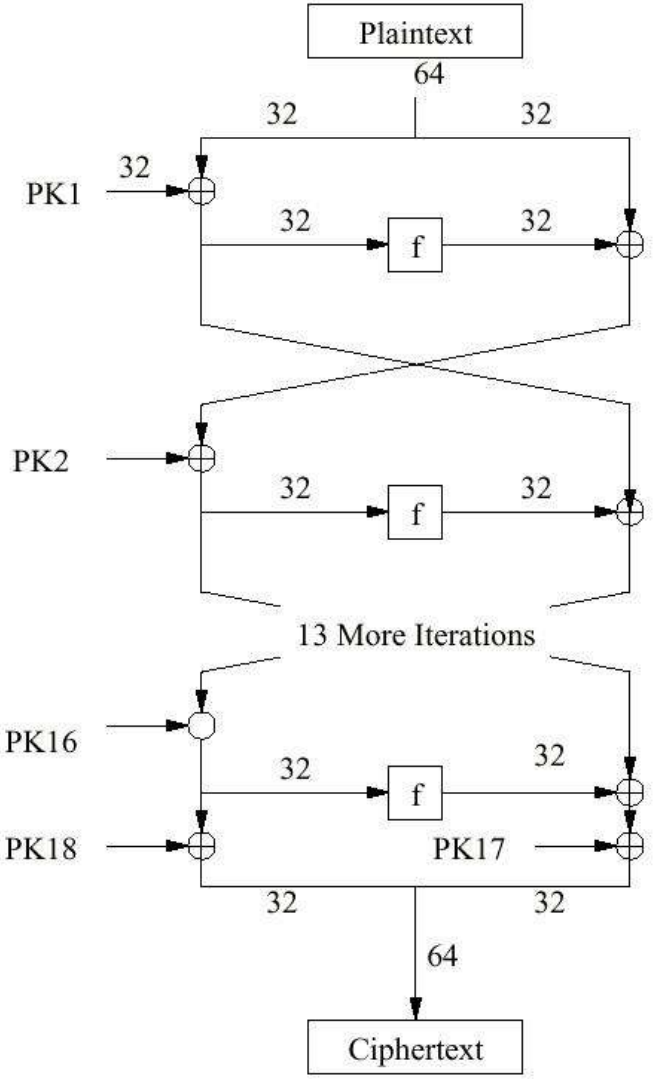
- Evoluzione del DES, è basato su un utilizzo del cifrario DES ripetuto, chiavi di 112 bit.
- Si utilizza la tecnica della codifica-decodifica-codifica (EDE, Encrypt-Decrypt-Encrypt) utilizzando il cifrario DES.



3DES (key = key1+key2, 112 bit)

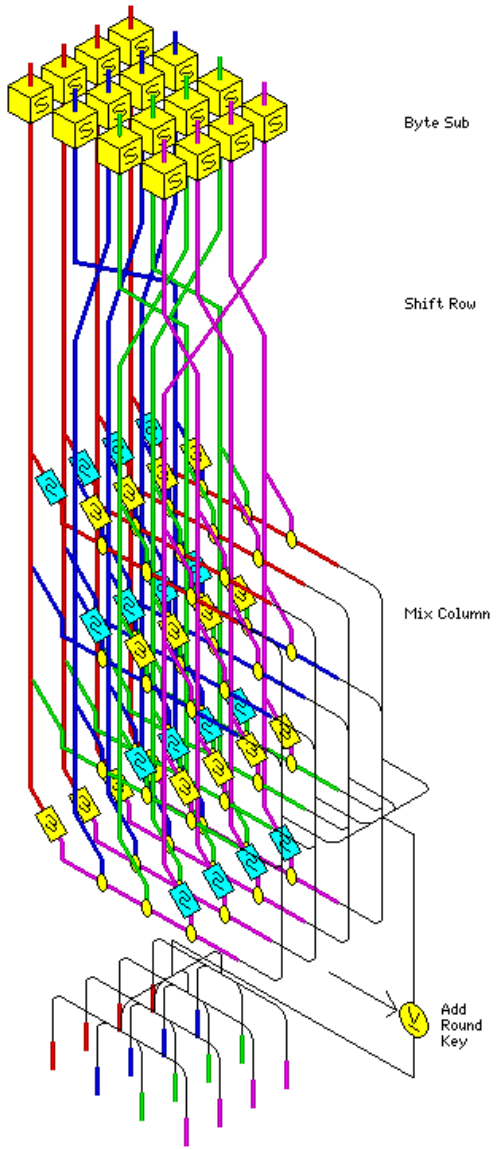
Blowfish

- Ideato nel 1993 da Bruce Schneier.
- E' stato sviluppato come algoritmo di encryption: veloce, compatto, semplice da implementare e sicuro con chiavi di dimensioni variabili fino a 448 bit.
- E' un cifrario a blocchi di 64 bit, basato sulle reti di Feistel.
- Non si conoscono attacchi efficaci.
- E' un algoritmo non patentato, utilizzato in molti sistemi open source (come ad esempio in OpenBSD).



Rijndael (AES)

- Sviluppato Joan Daemen e Vincent Rijmen.
- Questo algoritmo ha vinto la selezione per l'Advanced Encryption Standard (**AES**) il 2 Ottobre 2000. Ufficialmente il Rijndael è diventato lo standard per la cifratura del XXI secolo.
- Il cifrario utilizza chiavi di lunghezza variabile 128, 192, 256 bit (gli autori hanno dimostrato come è possibile variare le dimensioni delle chiavi con multipli di 32 bit). Lo schema del Rijndael è stato influenzato dall'algoritmo SQUARE.

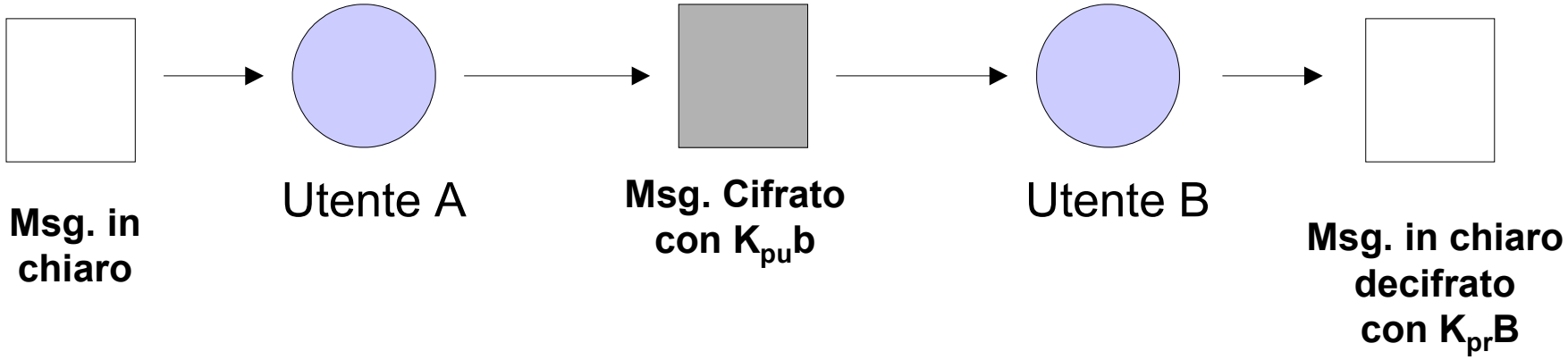


La crittografia a chiave pubblica

- Utilizza una coppia di chiavi per le operazioni di cifratura (*encryption*) e decifrazione (*decryption*).
- Una chiave detta pubblica (**public key**) viene utilizzata per le operazioni di encryption.
- L'altra chiave, detta privata (**private key**), viene utilizzata per le operazioni di decryption.
- A differenza dei cifrari simmetrici non è più presente il problema della trasmissione delle chiavi.
- Sono intrinsecamente sicuri poiché utilizzano tecniche di tipo matematico basate sulla teoria dei numeri, sulla teoria delle curve ellittiche, etc.

La crittografia a chiave pubblica

- Esempio di encryption (trasmissione sicura):

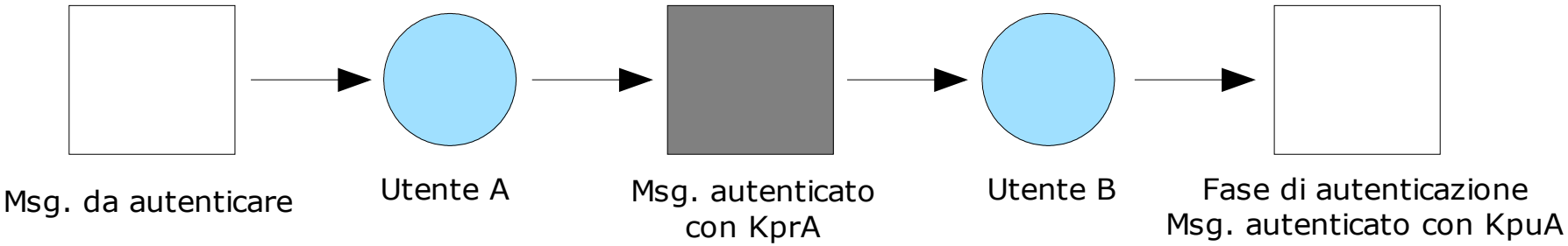


$K_{pu}B$ = chiave pubblica dell'utente B

$K_{pr}B$ = chiave privata dell'utente B

La crittografia a chiave pubblica

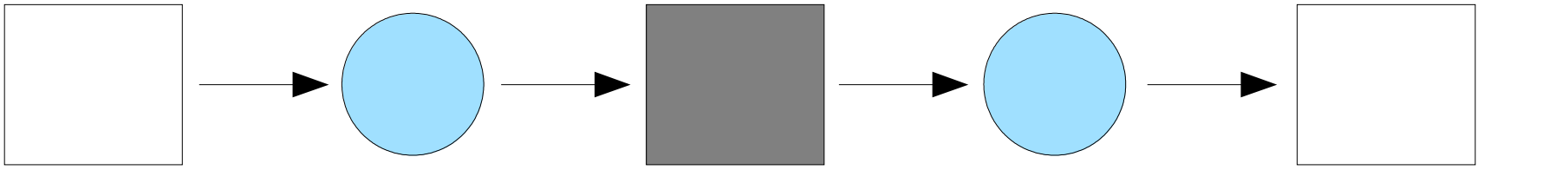
- Esempio di autenticazione:



KprA = chiave privata dell'utente A
KpuA = chiave pubblica dell'utente A

La crittografia a chiave pubblica

- Esempio di encryption ed autenticazione:



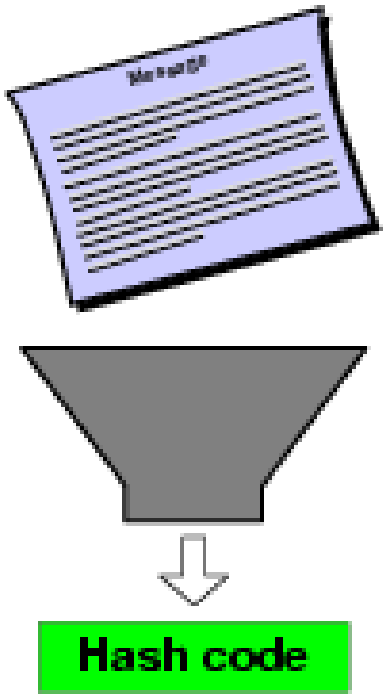
KprA = chiave privata dell'utente A
KpuA = chiave pubblica dell'utente A
KprB = chiave privata dell'utente B
KpuB = chiave pubblica dell'utente B

La firma digitale e le funzioni hash sicure

- Nasce come applicazione dei sistemi a chiave pubblica.
- Viene utilizzata per autenticare la paternità di un documento informatico e la sua integrità.
- Si utilizza un cifrario a chiave pubblica e si “cifra” un documento (file) con la propria chiave segreta. Chiunque può verificare la paternità del documento utilizzando la chiave pubblica dell'utente firmatario.
- Problema: per l'autenticazione di un documento di grandi dimensioni con un algoritmo a chiave pubblica occorre molto tempo.
- Soluzione: posso autenticare solo un “riassunto” del documento tramite l'utilizzo di una funzione hash sicura.

Le funzioni hash sicure

- Vengono utilizzate per generare un sorta di "riassunto" di un documento informatico (file).
- Una funzione hash accetta in ingresso un messaggio di lunghezza variabile M e produce in uscita un digest di messaggio H(M) di lunghezza fissa.
- Questo digest (impronta digitale, targa, riassunto) è strettamente legato al messaggio M, ogni messaggio M genera un H(M) univoco.
- Anche considerando due messaggi M ed M' differenti solo per un carattere le loro funzioni hash H(M) e H(M') saranno diverse.



Requisiti di una funzione hash sicura H(x):

- H può essere applicata a un blocco di dati di qualsiasi dimensione;
- H produce in uscita un risultato di lunghezza fissa (ad esempio 160 bit);
- Per qualunque codice h il calcolo di x tale che H(x)=h deve avere una complessità computazionale improponibile;
- Per qualunque blocco di dati x il calcolo di $y \neq x$ tale che H(x) =H(y) deve avere una complessità computazionale improponibile.
- Ai fini pratici H(x) deve essere relativamente semplice da calcolare.

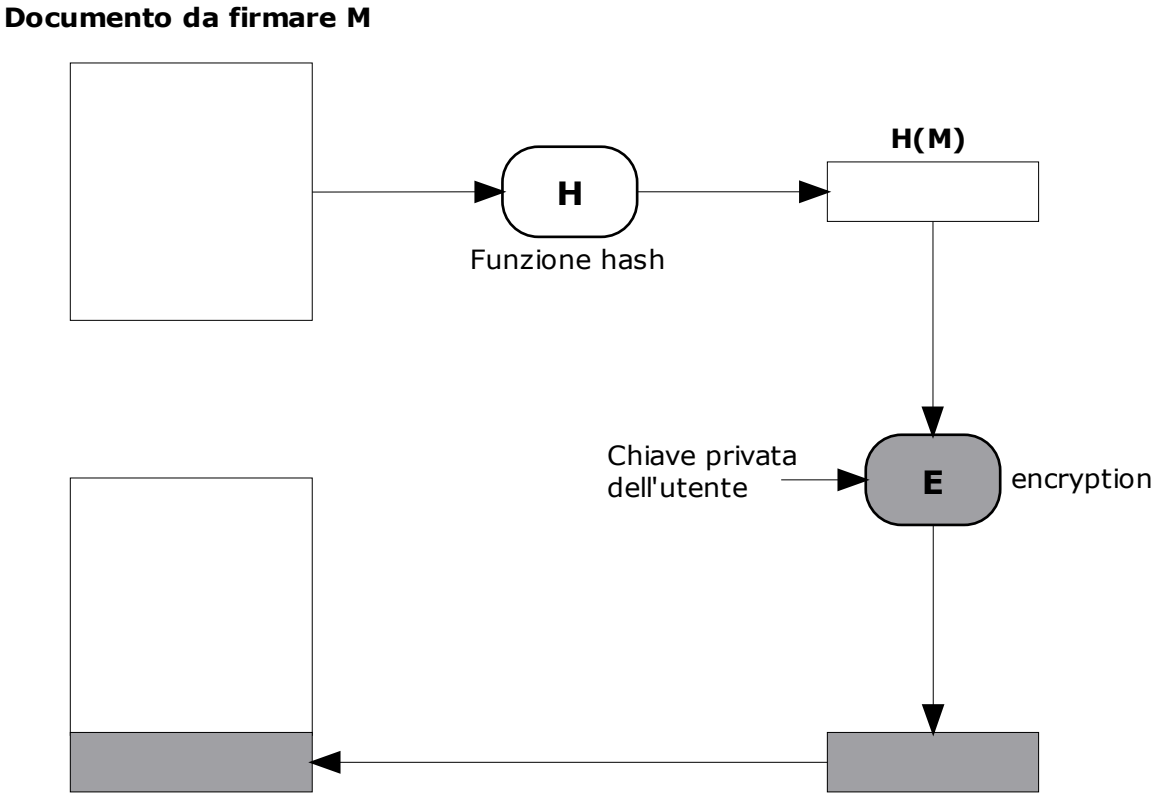
Esempio di funzione hash:

- Tutte le funzioni hash operano sulla base del seguente principio: i dati in ingresso sono considerati come una sequenza di blocchi di n bit, essi vengono elaborati un blocco alla volta iterativamente per produrre una funzione hash di n bit.
- Una delle più semplici funzioni hash è quella che esegue lo XOR bit a bit di ciascun blocco, ossia:

$$C_i = b_{i1} \oplus b_{i2} \oplus \ldots \oplus b_{im}$$

- Dove C_i rappresenta l'i-esimo bit del codice hash, m il numero di blocchi di n bit, b_{ij} l'i-esimo bit all'interno del j-esimo blocco e l'operatore $\oplus$ l'operazione di XOR.
- La probabilità che un errore nei dati produca lo stesso valore hash è 2^{-n}, con n=128 bit $2^{-128} \approx 2{,}9387 * 10^{-39}$.

Esempio di firma digitale di un documento:

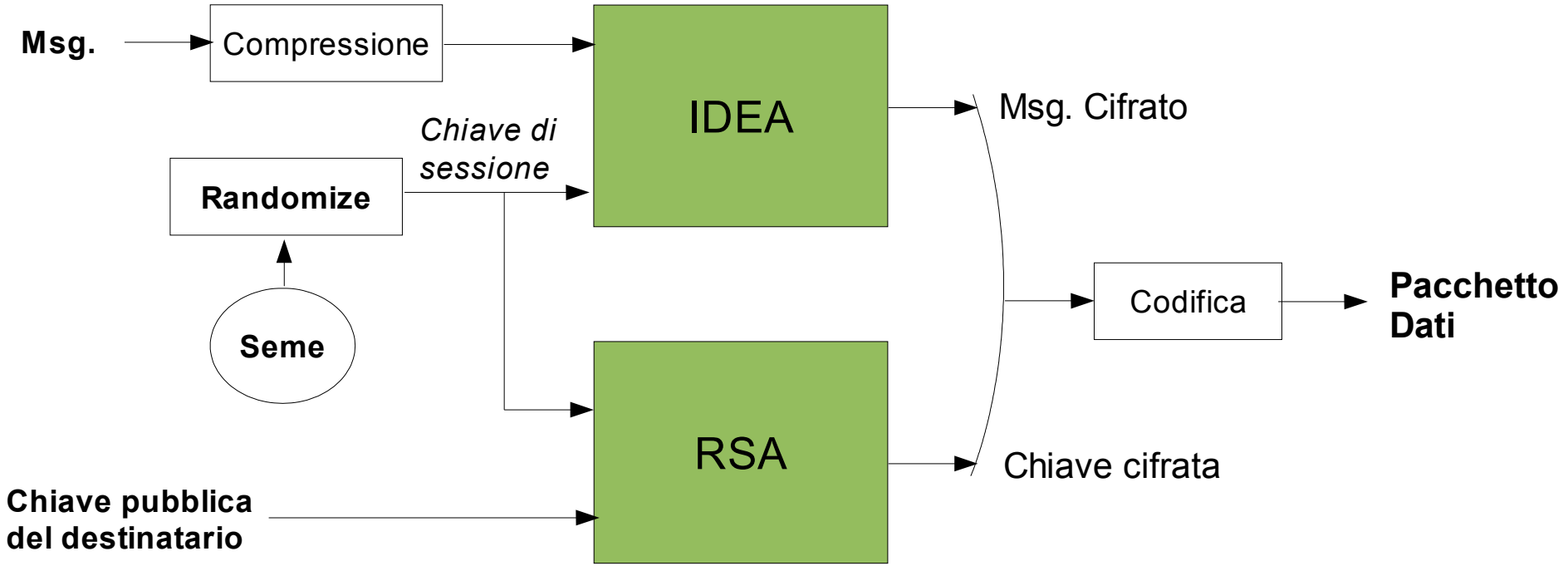


Documento firmato:
Il ricevente può verificare
la firma utilizzando la
chiave pubblica dell'utente firmatario
e riapplicando la funzione hash

Esempio di un sistema crittografico ibrido: il PGP

- PGP (Pretty Good Privacy) è un software di pubblico dominio creato da Phil Zimmermann nel 1991.
- E' un software per la privacy personale: protezione delle email, dei files, firma digitale.
- Utilizza gli algoritmi di crittografia a chiave pubblica RSA, Diffie-Hellman, DSA e gli algoritmi simmetrici IDEA, CAST, 3-DES.
- E' basato su di un sistema di crittografia "ibrido" nel senso che utilizza crittografia simmetrica per le operazioni di encryption sui dati generando delle chiavi di sessione pseudo-casuali cifrate con un algoritmo a chiave pubblica.
- Attualmente il PGP viene distribuito dalla Pgp Corporation come software commerciale. Sul sito www.pgp.com è disponibile una versione gratuita, la 8.0.3, per scopi non commerciali.

Il funzionamento del PGP: esempio di cifratura



Perchè la crittografia è in grado di garantire la sicurezza?

- Perchè è basata sull'impossibilità di risolvere, allo stato attuale, dei problemi matematici in tempi "ragionevoli".
- In altre parole la sicurezza della crittografia è basata sulla difficoltà di risoluzione di alcuni problemi, ad esempio come il problema della fattorizzazione di grandi numeri.
- Siano p e q due numeri primi, scelti a caso, di dimensioni elevate (dell'ordine di 10^{20}) e sia $n=pq$ il prodotto di questi primi. Conoscendo solo n è molto difficile scomporlo nei suoi fattori primi, ossia calcolare p e q.
- Non esiste, allo stato attuale, un algoritmo di risoluzione del problema della fattorizzazione in tempi "ragionevoli" (al più polinomiali).
- La sicurezza del cifrario **RSA** è basata proprio su questo assunto.

Gare di fattorizzazione su Internet

- La società RSA Security offre 20'000 $ al primo che riesca a fattorizzare il seguente numero di 193 cifre decimali:

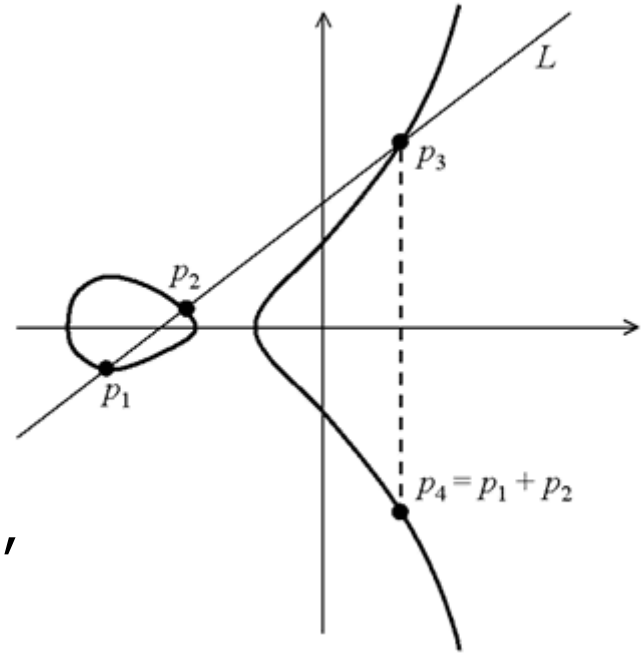
  n=3107418240490043721350750035888567930037346022842727545720161948823206440518081504556346829671723286782437916272838033415471073108501919548529007337724822783525742386454014691736602477652346609

  p=?, q=?

- Per chi volesse cimentarsi in questa impresa può consultare il sito Internet della RSA Security:
  http://www.rsasecurity.com/rsalabs/challenges/factoring/numbers.html

Le frontiere matematiche della crittografia

- Fondamentalmente la base della crittografia è la teoria dei numeri e la costruzione di funzioni "difficilmente" invertibili.
- La tendenza attuale è quella di utilizzare insiemi costruiti su strutture algebriche particolari come le curve ellittiche.
- Una curva ellittica è l'insieme dei punti (x,y), su di un piano cartesiano, che soddisfano le condizioni di un'equazione cubica, ad esempio $y^2=x^3+ax+b$, con l'aggiunta di un elemento denotato *O* e chiamato *punto infinito*.



Alcuni argomenti di ricerca

- **NTRU** è un Crittosistema a chiave pubblica nato nel 1982 da un famoso articolo di J. Hoffstein, J. Pipher e J. H. Silverman. Il sistema si costruisce sull'anello dei polinomi troncati ad un certo grado N a coefficienti interi modulo q. Come prodotto si usa il prodotto di convoluzione tra i vettori delle coordinate dei polinomi.
- **Reticoli interi**: Closest Vector Problem (CVP) e Shortest Vector Problem (SVP) sembrano costituire una solida primitiva per sistemi crittografici, data la loro appartenenza alla classe dei problemi NP-hard, cioè i problemi computazionalmente intrattabili.
- **Algoritmo LLL e derivati**: LLL è il famosissimo algoritmo polinomiale di A. K. Lenstra, H. W. Lenstra e L. Lovász che approssima SVP.
- **Pseudorandom:** Generazione di vettori pseudorandom utilizzando il "random walk" su gruppi finiti (o più in generale compatti). Sistemi lineari ricorsivi di generazione e comportamento della distanza variazionale sulla distribuzione uniforme.

La rivoluzione della crittografia quantistica

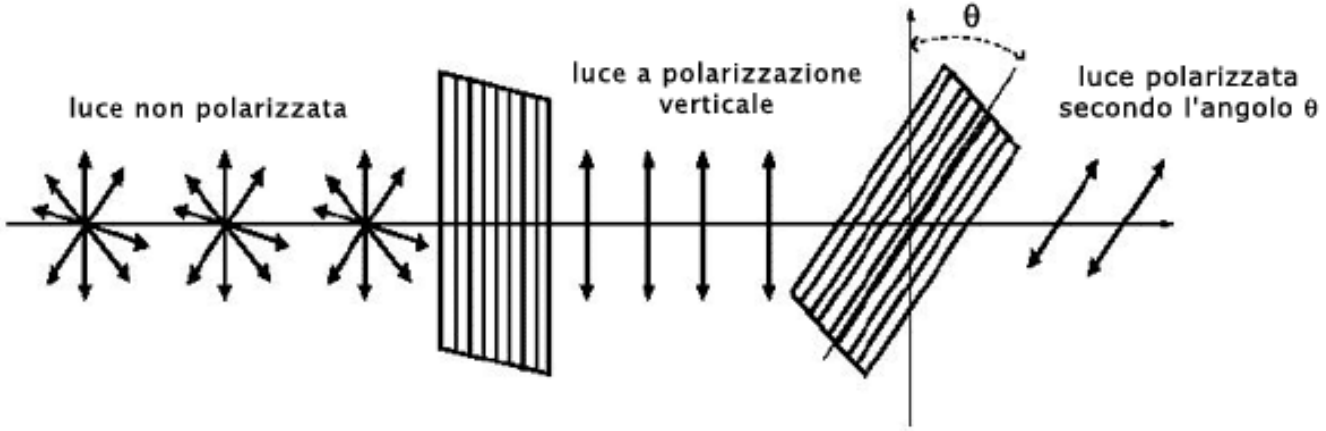
- Mentre nella crittografia classica si utilizzano tecniche matematiche per garantire la privacy delle comunicazioni, nella crittografia quantistica sono le leggi della fisica a proteggere l'informazione.
- La crittografia quantistica è basata sulle leggi della meccanica quantistica ossia lo studio della fisica a livello microscopico delle particelle elementari della materia.
- Una delle leggi fondamentali della meccanica quantistica, il principio di indeterminazione di Heisenberg, ci dice che ogni misura effettuata su un sistema quantistico perturba il sistema stesso.
- La crittografia quantistica sfrutta questa proprietà per garantire una comunicazione sicura. Nessuno è in grado di intercettare un messaggio senza modificarne il contenuto!

La rivoluzione della crittografia quantistica

- La crittografia quantistica si utilizza convenzionalmente per scambiare la chiave di cifratura di due interlocutori e non il messaggio vero e proprio.
- Successivamente con la chiave di cifratura ed un algoritmo di tipo simmetrico è possibile cifrare le comunicazioni.
- Lo scambio dell'intero messaggio su un canale quantistico non protegge in sé l'informazione ma consente solo di stabilire se non ci sono intrusi in ascolto.
- Per questo è conveniente generare a caso una chiave di cifratura inviarla su di un canale di comunicazione quantistico e determinare se è stata o meno intercettata. Nel caso in cui la chiave è stata intercettata si ripete l'operazione con una nuova chiave di cifratura fino a quando la comunicazione non risulterà sicura.

La polarizzazione dei fotoni

- Si utilizza come canale quantistico un cavo a fibra ottica per il passaggio dei fotoni, ossia degli elementi costitutivi della luce.
- La luce è di natura ondulatoria ossia è una funzione d'onda con un proprio angolo di polarizzazione θ compreso fra 0° e 180°.
- Utilizzando degli opportuni filtri di polarizzazione è possibile variare l'angolo θ (θ-filter).

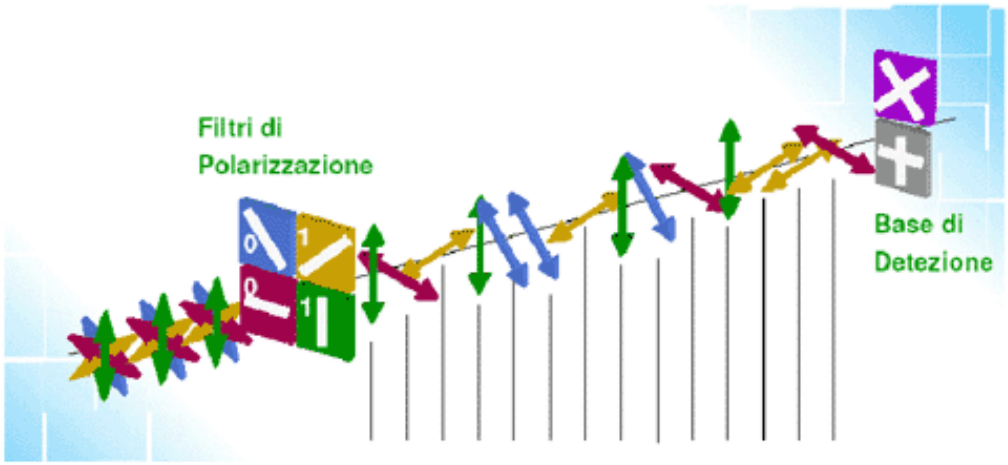


La polarizzazione dei fotoni

- Le leggi della meccanica quantistica ci dicono che un fotone a monte del filtro polarizzato con un angolo ϕ oltrepassa un θ-filter con probabilità $p_{\phi}(\theta)= \cos^2(\phi-\theta)$ emergendo ovviamente con polarizzazione θ. La probabilità che lo stesso fotone sia invece respinto dal filtro è naturalmente $1-p_{\phi}(\theta)= \text{sen}^2(\phi-\theta)$ (dal momento che $\text{sen}^2(x)+\cos^2(x)=1$).
- Consideriamo solo 4 tipi di polarizzazione, 0°, 45°, 90°, 135° e suddividiamoli in due basi ortogonali una rettilinea (0°,90°) e l'altra diagonale (45°,135°).
- Si associano questi 4 tipi di polarizzazione ai valori 0 ed 1 binari, ad esempio 0°=0, 90°=1, 45°=1, 135°=0.

La polarizzazione dei fotoni

- Con una stima probabilistica dei fotoni ricevuti è possibile determinare se la comunicazione è stata intercettata o meno.
- In accordo con le leggi della meccanica quantistica, il ricevitore può distinguere fra polarizzazioni orizzontali e verticali (0 e 90 gradi), oppure può essere riconfigurato per poter distinguere le due polarizzazioni diagonali (45 e 135 gradi). Il ricevitore non può in nessun caso distinguere polarizzazioni appartenenti a classi differenti (ad esempio 0 e 45 gradi).



Bibliografia italiana essenziale

- "Crittografia - Principi, Algoritmi, Applicazioni" di P. Ferragina e F. Luccio, Bollati Boringhieri Editore.
- "Teoria dell'informazione, codici, cifrari" di Francesco Fabris, Bollati Boringhieri Editore.
- "Crittografia" di Andrea Sgarro, Franco Muzzio Editore.
- "Segreti, Spie e Codici Cifrati" di C.Giustozzi, A.Monti, E.Zimuel, Apogeo Editore.
- "Codici & Segreti" di Simon Singh, Rizzoli Editore.
- "Crittologia" di L. Berardi, A.Beutelspacher, FrancoAngeli Editore.
- "Crypto" di Steven Levy, Shake Edizioni Underground.
- "Sicurezza dei sistemi informatici" di M.Fugini, F.Maio, P.Plebani, Apogeo Editore.
- "Crittografia e sicurezza delle reti" di William Stallings, Mc-Graw Hill Editore.

Alcuni siti Internet d'interesse

In Italiano:

- http://alpha01.dm.unito.it/personalpages/cerruti/cp0/crittoprimistart.html
- http://www.liceofoscarini.it/studenti/crittografia/index.html
- http://telemat.det.unifi.it/book/1997/cryptography/
- http://www.ecn.org/kriptonite/
- http://www.enricozimuel.net/
  http://www.esng.dibe.unige.it/Students/Courses/ei/Files/CQ1.pdf

In Inglese:

- http://www.philzimmermann.com
- http://www.rsasecurity.com/
- http://theory.lcs.mit.edu/~rivest/
- http://www.schneier.com/
- http://www.esat.kuleuven.ac.be/~rijmen/rijndael/
- http://www.crypto.com/