DENTRO LA SCATOLA

Rubrica a cura di

Fabio A. Schreiber

Con questo numero si conclude il ciclo di articoli "Dentro la scatola". Scopo dichiarato della rubrica era di presentare, in modo semplice e succinto, ma rigoroso, i concetti basilari dell'Informatica a coloro che, a vario titolo, di Informatica si occupano o con l'Informatica lavorano, pur senza avere una formazione tecnica specifica.
Il tema scelto per il 2004 è stato: "*Perché gli anglofoni lo chiamano "computer*", ovvero: introduzione alle aritmetiche digitali", e ha affrontato il modo di codificare l'informazione digitale. Per il 2005 il filo conduttore della serie è stato: "*Ma ce la farà veramente*? Ovvero: introduzione alla complessità computazionale e alla indecidibilità", con l'intento di guidare il lettore attraverso gli argomenti fondanti dell'Informatica e le loro implicazioni pratiche e filosofiche. La terza serie, negli anni 2006 e 2007, ha esplorato "*Come parlano i calcolatori*", ovvero i linguaggi di programmazione delle procedure e di interazione con i dati e i protocolli di comunicazione per i sistemi distribuiti. Non a caso la serie si conclude con un contributo sulla crittografia, argomento di grande attualità e di fondamentale importanza in un mondo sempre più informatizzato, nel quale la sicurezza e la privatezza dell'informazione diventano una delle principali preoccupazioni a livello tecnico e sociale per lo sviluppo di nuove applicazioni.
Colgo l'occasione per ringraziare sentitamente tutti i colleghi che hanno compiuto il non piccolo sforzo di condensare in testi sintetici e divulgativi i loro contributi e il Consiglio Direttivo dell'AICA per aver intrapreso un'iniziativa, che ci auguriamo possa avere un seguito nel futuro.

Crittografia elettronica

Luca Breveglieri, Mariagiovanna Sami

Il problema di proteggere l'informazione da sguardi indiscreti – in particolare, anche se non solamente, quando l'informazione deve essere comunicata fra due partecipanti – si è presentato millenni fa; dopo la seconda guerra mondiale, con l'avvento dei calcolatori elettronici, la scienza della protezione dell'informazione mediante crittografia ha avuto grande sviluppo, dato che i nuovi strumenti consentono la realizzazione di sistemi complessi di codifica che non sarebbero stati praticamente applicabili con tecniche manuali.
In questo articolo, dopo un cenno ai diversi aspetti della *sicurezza* (*security*) dell'informazione, si presenteranno brevemente le metodologie principali per la *confidenzialità* ottenuta mediante codifica (e relativa decodifica), dal momento che proprio la *confidenzialità* è alla base di tutte le soluzioni di sicurezza. Si introdurranno le cifre di merito che portano a valutazioni delle diverse metodologie, e si indicheranno i metodi oggi più significativi miranti a infrangere la sicurezza.

1. Introduzione

Si può definire la *crittologia* (*cryptology*) come lo studio della messa in sicurezza dell'informazione, vale a dire come l'analisi e il progetto, da un lato, di metodi per proteggere l'informazione contro utilizzi indesiderabili e, in senso opposto, di metodi per infrangere (o "rompere") una tale protezione [1]. La *crittografia* (*cryptography*) [2, 3] raggruppa pertanto i metodi destinati alla protezione dell'informazione, mentre il suo contrario è la *crittanalisi* (*cryptanalysis*) [2, 3]: queste due discipline costituiscono complessivamente la crittologia. Qui l'attenzione sarà posta principalmente su metodologie e algoritmi (o "primitive crittografiche") in uso nella crittografia contemporanea, la quale è basata in modo essenziale sull'uso del calcolatore elettronico; prima di entrare in argomento si farà però un breve cenno storico.
La proprietà essenziale che si desidera conferire all'informazione è la *sicurezza* (*security*). In realtà tale termine naturale è un po' vago e va

precisato caso per caso specificando quali aspetti di sicurezza si vogliono considerare. L'esempio forse più immediato e intuitivo è dato dalla proprietà di *confidenzialità* (*confidentiality*), vale a dire la protezione dell'informazione, sia essa un messaggio da trasmettere o un dato memorizzato, contro l'interpretazione e di conseguenza lo sfruttamento da parte di un'entità, essere umano o macchina, non esplicitamente autorizzata a svolgere tale funzione.

L'operazione fondamentale per ottenere la confidenzialità è la *cifratura* (*encryption*), mediante la quale si altera e confonde la forma codificata dell'informazione al fine di renderla non interpretabile; l'operazione inversa, mediante la quale si ricostruisce la forma originale, si chiama *decifrazione* (*decryption*)[1]. L'informazione in forma *non cifrata* e *cifrata* si chiama, rispettivamente, *testo in chiaro* (*cleartext* o *plaintext*) e *testo cifrato* (*ciphertext*). Le operazioni di cifratura e decifrazione si effettuano mediante *algoritmi* (o *primitive*) crittografici, e spesso all'insieme dei due algoritmi – di cifratura e di decifrazione – si dà nome di *crittosistema* (*cryptosystem* o *ciphersystem*).

La confidenzialità è tuttavia solo la prima e più semplice proprietà di sicurezza dell'informazione; ce ne sono altre più raffinate e specifiche, che si vedranno più avanti. È pur vero che la proprietà di confidenzialità funziona in molti casi come base per ottenere le altre proprietà più sofisticate.

La crittografia contemporanea è interamente basata sull'uso del calcolatore elettronico. Tralasciando dunque interpretazioni cinematografiche fantasiose, non è neppure lontanamente pensabile che la cifratura di un blocco di informazione, e meno che mai la decifrazione, svolta mediante un algoritmo crittografico contemporaneo di uso commerciale (come DES o AES), sia effettuabile da parte di un essere umano a mente o al massimo solo con l'ausilio di carta, matita e poco altro. Non è tuttavia sempre stato così, e la crittografia per secoli fu applicata manualmente o tramite qualche ausilio meccanico piuttosto semplice.

Ecco una brevissima sintesi storica [6], relativa a sistemi crittografici miranti a garantire la funzione di confidenzialità per messaggi da spedire. I primi algoritmi di cifratura di cui si abbia notizia, risalenti all'antichità, furono di tipo alfabetico a schema di sostituzione prefissato: i caratteri del testo in chiaro venivano sostituiti secondo uno schema fisso e naturalmente invertibile. Assai noto è l'algoritmo di Cesare, dove la lettera "A" viene sostituita da "C", "B" da "D", ..., "X" da "Z", "Y" da "A" e infine "Z" da "B". Naturalmente tale schema è invertibile, per potere decifrare.

Lo schema di sostituzione prefissato costituisce un esempio elementare di *chiave segreta o privata di cifratura* (*secret* o *private key*). Nell'algoritmo di Cesare la chiave è data da due elementi: la specifica del tipo di schema e il numero di passi. In generale la chiave sarà costituita da tutti gli elementi e i parametri caratterizzanti lo schema.

La proprietà di confidenzialità è garantita dal possesso esclusivo della chiave, che deve essere nota solo a chi o che cosa sia autorizzato a interpretare il contenuto del messaggio. La chiave segreta può restare in uso per sempre, oppure (per maggior sicurezza) la si può mantenere per un certo intervallo di tempo o per un dato numero di operazioni di cifratura, cambiandola poi con frequenza proporzionata al livello di confidenzialità che si desidera conseguire.

Lo scopo della crittanalisi consiste nel trovare la chiave segreta in uso, ovvero lo schema di sostituzione, avendo a disposizione come materiale di lavoro qualche esempio di testo cifrato e possibilmente anche qualche esempio di testo in chiaro corrispondente. Ingenuamente, si può sempre tentare la crittanalisi mediante *forza bruta* (*brute force*) provando tutte le chiavi segrete. Ne segue che per garantire un buon livello di protezione alla cifratura occorre garantire che l'insieme di chiavi ammissibili, o *spazio delle chiavi* (*key space*), sia adeguatamente ampio, sì da rendere impossibile o almeno non conveniente il ricorso a forza bruta.

Gli algoritmi crittografici a schema di sostituzione prefissato, alfabetici o più realisticamente a blocchi di caratteri, sono però vulnerabili all'*analisi delle frequenze* dei caratteri (*frequency analysis*). Il principio secondo cui analizzando le frequenze relative dei caratteri (o dei blocchi di caratteri) nel testo cifrato e confrontandole con quelle caratteristiche del testo in chiaro, possedendone un esemplare oppure facendo qualche ipotesi ragionevole su come esso sia fatto, si possa ritrovare lo schema di sostituzione (la chiave segreta), risale al tardo Medioevo

[1] Qualcuno purtoppo dice e scrive *crittare* e *decrittare*, *crittazione* e *decrittazione*.

e al Rinascimento; tale scoperta è accreditata a Leon Battista Alberti [6]. Ovviamente per potere adottare la *frequency analysis* occorre conoscere la lingua naturale in cui il testo è stato scritto ed è necessario che il testo cifrato sia sufficientemente lungo da consentire l'applicazione di sia pur semplici ragionamenti statistici. Un esempio letterario molto celebre è presentato in una delle avventure di Sherlock Holmes, "I pupazzi ballerini", dove a ogni lettera è sostituito un disegno schematico.

A fronte dello sviluppo delle tecniche di analisi delle frequenze vennero sviluppati metodi combinati di cifratura mediante sostituzione e permutazione di elementi di informazione (caratteri o blocchi), secondo schemi guidati da una chiave segreta. Infatti, applicando sostituzioni differenziate si riduce l'efficacia dell'analisi delle frequenze, e intercalando permutazioni (cioè cambiamenti di ordine) diverse si tendono a cancellare le regolarità ripetitive eventualmente presenti nel testo in chiaro. I crittosistemi di tale tipo in uso in età moderna furono numerosi, spesso incentrati su un repertorio prefissato ma abbastanza ampio di tabelle di sostituzione e permutazione, di uso manuale, da scegliere caso per caso come specificato dalla chiave segreta [6]. In séguito l'operazione di cifratura manuale venne accelerata mediante l'uso di sistemi elettromeccanici, quando tale tecnologia fu inventata, senza però cambiare il principio del metodo.

Il culmine di tale tendenza fu raggiunto dalla notissima macchina elettromeccanica Enigma progettata poco prima e posta in uso durante la seconda guerra mondiale [6]. Essa sostituisce e permuta i caratteri del testo secondo schemi complicati selezionati dalla chiave segreta. Di fatto l'analisi delle frequenze di testi cifrati da Enigma risultava molto complessa con i metodi dell'epoca, e fattibile solo in casi speciali e fortunati.

Con la fine della guerra e la contemporanea invenzione del calcolatore elettronico inizia la storia della crittografia contemporanea, dove cifratura e decifrazione sono realizzate da strumenti elettronici. Il primo crittosistema, ideato secondo criteri che aspiravano a essere sistematici, documentato (almeno parzialmente) e realizzato sia in software sia in hardware, che risultò di rilevante successo commerciale, fu DES (*Data Encryption Standard*), progettato da IBM nel 1973-74 [2] e reso standard da NIST nel 1976. Con esso inizia ufficialmente la storia della crittografia contemporanea. Oggi DES è superato da altre soluzioni più moderne, come AES (*Advanced Encryption Standard* - 2002) [2] [5], ma con esso si può concludere la nostra breve rassegna storica.

Nel resto dell'articolo si illustreranno sinteticamente i concetti e i metodi fondamentali della crittografia contemporanea basata su calcolatore elettronico: prima se ne presenteranno concetti e modelli di base (sezione 2), poi si descriveranno gli algoritmi commerciali attualmente in uso, a chiave segreta (sezione 3) e pubblica (sezione 4). Le conclusioni tratteggeranno alcune tendenze di ricerca attuali (sezione 5).

2. Concetti e modelli di base

La crittografia può garantire diverse proprietà di sicurezza dell'informazione. Come già detto, la proprietà fondamentale (e più semplice) è la *confidenzialità*, in pratica la restrizione della facoltà di interpretazione e sfruttamento dell'informazione solo a entità specifiche, proprietà che si ottiene tramite la funzione di *cifratura*. La proprietà di *autenticazione* (*authentication*) consiste nella garanzia che l'informazione sia prodotta o riconosciuta da un'entità identificabile, e si ottiene mediante la funzione di *firma digitale* (*digital signature*), che si vedrà più avanti. La firma digitale può anche garantire la proprietà di *integrità* (*integrity*) dell'informazione, che consiste nella garanzia che l'informazione non sia alterata, accidentalmente o intenzionalmente, e anche la proprietà di *non-ripudio* (*non-repudiation*), che consiste nella garanzia che l'informazione prodotta o riconosciuta da un'entità non possa essere da questa misconosciuta in séguito. Infine la proprietà di *autorizzazione* (*authorization*) consiste nella garanzia che l'informazione (o il servizio) sia accessibile solo da parte di un'entità o di un gruppo esplicitamente autorizzato a tale funzione, e si ottiene tramite combinazioni delle funzioni citate in precedenza.

Ogni crittosistema è dotato di uno o più elementi di informazione che costituiscono la *chiave* (*key*), da cui dipende il livello di sicurezza del sistema. In definitiva la chiave è sempre rappresentabile come stringa di bit. I crittosistemi odierni sono progettati in conformità al cosiddetto princìpio di Kerchoff (formulato già nel 1883): *tutti gli aspetti del crittosistema sono pubblici* (struttura, algoritmi e parametri), *tranne la chiave*. L'idea è che un crittosistema con

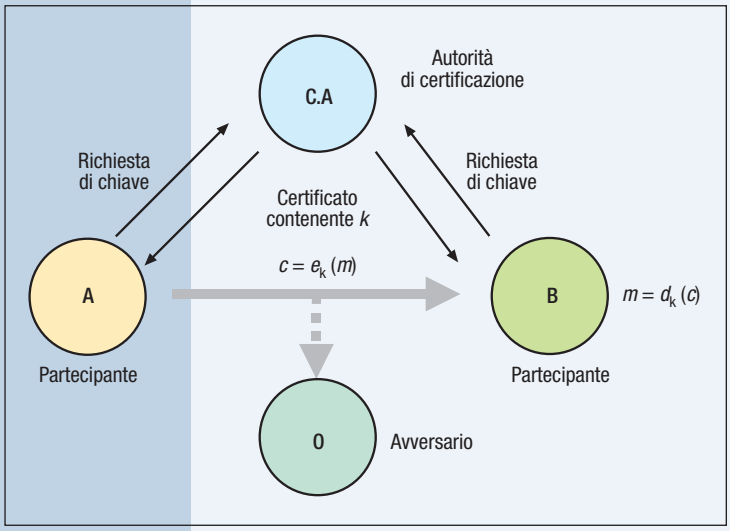

FIGURA 1

Modello generale di crittosistema (canale di comunicazione cifrato)

troppi aspetti segreti finisca per essere meno sicuro di uno con pochi. Infatti, poiché la crittografia oggi è diffusissima e primitive crittografiche si trovano realizzate in numerosi dispositivi elettronici, fissi e mobili, prodotti e distribuiti in milioni di esemplari, sarebbe del tutto vano illudersi di riuscire a celare, anche solo per breve tempo, gli aspetti strutturali e algoritmici o la configurazione di un crittosistema di successo commerciale. Senza contare che rendendo pubblico il crittosistema numerosi ricercatori possono studiarlo ed eventualmente scoprire per tempo difetti sfuggiti a una prima analisi. Il massimo che si riesca a fare, e d'altra parte il minimo indispensabile, è tenere segreta la chiave o parte di essa. Pertanto il principio di Kerchoff rappresenta un requisito formale del tutto coerente con l'uso concreto che oggi si fa della crittografia.

La figura 1 mostra il modello generale di crittosistema, nel caso l'informazione da rendere confidenziale sia un messaggio. Le entità A e B sono i *partecipanti* (*partner*) del sistema. Ecco (1) il modello espresso in simboli:

$$c = e_k(m) \quad \text{e} \quad m = d_k(c)$$
$$\text{dove} \quad m, c, k \in M, C, K \qquad (1)$$

Gli elementi *m*, *c* e *k* sono il testo in chiaro (messaggio originale), il testo cifrato (messaggio cifrato) e la chiave, rispettivamente, e appartengono agli insiemi o spazi *M*, *C* e *K* del testo in chiaro, cifrato e della chiave, rispettivamente. Di norma il testo in chiaro e quello cifrato (*m* e *c*) hanno dimensione (in bit) prefissata. I simboli e_k e d_k sono gli algoritmi di cifratura e decifrazione, rispettivamente. Li si suole formulare come funzioni a un solo argomento parametrizzate dalla chiave *k*, e naturalmente si ha l'identità $\forall\, m, k\ d_k[e_k(m)] = m$.

L'entità *O* rappresenta l'*avversario* (*adversary*), che si procura materiale vario, generalmente esemplari di testo cifrato ed eventualmente anche in chiaro, su cui lavorare per effettuare la crittanalisi e trovare la chiave *k* o almeno parte del testo in chiaro *m*.

La chiave *k* può essere la medesima per entrambi i partecipanti o differenziarsi tra i due. Nel primo caso si parla di crittosistema a chiave *segreta* (o *privata*) o *simmetrico*, nel secondo a chiave *pubblica* o *asimmetrico*. Nelle sezioni 3 e 4 se ne vedranno due esempi caratteristici. Per ora si prosegue supponendo unica la chiave, ma quanto si dirà vale anche per chiavi differenziate.

È necessario garantire che la chiave *k* sia nota solo ai partecipanti A e B, e che sia valida, cioè effettivamente legata alla coppia AB e non ad altri partecipanti. Insomma, si deve stabilire un legame sicuro tra chiave e identità del partecipante. È questo il problema di *distribuzione della chiave* (*key distribution*). La soluzione generale è data dal *certificato* (*certificate*): c'è una *terza parte fidata* (*trusted third party*), intrinsecamente sicura, chiamata *autorità di certificazione* o CA (*Certification Authority*), che rilascia la chiave dietro richiesta del partecipante. La CA mantiene una base di dati dove registra i partecipanti afferenti e le rispettive chiavi. I partecipanti possono iscriversi alla CA, essere registrati e associati a una chiave, oppure ritirarsi (o essere espulsi) rilasciando la chiave.

Per esempio, il partecipante A richiede alla CA la chiave di A per la coppia AB, e la CA risponde inviando ad A un certificato (messaggio) contenente la chiave in questione; il partecipante B procede in modo simile. Naturalmente la CA deve essere un'entità la cui sicurezza è presupposta e indiscussa; spesso è un'autorità pubblica o legalmente riconosciuta. Il certificato deve essere autentico e integro, e pertanto molto spesso viene protetto con firma digitale, come spiegato prima.

Si è riconosciuto che per ottenere la proprietà di confidenzialità (e di riflesso anche le altre) la funzione di cifratura deve possedere le due caratteristiche seguenti: *confusione* e *diffusione*. Confondere significa modificare le frequenze

relative dei simboli (caratteri, numeri, bit o blocchi) nel testo in chiaro in dipendenza dalla chiave, sì vanificare la crittanalisi mediante frequency analysis. Diffondere significa permutare i simboli (come prima) secondo la chiave, sì da cancellare le regolarità ripetitive eventualmente presenti nel testo in chiaro.
Numerosissimi algoritmi di cifratura, in particolare di tipo simmetrico, sono costruiti secondo una "ricetta" comune: suddividono il testo in blocchi di simboli, hanno struttura iterativa e il corpo del ciclo applica al blocco una successione, fissa o dipendente dall'iterazione, di trasformazioni costituite da sostituzioni e permutazioni di simboli pilotate dalla chiave, rispettivamente per confondere e diffondere il testo; in ingresso c'è il testo in chiaro e in uscita quello cifrato. Il crittosistema AES esposto nella sezione 3 ne è un'esemplificazione per così dire "da manuale".
Nella crittografia odierna le trasformazioni applicate al testo sono quasi sempre di tipo matematico ben definito e sono generalmente formulate nell'ambito della teoria dei *campi* (o degli *anelli*) *finiti*, una branca dell'algebra. Il riquadro dà una sintesi di tale teoria.

Campo Finito [4]. Il *campo finito* o di *Galois*[1] (*finite* o *Galois field*) costituisce la *struttura algebrica* dove sono ambientati tutti o quasi i crittosistemi attuali. Un campo finito F è un *insieme finito* di elementi a, b, c, ..., dotato di due *operazioni binarie* tra elementi: addizione $a + b$ e moltiplicazione $a \times b$, associative, commutative, dotate di elemento neutro e opposto o reciproco. Il campo finito può ridursi a un *anello finito*, cui manca solo la possibilità di definire il reciproco di ciascun elemento. Alcuni crittosistemi (come per esempio RSA) sono ambientati in anelli finiti.

[1] Dal nome dello scopritore, il matematico francese Evariste Galois.

Ci sono motivazioni precise che hanno contribuito a orientare la crittografia verso la matematica dei campi finiti. Ecco le due principali:
❑ L'insieme di elementi è finito ed essi sono pertanto rappresentabili in modo esatto.
❑ Le operazioni di addizione e moltiplicazione del campo modellano a grandi linee le trasformazioni di sostituzione e permutazione, e hanno dunque effetto confusivo e diffusivo.
Un esempio di campo è dato dagli interi con addizione e moltiplicazione modulo un numero primo $p > 1$. Per esempio, $2 + 4 \bmod 5$ = resto di 6 rispetto a $5 = 1$, e $2 \times 4 \bmod 5$ = resto di 8 rispetto a $5 = 3$. Esistono anche campi finiti i cui elementi sono polinomi. Insomma, il campo finito offre un ambiente matematico sistematico e ricco di proprietà utili dove costruire primitive crittografiche.
Si può quantificare il *livello di sicurezza* (di confidenzialità ecc.) del crittosistema in tre categorie generali. Il parametro caratterizzante la dimensione del crittosistema è il numero di bit $n \geq 1$ necessari per codificare in binario la chiave (lunghezza di chiave). Ecco le tre categorie di livello:
❑ *assoluto*: non esiste in linea di principio nessun metodo di crittanalisi;
❑ *computazionale*: si dimostra matematicamente che l'algoritmo di crittanalisi computazionalmente più efficiente è NP-completo;
❑ *pratico*: l'algoritmo di crittanalisi computazionalmente più efficiente noto al momento ha costo economicamente non conveniente rispetto al valore dell'informazione in chiaro ottenibile.
Il livello di sicurezza assoluto è conseguibile unicamente usando la chiave una volta sola per cifrare e sostituendola continuamente. Dato che non c'è riuso della chiave, la crittanalisi è impossibile in via teorica. I crittosistemi di questo tipo sono detti *usa-e-getta* (*one-pad*).
È molto arduo costruire un crittosistema collocato a livello di sicurezza computazionale, a causa della difficoltà matematica di dare una dimostrazione rigorosa di NP-completezza. Gli esempi noti di crittosistemi la cui analisi è stata dimostrata essere NP-completa sono pochi [1] e nessuno è di interesse commerciale rilevante. Di fatto quasi tutti i crittosistemi in uso commerciale si collocano a livello di sicurezza soltanto pratico.
Si conclude con un breve accenno alla crittanalisi. In generale essa ha lo scopo di trovare il testo in chiaro, partendo dal testo cifrato, senza conoscere la chiave. In pratica comunque mira quasi sempre a trovare proprio la chiave, perché una volta questa è nota qualunque esemplare di testo cifrato è decifrabile. Il metodo ingenuo e universale di crittanalisi è la f*orza bruta*; la *frequency analysis* è un metodo meno generico ma pure universale[2], più o meno efficace. I metodi di crittanalisi *lineare* e *differenziale* sono molto generali e consistono nell'approssimare il crittosistema con funzioni lineari; sono tuttavia meto-

[2] Tranne che nel caso usa-e-getta.

di complessi, che richiedono parecchi aggiustamenti *ad hoc* dipendenti dal crittosistema in esame [7, 8]. Oltre a questi metodi in un certo senso universali, la crittanalisi si divide in famiglie particolari di cosiddetti *attacchi* (*attack*) mirati sul crittosistema specifico, i quali ne sfruttano le debolezze matematiche o i difetti di implementazione (in SW o in HW); gli attacchi saranno ripresi brevemente nelle conclusioni (sezione 5).

3. Crittosistemi simmetrici

Un crittosistema mirato a garantire la confidenzialità dell'informazione è detto *simmetrico* o a *chiave segreta* se i due partecipanti A e B usano la stessa chiave k_s, detta appunto segreta (o privata); si rivedano la figura 2 e il modello (1).

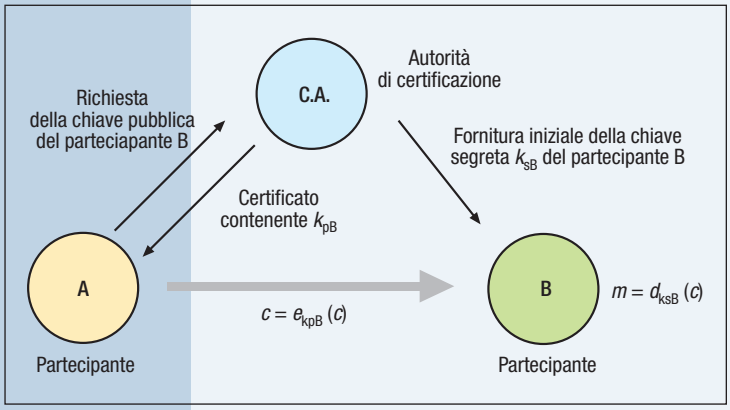

FIGURA 2
Modello di crittosistema asimmetrico (canale cifrato con chiave pubblica)

La chiave segreta k_{sAB} è dunque univoca per la coppia di partecipanti AB. Essa va calcolata inizialmente e deve essere conosciuta solo da parte di A e B; in generale può essere distribuita mediante certificazione.

Il primo crittosistema simmetrico standard di larga diffusione fu DES: ha testo in chiaro, cifrato e chiave segreta tutti di lunghezza 64 bit. Una variante, Triple-DES o 3DES, consiste nell'applicare in cascata tre cifrature DES con due chiavi differenti ed è talvolta ancora oggi utilizzata.

Il crittosistema standard oggi di maggiore diffusione è Rijndael (*Rijemen-Daemen* 2000) o AES (*Advanced Encryption Standard*) [5]. Lo standard AES è stato definito nel 2002 a séguito di un processo di selezione internazionale gestito dal NIST e durato due anni. Alla fine è stato scelto tra sei crittosistemi concorrenti e ha rapidamente rimpiazzato DES in numerose applicazioni. Ha la caratteristica importante di essere calcolabile in modo efficiente sia in software sia in hardware. Infatti lavora essenzialmente su byte, ed è efficiente anche se programmato su processori da 8 bit, di potenza limitata. È del tutto pubblico, molto studiato e ben documentato, e pertanto soddisfa al principio di Kerchoff. Per una breve presentazione del crittosistema [2, 5] si veda il riquadro.

Il crittosistema AES fornisce gli algoritmi di cifratura e decifrazione, e garantisce la proprietà

Crittosistema AES (o Rijndael) [2, 5]

Ambiente matematico: campo finito F_2^8 dei polinomi di grado max 8 (byte), a coeff. binari (0 e 1).

Testo in chiaro, cifrato e chiave segreta: stringhe *m*, *c* e *k*, ciascuna da 128 bit

Algoritmo di schedulazione della chiave (va effettuato *una-tantum*): pre-elabora la chiave segreta *k* (usando varie trasformazioni molto simili a quelle dell'algoritmo di crifatura), derivando da essa 10 chiavi di round k_r ($1 \le r \le 10$), ciascuna delle quali è una stringa di 128 bit.

Algoritmo di cifratura: suddividi il testo in chiaro *m* in 16 byte $m_{u,v}$, organizzati a matrice quadrata di tipo 4×4 ($1 \le u, v \le 4$), ed esegui il ciclo seguente (ciascuna iterazione si chiama "round")

for *r* **from** 1 **to** 10 **do**
applica a ciascun byte $m_{u,v}$ la trasformazione (non-lineare) SubByte
applica a ciascuna riga m_u della matrice la trasformazione (lineare) ShiftRow
applica a ciascuna colonna m_v della matrice la trasformazione (lineare) MixColumn
applica all'intero testo *m* la trasformazione (lineare) AddRoundKey(kr)
end for

il testo finale elaborato dai 10 round del ciclo costituisce il testo cifrato c; ciascuna delle quattro trasformazioni (o funzioni) interne al round SubByte, ShiftRow, MixColumn e AddRoundKey è biunivoca; per i particolari si veda per esempio, [5].

Algoritmo di decifrazione: il ciclo di cifratura, eseguito in ordine retroverso e con le trasformazioni SubByte, ShifRow, MixColumn e AddRoundKey sostituite dalle rispettive inverse InvSubByte, InvShiftRow, InvMixColumn, InvAddRoundKey($k_{10} - r$) (chiavi di round usate in ordine inverso).

di confidenzialià dell'informazione. Si colloca a livello di sicurezza pratico (non computazionale), poiché non è formalmente dimostrato che la crittanalisi di AES sia infattibile in tempo meno che esponenziale rispetto al numero di bit della chiave, benché di fatto oggi per infrangere AES non si sappia fare di meglio che esplorare l'intero spazio delle chiavi mediante forza bruta, ciò che implica circa $2^{128} \approx 10^{43}$ tentativi. Allo stato attuale la crittanalisi di AES è del tutto infattibile con i supercalcolatori più potenti disponibili.
Ci sono altri crittosistemi simmetrici concorrenti di AES, più o meno diffusi. Tra quelli più noti si trovano RC4 e RC5, molto semplici e talvolta usati per esempio in reti di sensori con potenza di calcolo limitatissima e alcuni altri, ma non molti.

4. Crittosistemi asimmetrici

Una comunità di membri che usano a coppie un crittosistema simmetrico deve gestire in generale tante chiavi segrete quante sono le coppie possibili, e la distribuzione delle chiavi risulta pertanto onerosa e rischiosa. Un crittosistema mirante a garantire la confidenzialità dell'informazione è detto *asimmetrico* o a *chiave pubblica* se i due partecipanti A e B usano chiavi differenti; si veda la figura 1. Ecco (2) il modello espresso in simboli (questo particolarizza il modello (1)):

$$c = e_{ks}(m) \quad \text{e} \quad m = d_{kp}(c)$$
$$\text{dove } m, c, k_s, k_p \in M, C, K_s, K_p \qquad (2)$$

Ciascun partecipante (qui A e B) è dotato di due chiavi: segreta k_s e pubblica k_p. Per cifrare il messaggio per B, A si serve della chiave pubblica di B, k_{pB}; per decifrare B si serve della propria chiave segreta, k_{sB}; lo stesso accade se B deve cifrare per A. La chiave pubblica di ciascun partecipante può essere resa nota a tutti gli altri partecipanti, mentre quella segreta è conosciuta solo dal partecipante cui si riferisce. Ne segue che la comunità di partecipanti usa molte meno chiavi che nel caso simmetrico.
Tuttavia, dato che deve valere l'identità $\forall m \in M : d_{kp}[e_{ks}(m)] = m$, le due chiavi di una coppia segreta-pubblica di un partecipante sono legate funzionalmente. Affinché il crittosistema sia sicuro deve essere computazionalmente infattibile trovare la chiave segreta conoscendo quella pubblica (poiché allora tutti potrebbero farlo e le chiavi segrete cesserebbero di essere tali!). Il proprietario della chiave segreta conosce dall'inizio la sua chiave e non ha mai bisogno di calcolarla.
Non esiste, o quanto meno oggi non si conosce, nessuna "ricetta" per definire crittosistemi asimmetrici. Di conseguenza ci sono a tutt'oggi ben pochi esempi di crittosistemi asimmetrici ed essi differiscono profondamente uno dall'altro.
Il crittosistema asimmetrico principale e più diffuso è RSA (*Rivest-Shamir-Adelman*), ideato nel 1977. Per una breve presentazione del crittosistema [2, 3] si veda il riquadro. Essenzialmente, RSA fornisce gli algoritmi di cifratura e decifrazione, e garantisce la proprietà di confidenzialià dell'informazione. Fornisce però anche l'algoritmo di firma digitale e garantisce le proprietà di autenticazione, integrità e non-ripudio. Si basa su un teorema di teoria dei numeri (si veda il riquadro); qui ci si

Crittosistema RSA [2]

Ambiente matematico: anello Z_n degli interi modulo $n > 1$, dove l'intero $n = pq$ (composto) è il prodotto di due fattori primi differenti $p, q > 1$ rappresentabili con circa il medesimo numero di bit.

Testo in chiaro e cifrato: numeri interi m e c $(0 \le m, c < n)$.

Fase di preparazione (va effettuata *una-tantum* per impostare il sistema):

- calcola la funzione di Eulero $\varphi(n) = (p-1)(q-1)$, dove $1 < \varphi(n) < n$
- scegli (a caso) un intero $a > 1$ [e $< \varphi(n)$] coprimo con $\varphi(n)$, cioè tale che il massimo comun divisore di a e $\varphi(n)$ sia 1, ovvero tale che m.c.d.$[a, \varphi(n)] = 1$
- calcola l'intero $b > 1$ reciproco di a modulo $\varphi(n)$, cioè tale che $ab = 1 \bmod \varphi(n)$ (b esiste ed è unico se e solo se m.c.d.$[a, \varphi(n)] = 1$, e si calcola facilmente)
- ***chiave pubblica e segreta***: la coppia e la terna di interi $k_p = (a, n)$ e $k_s = (b, p, q)$

Algoritmo di cifratura: calcola l'esponenziale $c = m^a \bmod n$

Algoritmo di decifrazione: calcola l'esponenziale $m = c^b \bmod n$

Dimostrazione di correttezza del sistema: dato l'intero a e la condizione m.c.d.$[a, \varphi(n)] = 1$, vale il *teorema fondamentale* seguente:

$\forall m$ si ha $[(m^a)^b] = [(m^b)^a] = m \bmod n$

pertanto l'algoritmo di decifrazione è effettivamente l'*inverso* di quello di cifratura (e viceversa).

Valutazione del livello di sicurezza del sistema: per trovare l'intero b (segreto), indispensabile per decifrare, partendo dall'intero a (pubblico), *bisogna conoscere* $\varphi(n) = (p-1)(q-1)$, e dunque i fattori primi p e q di n; però *scomporre* l'intero n in fattori primi ha complessità temporale esponenziale con l'algoritmo migliore noto (Crivello di Eratostene, con svariate ottimizzazioni); se ne conclude che il crittosistema RSA si colloca a *livello di sicurezza pratico* (non computazionale).

limita a enunciarlo, si vedano [2] o [3] per la dimostrazione.
RSA non si colloca a livello di sicurezza computazionale perché non esiste una dimostrazione formale che il problema della scomposizione in fattori primi sia NP-completo. In pratica oggi è possibile scomporre in fattori primi numeri rappresentabili con 600-700 bit, tramite mezzi di supercalcolo molto potenti. La lunghezza di chiave standard per RSA (il numero di bit per codificare *n* in binario) è fissata a 1024 bit, più che adeguata per applicazioni civili, e quella per applicazioni militari a 2048 bit, ben oltre ogni immaginabile possibilità di decifrazione.
Il crittosistema RSA è assai versatile e realizza facilmente la funzione di *firma digitale* (*digital signature*) per autenticazione, integrità e non-ripudio. Ecco come procedere a questo scopo.

- Il partecipante A *firma* il testo *m* cifrandolo mediante la funzione e_{ksA} e la chiave segreta k_{sA} (non quella pubblica), cioè con l'esponente *b* (non *a*): la coppia (*m*, *c*) costituisce il *testo firmato* (*m* è il testo e *c* la firma).
- Il partecipante B *verifica* la firma prendendo il testo firmato (*m*, *c*), decifrando la firma *c* mediante la funzione d_{kpA} e la chiave pubblica k_{pA} (non quella privata), cioè con l'esponente *a* (non *b*), e confrontando il risultato della decifrazione con *m* (cioè con il testo in chiaro):
  - se coincidono la firma c è valida e il testo *m* è autentico e integro;
  - altrimenti non è valida e il testo *m* è da rifiutare in quanto falso o alterato.

Dato che la chiave segreta è in possesso solo del partecipante che firma il testo, questo non può misconoscere la firma in un secondo tempo. Pertanto anche la proprietà di *non-ripudio* del testo firmato è garantita. La funzione di firma digitale è ampiamente utilizzata per autenticare il certificato e garantirne l'integrità. L'insieme di regole standard che definiscono struttura, metodo di firma e modo di gestione del certificato è noto come *infrastruttura di chiave pubblica* o PKI (*Public Key Infrastructure*), ed è un componente molto importante dei sistemi crittografici applicativi.

5. Conclusioni

Vale la pena di concludere illustrando le tendenze di ricerca attuali, di stampo scientifico cioè mirate a scoperte nuove, e ingegneristico cioè mirate alla realizzazione. L'ideazione e messa punto di crittosistemi originali, simmetrici e asimmetrici, è un campo di ricerca scientifica sempre aperto, dove si mira a individuare nuovi crittosistemi con costo computazionale decrescente e livello di sicurezza crescente, senza però ampliare troppo lo spazio delle chiavi, anzi se possibile restringendolo. In questa direzione i crittosistemi asimmetrici *basati su curve ellittiche* [10, 11] sono di grande interesse perché ammettono numerose varianti algoritmiche, non tutte ancora identificate e ben studiate. La *crittografia quantistica* (*quantum cryptography*) è un approccio promettente [12] ma richiede a tutt'oggi sforzi di ricerca notevoli sia metodologici sia tecnologici. C'è infine una mole consistente di ricerca ingegneristica mirante a ottimizzare l'implementazione di crittosistemi già noti e in uso corrente, per evidenti ragioni di tipo applicativo e industriale.
In senso opposto, anche la crittanalisi è oggetto di ricerca, realizzativa e metodologica. Il primo approccio fa leva sull'aumento costante della potenza dei mezzi di supercalcolo. Invece i metodi universali di crittanalisi sono estremamente difficili da ottimizzare e i progressi in tale campo sono graduali e avvengono con una certa lentezza. Ben più fiorente, anzi oggi dominante, è lo studio dei metodi di *attacco*, cioè dei metodi di crittanalisi mirati sul crittosistema specifico. Qui l'interesse maggiore è diretto verso il campo vastissimo dei cosiddetti *attacchi collaterali* (*side-channel attack*), che sfruttano le debolezze dell'*implementazione* del crittosistema. In ambito industriale gli attacchi collaterali sono ben più temuti di ogni altra minaccia alla sicurezza di dispositivi elettronici.
Lo scopo dell'attacco collaterale è di raccogliere informazione rilasciata in modo inavvertito o imprevisto, sotto forma di grandezza misurabile o quantità valutabile, da parte del dispositivo crittografico, e di elaborarla successivamente per effettuare la crittanalisi e in definitiva trovare il testo in chiaro o meglio ancora la chiave. In ordine di tempo sono state scoperte tre famiglie principali di attacco di tipo collaterale, con numerose ramificazioni più o meno fini:

1. l'attacco *in tempo* (*timing attack*);

2. l'attacco *in potenza* (*power attack*);
3. l'attacco *basato* su *induzione di guasto* (*fault-injection-based attack*).

Nei primi due casi si misura una grandezza fisica: tempo di calcolo e potenza elettrica consumata o elettromagnetica irradiata, rispettivamente; nel terzo si valuta una quantità, ossia il risultato errato emesso in caso di guasto indotto intenzionalmente. Capita spesso che queste grandezze e quantità siano correlate in modo determinabile al testo in chiaro o alla chiave, e che pertanto rilevandole e poi analizzandole con metodi di carattere essenzialmente statistico, si possano inferire l'uno o l'altra. Per misurare con precisione il tempo di calcolo basta rilevare il segnale di clock del circuito, per la potenza consumata o irradiata basta ricorrere a oscilloscopio o altro strumento similare, e per indurre intenzionalmente un guasto (temporaneo) ben localizzato e sincronizzato si può ricorrere per esempio alla tecnologia laser (light-based fault attack).

Infine, giova ricordare che naturalmente si sviluppano in continuazione applicazioni dotate di funzionalità di sicurezza basata anche su crittografia, sia nel campo delle comunicazioni sia in quello delle basi di dati, e non di rado un'applicazione sicura tiene conto di entrambi gli aspetti.

Bibliografia

[1] Simmons G.: *Contemporary Cryptology*. IEEE Press, 1992.

[2] Menezes A., van Oorschot P.C., Vanstone S.: *Handbook of Applied Cryptography*. CRC Press, 1997.

[3] Stinson D.: *Cryptography: Theory and Practice*. CRC Press, 1995.

[4] Lidl R., Niederreiter H.: *Finite Fields*. 2ª edizione, Cambridge, University Press, 1997.

[5] FIPS 197: *Announcing the Avdanced Encryption Standard (AES)*. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf, 2001.

[6] Singh S.: *Codici e Segreti*. Rizzoli, Milano, 1999.

[7] Matsui M., *Linear Cryptanalysis Method for the DES Cipher*. Advances in Cryptology - EUROCRYPT '93, LCNS n. 765, Springer-Verlag, 1994, p. 386-397.

[8] Biham E., Shamir A.: Differential Cryptanalysis of DES-like Cryptosystems. *Journal of Cryptology*, Vol. 4, n. 1, 1991, p. 3-72.

[9] Thomas S.: *SSL and TLS Essentials securing the Web*. New York, John Wiley & Son, 2000.

[10] Menezes A.: *Elliptic Curve Public Key Cryptosystems*. Kluwer Academic Publishers, 1993.

[11] Boneh D., Franklin M.: Identity Based Encryption from the Weil Pairing (IBE). *SIAM Journal of Computing*, Vol. 32, n. 3, 2003, p. 586-615.

[12] Bennett C., Bessette F., Brassard G., Salvail L., Smolin J.: Experimental Quantum Cryptography. *Journal of Cryptology*, Vol. 5, n.1, 1992, p. 3-28.

Come detto nel riquadro iniziale, questa è l'ultima puntata di "Dentro la scatola".
Mondo Digitale coglie l'occasione per porgere un caldo "Grazie!"al prof. Fabio Schreiber che ha proposto la rubrica, ideato la sequenza degli articoli e seguito passo passo l'implementazione.

Mariagiovanna Sami è professore ordinario di prima fascia presso il Politecnico di Milano, nell'area dei Sistemi di Elaborazione. Dal 1987 al 1990 è stata Direttore del Dipartimento di Elettronica del Politecnico di Milano. È Direttore Scientifico del Master of Science in Embedded Systems Design presso l'Università della Svizzera Italiana a Lugano.
La sua attività di ricerca riguarda le architetture hardware dei sistemi digitali, con particolare riguardo alle metodologie di progetto di sistemi dedicati caratterizzati da elevate prestazioni, robustezza e basso consumo di potenza. È autrice o co-autrice di oltre duecento lavori scientifici in sede internazionale ed ha ricevuto alcuni premi per la sua attività di ricerca. È membro dell'Accademia Italiana delle Scienze (detta dei Quaranta).
E-mail: sami@elet.polimi.it

Luca Breveglieri, Nato nel 1962. Laureato in ingegneria elettronica nel 1986 e dottore di ricerca in ingegneria elettronica dell'informazione e dei sistemi nel 1991. Dal 1991 al 1998 è stato collaboratore tecnico (per la rete telematica) presso il Politecnico di Milano. Dal 1998 è professore associato di ingegneria informatica presso il Politecnico di Milano. Ha svolto attività di ricerca nel campo della progettazione digitale e delle architetture di elaborazione, in particolare per aritmetica del calcolatore ed elaborazione di segnale e immagine. I suoi intessi di ricerca attuali comprendono algoritmi e dispositivi per crittografia, e aspetti della teoria dei linguaggi formali.
E-mail: luca.breveglieri@polimi.it