UNIVERSITÀ DEGLI STUDI "ROMA TRE"
FACOLTÀ DI INGEGNERIA
CORSO DI STUDI IN INGEGNERIA INFORMATICA

Elementi di crittografia - A.A. 2004/2005

Crittografia e Sistemi Bancomat

A cura di:

BERNARDI PAOLO (matr. 249594)
BONACCORSO MICHELE (matr. 249157)
ZAMPONI ROBERTO (matr. 249669)

Gli **ATM** (*Automatic Teller Machine*), meglio conosciuti in Italia come "sportelli Bancomat", offrono servizi bancari rapidi per i possessori di conti correnti.

Generalmente le misure di sicurezza adottate consistono nel riconoscimento dell'utente mediante scheda magnetica e codice *PIN.* La sicurezza di questi sistemi può essere però violata in vari modi che dipendono dalla struttura Hardware/Software utilizzata e dagli algoritmi crittografici impiegati nella generazione dei codici PIN.

Questo documento esaminerà le problematiche legate alla sicurezza del servizio bancomat nel suo insieme, analizzando i vari aspetti sopra citati, e introdurrà il lettore alle nuove tecnologie che si stanno diffondendo in questo campo, in cui la crittografia (nel senso di algoritmi crittografici) è il fulcro per il conseguimento della sicurezza. Verrà inoltre analizzato una tipologia di attacco che è possibile effettuare agli ATM sfruttando proprio l'algoritmo di generazione del codice PIN.

1 – Introduzione

A cura di Paolo Bernardi

1.1 – Il funzionamento

Il servizio Bancomat, offerto dalle singole banche, è realizzato mediante l'integrazione di due o più apparati hardware e da un'architettura software basata sui servizi offerti da un modulo chiamato HSM o MSS che offre funzionalità crittografiche specifiche.

Se però un utente vuole prelevare del denaro da uno sportello di un'altra banca la verifica del PIN non avviene in locale ma viene inoltrata alla banca detentrice del conto corrente, attraversando uno o più *switch* nella rete EFTPOS (Electronic Funds Transfer at the Point of Sale).

Gli ATM

Il compito degli Auto-Teller-Machines è quello di fornire un'interfaccia con l'utente, pertanto prevedono periferiche di I/O come monitor, tastiera, lettore di schede magnetiche e/o smartcard, un dispositivo per fornire il denaro richiesto e una telecamera.

Gli ATM generalmente utilizzavano un sistema operativo della IBM e comunicavano con l'HSM mediante un protocollo di rete proprietario IBM chiamato SDLC. Negli ultimi anni però le banche hanno richiesto la migrazione al sistema operativo Windows Xp per le sua interfaccia "user friendly" e l'utilizzo del classico protocollo di comunicazione TCP/IP.

HSM e MSS

Gli HSM (Hardware Security Module) e gli MSS (Modular Security System) sono componenti dedicati in grado di offrire un ristretto numero di API (Application Program Interface) per la generazione e validazione di codici PIN secondo gli standard utilizzati dalle diverse compagnie quali VISA, MasterCard, ecc. Questi critto-calcolatori dedicati sono in grado di utilizzare vari algoritmi crittografici quali il DES, 3DES, RSA e SHA1. Essendo moduli hardware dedicati, hanno elevate prestazioni crittografiche e riescono ad esempio a validare fino a 60 PIN al secondo.

Negli ultimi anni la crittografia è divenuta indispensabile per garantire la sicurezza informatica e questa è chiaramente una delle prime necessità delle banche che, ormai sempre di più, offrono servizi anche su internet. Per soddisfare tutte queste differenti tipologie di richieste crittografiche, vengono prodotti diversi tipi di MSS a seconda del mercato target: Alcuni modelli di MSS sono in effetti dei veri e propri server crittografici (HTTPS / COM / WSDL / Java / C / .NET).

Switch

La necessità di poter verificare un PIN di un'altra banca viene risolta introducendo le chiavi di zona (Zone Master Keys): Ogni banca conosce i propri PIN e possiede una chiave di zona per cifrarli / decifrarli. Il compito degli switch è quello di collegare le diverse banche nella rete ETFPOS e quello di tradurre PIN cifrati con chiavi differenti.

Casi d’uso del sistema

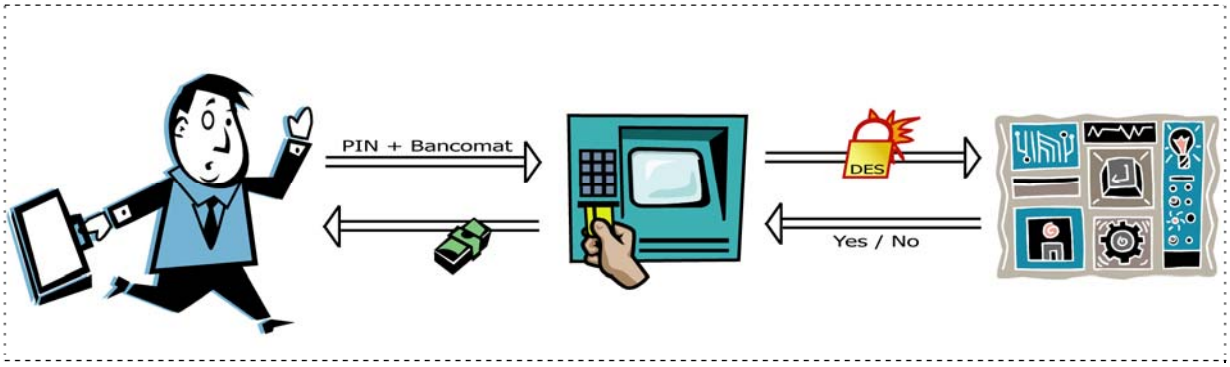

Figura 1 - Prelievo alla propria banca

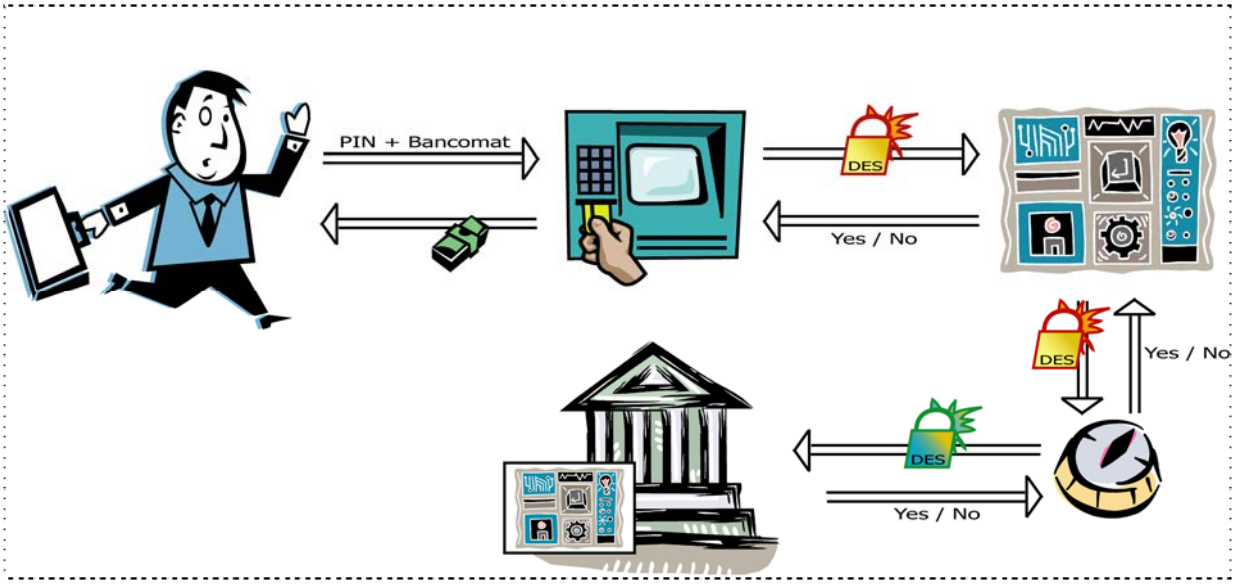

Figura 2 - Prelievo da un'altra banca

1.2 – Sicurezza del servizio bancomat

Per come è stato progettato, il servizio bancomat ha vari punti di debolezza:

- L'ATM
- La rete ETFPOS
- L'HSM

ATM

La migrazione del sistema operativo degli ATM da IBM a Windows Xp embedded ha di fatto esposto gli ATM a tutti i problemi di sicurezza che ha un normale PC di casa; infatti nei primi mesi del 2003 molti ATM della Diebold (leader nella realizzazione di ATM) sono stati colpiti dal worm "Welchia" che sfruttava delle vulnerabilità nel servizio RPC (remote procedure call) di Windows Xp. Da allora anche negli ATM è stato installato un firewall per contrastare questo tipo di situazioni.

Questo fatto di pura cronaca mette in luce un problema generale riguardo l'utilizzo di Sistemi operativi generici come windows xp in ambienti dedicati quali gli ATM: se è possibile infettare un ATM allora è anche possibile intercettare le richieste che fa al modulo HSM e se queste non sono fossero cifrate il livello di sicurezza offerto scenderebbe notevolmente.

La rete ETFPOS

Una ulteriore motivazione ad utilizzare la crittografia è legata proprio alla comunicazioni interbanca; se queste avvenissero in chiaro, un utente collegato all'interno della rete potrebbe ricavare i PIN e i relativi numeri di conti correnti con uno sniffer.

I PIN attualmente vengono cifrati con il DES o il 3DES. Considerando che il DES può essere rotto anche con algoritmi di brute force dipendenti esclusivamente dalla potenza di calcolo dei PC, l'intercettazione di un PIN è comunque pericolosa e va evitata.

HSM

Anche se il modulo HSM e la verifica del PIN fossero di per sé sicuri, un programmatore collegato alla stessa rete interna della banca potrebbe effettuare un brute force all'HSM per un conto corrente di un cliente per ricavare il PIN: per un codice di 5 cifre ci sono 10^5 possibili combinazioni, quindi considerando che l'HSM è in grado di verificare 60 PIN al secondo, nel caso peggiore si potrebbe ricavare il PIN esatto in 28 minuti circa.

La tecnica di critto-analisi presentata da Mike Bond dell'università di Cambridge si basa sull'identificazione di un PIN a 4 cifre (probabilmente in uso negli ATM della CityBank) ed il suo algoritmo permette di trovare un PIN in relativamente pochi tentativi e sfrutta proprio il metodo di generazione del PIN. Infatti l'HSM utilizza i classici algoritmi **DES** o **3DES** per la generazione e la verifica del codice PIN a partire dal numero di conto corrente memorizzato sulla banda magnetica della carta bancomat in più viene utilizzata una tabella di decimalizzazione per la conversione da esadecimale a decimale. Come vedremo in seguito, il principale punto di debolezza nell'algoritmo di verifica di un PIN è proprio l'utilizzo della tabella di decimalizzazione.

1.3 – Migrazione verso sistemi più sicuri

SmartCard

Il limite principale all'attuale tecnologia è dovuto al fatto che la banda magnetica non può conservare i dati relativi al proprietario della carta in modo sicuro. L'innovazione principale delle smart card è proprio la memorizzazione sicura dei dati personali del possessore, inoltre, la

grande quantità di memoria a disposizione permetterebbe ad una smart card di essere utilizzata in molteplici applicazioni.

Biometria

L'identificazione mediante password è un metodo di autenticazione "innaturale"… non può cioè attestare con sicurezza l'identità della persona, ma semplicemente garantire che l'utente sia a conoscenza di qualcosa o lo possegga… (il PIN e la carta nel nostro caso)

La biometria invece si basa sull'identificazione delle caratteristiche biologiche quali ad esempio l'iride, le impronte digitali o il percorso del sistema venoso e quindi è in grado di attestare la presenza fisica della persona. Alcuni esperimenti sono già stati condotti da alcune banche giapponesi.

Crittografia quantistica

Un approccio completamente differente è quello della crittografia quantistica, dove la sicurezza viene garantita dal fatto che un'eventuale intercettazione viene rilevata dal mittente e dal destinatario del messaggio. Questa tecnica viene infatti sfruttata per lo scambio delle chiavi.

2 – Decimalization Table Attack

A cura di Roberto Zamponi

Presentiamo un attacco ai moduli di sicurezza dell'hardware utilizzato per la memorizzazione e la verifica dei PIN degli utenti di un servizio di bancomat.
Utilizzando PIN composti da quattro cifre se ne hanno 10000 differenti (da 0000 a 9999). Ad un malintenzionato che voglia indovinare il PIN corretto di un cliente del servizio di bancomat facendo dei tentativi a caso, occorrerebbero quindi in media 5000 tentativi.

2.1 Sicurezza bancaria

Le banche hanno intrapreso la strada della lotta contro le frodi: sia di quelle originate all'interno dello staff bancario, che di quelle originate da estranei.
Esse hanno sviluppato metodi di protezione contro le frodi dei membri interni introducendo contabilità a duplice inserimento, separazione delle funzionalità bancarie, periodi di vacanza obbligatori per lo staff, e riconoscendo la necessità di regolari verifiche della sicurezza. Questi metodi hanno avuto successo nel ridurre le frodi ad un livello accettabile, e, in concomitanza ad una struttura legale per stabilire le responsabilità in caso di frodi, possono anche proteggere il cliente della banca che usufruisce del servizio di bancomat.
Le pratiche bancarie tradizionali puntano sì a ridurre il numero delle frodi ad un livello accettabile, ma ciò si traduce scarsamente nell'individuazione di quali dovrebbero essere i requisiti di sicurezza; così spesso è impossibile stabilire se il manifestarsi di un difetto nel sistema possa considerarsi un caso isolato o la punta di un enorme iceberg.
L'introduzione degli HSM (Hardware Security Module: modulo di sicurezza hardware) per proteggere i PIN dei clienti è stato un passo nella giusta direzione, ma fino al 2002 questi dispositivi non sono stati universalmente adottati e quelli che lo sono stati hanno dimostrato parecchie volte di non essere immuni ad attacchi, come vedremo nel paragrafo 2.3.

Scenario:

Gli sportelli bancomat sono usati ogni giorno da milioni di intestatari di conti bancari per effettuare dei prelievi di denaro.
La loro vasta diffusione e i luoghi solitari in cui a volte essi sono ubicati li rende dei mezzi ideali per i criminali.
L'autenticazione dell'utente dello sportello bancomat è la misura di sicurezza primaria contro le frodi: la clonazione della carta bancomat è un'operazione banale in confronto all'acquisizione del PIN.
I programmatori di una banca hanno accesso al sistema informatico impiegato per la memorizzazione dei PIN degli intestatari dei conti, che normalmente consiste in un mainframe connesso con un solo HSM che è tamper-resistant, cioè resistente ai tentativi più disparati di manomissione, ed ha un API (Application Programmer Interface: elenco di dati e funzioni che possono essere utilizzati dal programmatore) che per motivi di sicurezza risponde solo con un "SI" o con un "NO" alle interrogazioni.
Il sistema informatico della banca, su menzionato, può essere soggetto ad attacchi. Un esempio di attacco è quello di scrivere un programma per l'HSM che provi tutti i PIN, in forma cifrata o in chiaro, per un particolare conto bancario. Un algoritmo di ricerca esaustiva, per un PIN di quattro cifre, dovrà provare in media 5000 possibili PIN prima di trovare quello giusto, 10000 tentativi nel caso peggiore ma considerando una probabilità di successo al 50% i tentativi scendono a 5000. Normalmente un HSM permette di testare circa 60 PIN al secondo, oltre al normale traffico di richieste. Facendo dei calcoli otteniamo che 5000 interrogazioni all'archivio dei PIN possono essere soddisfatte in un tempo che si aggira intorno agli 83 secondi. Dunque, sotto queste ipotesi, per scoprire 25 PIN corretti occorreranno 83×25 secondi, ovvero circa 35 minuti.

Ad ogni modo, gli HSM, implementando diversi metodi di generazione del PIN, hanno dei difetti.
I primi ATM erano degli IBM 3624, introdotti negli Stati Uniti intorno al 1980, e molti dei metodi di generazione dei PIN sono basati ancora sul loro approccio.
Gli ATM basati sull'architettura degli IBM 3624 calcolano il PIN del cliente dal numero di conto bancario. Questo viene cifrato utilizzando il cifrario DES con una chiave segreta chiamata "chiave di generazione del PIN". La stringa ottenuta dalla cifratura DES è interpretata come una stringa esadecimale, sono necessarie le prime quattro cifre della stringa ottenuta, le altre sono scartate. Ognuna delle 4 cifre considerate è una delle seguenti: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F.
Per convertire questo quartetto di cifre esadecimali in un PIN che possa essere digitato su una tastiera decimale, è necessaria una Decimalization Table (Tabella di Decimalizzazione) che realizza un mapping many-to-one tra le cifre esadecimali e quelle decimali.
Il mapping many-to-one non è altro che una funzione suriettiva: ad ogni elemento del dominio corrisponde un solo ed unico elemento del codominio; ogni elemento del codominio è il corrispondente (l'immagine) di uno o più elementi del dominio.
Questa decimalization table si trova nell'HSM; in figura viene illustrata, a sinistra una tipica tavola di decimalizzazione e a destra una tavola che viene usata per sferrare un attacco:

0123456789ABCDEF	0123456789ABCDEF
0123456789012345	0000000100000000

Tavole di decimalizzazione tradizionale e d'attacco

La tavola di decimalizzazione non è considerata un input sensibile da molti HSM, così le API di questi prevedono l'inserimento, oltre che del numero di conto bancario e del PIN, di una tavola di decimalizzazione del tutto arbitraria. Questa è una vulnerabilità. Manipolando la tavola, ad ogni tentativo di inserimento di un PIN, è possibile apprendere molto sul valore corretto del PIN. Ad esempio, sostituiamo la decimalization table con la tavola sulla destra della figura precedente ed utilizziamo un PIN di prova 0000, interrogando l'API si può sapere se il numero 7 esista o no nel PIN originale. Vedremo successivamente meglio.

2.2 Generazione del PIN e tecniche di verifica

Esistono numerose tecniche per la generazione e la verifica del PIN.
Il nostro obiettivo è analizzare in maggiori dettagli il metodo IBM 3624-Offset in quanto caratterizzato dall'uso di tavole di decimalizzazione.

Il metodo di generazione del PIN cliente dell'IBM 3624-Offset

Il metodo IBM 3624-Offset, sviluppato per la prima generazione dei bancomat, è stato molto adoperato e imitato.
Viene adottato uno schema in cui i PIN degli utenti possono essere calcolati dal numero di conto bancario dell' utente stesso cifrandolo con una chiave segreta (mediante cifratura DES). Il numero di conto è reso disponibile su scheda magnetica, quindi la macchina adibita all'erogazione del denaro necessita solo di un modo sicuro per immagazzinare la singola chiave utilizzata per la cifratura detta "chiave di generazione del PIN".
Il numero di conto è rappresentato con cifre ASCII e può essere interpretato come un input esadecimale per il cifrario a blocchi DES.
Dopo la codifica con la "chiave di generazione del PIN" segreta, l'output è convertito in esadecimale e tutte le cifre, tranne le prime quattro, vengono scartate. Le quattro cifre esadecimali sono poi convertite in cifre decimali utilizzando il mapping secondo la Decimalization Table. Infine per permettere all'intestatario della carta bancomat di cambiare il proprio PIN, viene addizionato un offset al quartetto decimale ottenuto in precedenza (l'offset è pubblico e varia a seconda delle banche). Quando un bancomat verifica un PIN inserito, semplicemente sottrae l'offset e confronta il risultato ottenuto con il PIN naturale, ossia il

quartetto ottenuto dal processo descritto precedentemente, partendo dal numero di conto fornito dalla carta magnetica.

Esempio:

N° di conto: 4556 2385 7753 2239

-- Cifratura DES --

N° di conto cifrato: 3F7C 2201 00CA 8AB3

-- Scarto cifre non usate --

1° quartetto del N°c.c.: 3F7C

-- Mapping --

0 1 2 3 4 5 6 7 8 9 A B C D E F

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5

PIN naturale: 3572

Offset pubblico: 4344

-- PIN naturale + Offset pubblico --

PIN del cliente: 7916

API dei moduli di sicurezza hardware

I bancomat e i centri di controllo delle banche usano l'HSM (Hardware Security Module), un coprocessore che svolge servizi di crittografia e sicurezza, per proteggere le chiavi scelte per cifrare i PIN da impiegati corrotti e attaccanti. Una tipica API finanziaria ha funzioni per generare e verificare i PIN, funzioni per tradurre i PIN adottando diverse chiavi di cifratura quando questi viaggiano tra banche differenti, e altre importanti funzioni gestionali.
Tipicamente chiunque abbia accesso al computer HSM, secondo le politiche d'uso, può effettuare ogni giorno comandi come la verifica di un PIN, mentre funzioni sensibili come il settaggio di nuove chiavi di cifratura, possono essere effettuate solo con l'autorizzazione da parte di più impiegati fidati.
Nella figura viene riportato un prototipo di funzione per la verifica del codice PIN di prova.
La funzione prende in input i seguenti parametri: il `PAN_data`, ovvero il numero di conto, la `Tavola di Decimalizzazione` e l`'encrypted_PIN_block`. I primi due dati vengono immessi in chiaro e possono essere manipolati con relativa facilità da un attaccante; l'ultimo dato, un PIN cifrato, che rappresenta un PIN di prova scelto per l'attacco, è più difficile da ottenere.

```
Encrypted_PIN_Verify(
    A_RETRES , A_ED ,                 // return codes 0,0=yes 4,19=no
    trial_pin_kek_in , pinver_key ,   // encryption keys for enc inputs
    (UCHAR*)"3624    " "NONE    "     // PIN block format
    "       F"                        // PIN block pad digit
    (UCHAR*)"            " ,
    trial_pin ,                       // encrypted_PIN_block
    I_LONG(2) ,
    (UCHAR*)"IBM-PIN0" "PADDIGIT" ,   // PIN verification method
    I_LONG(4) ,                       // # of PIN digits = 4
    "0123456789012345"                // decimalisation table
    "123456789012    "                // PAN_data (account number)
    "0000            "                // offset data
                    );
```

Codice del prototipo della funzione per la verifica di un PIN di prova cifrato.

Oltre ai parametri di input, su citati, tutti gli altri dati presenti nella funzione "Encrypted_PIN_Verify" dipendono dal sistema informatico della banca, quindi non dovrebbero interessare un ipotetico attaccante. Come si può notare anche dai commenti in figura, questi dati sono le chiavi di cifratura, il formato del PIN_block, il tipo del metodo di verifica da usare, il numero di cifre che compongono il codice PIN e l'offset.
Gli altri valori sono delle costanti (A_RETRES e A_ED) rappresentanti gli outputs della funzione, ossia i codici 0 o 4,19 corrispondenti rispettivamente a una risposta affermativa e negativa alla richiesta di verifica.

Come notato, il PIN dato in input alla funzione di verifica è in forma cifrata; quindi a un attaccante non è possibile inserire il PIN di prova in chiaro.
Alcuni HSM, tuttavia, permettono l'inserimento in chiaro dei PIN di prova. Questa funzionalità può essere richiesta per inserire PIN di prova casuali quando si vogliono generare PIN cifrati per gli schemi che non utilizzano tabelle di decimalizzazione. L'appropriato comando è Clear_PIN_Encrypt, che, dal PIN in chiaro inserito, genera un PIN cifrato. Si noti che abilitando questo comando si corrono altri rischi oltre a quello di permettere l'attacco con tavole di decimalizzazione. Se non viene effettuato il padding randomizzato dei PIN prima che essi vengano cifrati, un attaccante può costruirsi un elenco di PIN di prova cifrati noti, confrontare ogni PIN cifrato prelevato dai pacchetti in transito verso l'HSM con quelli dell'elenco e determinarne il valore.

2.3 Attacco alle Tavole di Decimalizzazione

Lo schema di base di un attacco a una tavola di decimalizzazione è diviso in due fasi. Nella prima fase vengono determinate quali cifre sono presenti nel PIN da ricercare. Nella seconda fase sono testati tutti i PIN composti con le cifre identificate.
Sia D**orig** la tavola di decimalizzazione iniziale.
Per una data cifra decimale i, consideriamo una tavola di decimalizzazione binaria D**i** in cui compare 1 alla posizione x se e solo se D**orig** ha i in quella posizione.
In altre parole:

$$D_i[x] = \begin{cases} 1 & \text{if } D_{\text{orig}}[x] = i, \\ 0 & \text{otherwise.} \end{cases}$$

Per esempio, per una tavola standard D**orig** = 0123 4567 8901 2345, utilizzando la formula in figura, il valore di D_3 è 0001 0000 0000 0100.
Nella prima fase, per ogni cifra i, testiamo il PIN originale con una tavola di decimalizzazione D**i** e un PIN di prova 0000. E' semplice constatare che il test fallisce quando il PIN originale contiene la cifra i. Così, utilizzando al massimo dieci tentativi, determiniamo quali sono le cifre del PIN originale.
Nella seconda fase, proviamo tutte le possibili combinazioni di queste cifre. Il loro numero dipende da quante cifre differenti il PIN contiene.

CIFRE	POSSIBILITA'
A	AAAA(1)
AB	ABBB(4), AABB(6), AAAB(4)
ABC	AABC(12),ABBC(12),ABCC(12)
ABCD	ABCD(24)

Guardando la tabella:

nella prima riga AAAA(1) indica che esiste una sola combinazione quando abbiamo un' unica cifra;
nella seconda riga AABB(6) indica che esistono sei combinazioni quando abbiamo due cifre diverse che compaiono due volte ognuna; e cosi di seguito...
La tabella mostra che la seconda fase richiede al più 36 tentativi (quando il PIN originale contiene 3 cifre differenti), con un totale di 46 tentativi tra le due fasi, nel caso peggiore.

3 – Evoluzione nei sistemi di autenticazione per il Bancomat

A cura di Michele Bonaccorso

3.1 Le Smart Card

Iniziamo con la discussione sulle smartcard. Nell'attuale panorama internazionale l'incremento dell'uso, nonché dell'interesse, per le smart card può essere interpretato come uno dei fondamentali cambiamenti nell'industria dei pagamenti elettronici.

La soluzione della banca magnetica, introdotta ormai più di 30 anni fa, nonostante alcuni aggiornamenti tecnici intervenuti in questo lasso di tempo, oggi non è ritenuta più sicura ed adeguata ai bisogni crescenti di sicurezza e funzionalità.

Nonostante l'evoluzione dei nuovi sistemi di sicurezza, il rischio principale risiede nella limitata sicurezza che la banda magnetica può offrire nel custodire i dati relativi al proprietario della carta. In questo panorama si inserisce l'avvento delle smart card che, grazie alle loro potenzialità di immagazzinare dati in modo sicuro per poi utilizzarli nel corso della transazione, possono essere utilizzate per accedere a più servizi, e per tale ragione si parla di smart card "multi-applicazione".

Nel 1993 dalla collaborazione dei principali circuiti di pagamento a livello mondiale (Europay, Mastercard, Visa), in conseguenza dei sempre più frequenti casi di frodi e falsificazioni, è stato realizzato lo standard EMV basato sulla piattaforma di lavoro (EMVCo), per lo sviluppo delle specifiche legate al pagamento elettronico basato su smart card.

Tale standard definisce: i requisiti di carattere fisico ed elettrico (EMV Level 1), gli aspetti applicativi, ovvero le modalità con cui devono essere condotte le transazioni (EMV Level 2), la struttura delle carte dal punto di vista della sicurezza, l'interoperabilità tra le carte e i terminali a livello globale, le linee guida ed i tempi per la migrazione dalla banda magnetica ai nuovi sistemi. L'EMV, quindi, stabilisce le regole che permettono alla smart card e al terminale di pagamento di interagire tra loro. Queste sono basate sull' ISO 7816, una serie di standard per le carte a circuito integrato e per i dispositivi di accettazione.

L'EMV definisce i requisiti minimi di sicurezza per le smart card, ma lascia libertà ai circuiti finanziari di stabilire ulteriori parametri, il che ha portato allo sviluppo di differenti applicazioni finanziarie a seconda del circuito: Visa implementa "Visa Smart Debit Credit", Mastercard implementa "**M/Chip", tutti sistemi compatibili con lo standard EMV ma con parametri di gestione del rischio e transazioni offline differenti.**

L'EMV definisce quattro elementi principali per la sicurezza delle applicazioni finanziarie delle carte a microprocessore:

- *Autenticazione della carta offline:* il terminale POS deve identificare la carta come genuina.
- *Parametri di gestione del rischio*: la carta registra ogni transazione e manda un allarme in caso si verifichino certe condizioni.
- *Offline-Pin*: le smart card possono conservare i dati in modo sicuro, e ciò permette che la verifica del PIN avvenga internamente alla carta stessa.
- *Autenticazione della carta online*.

L'EMV non specifica gli algoritmi crittografici che devono essere usati nell'autenticazione, ma definisce un elemento di 8 bit chiamato "**Application Cryptogram**" che contiene in modo sicuro i dettagli di ogni transazione. Per quanto riguarda l'autenticazione, possiamo brevemente osservare i "parametri" cui solitamente si fa riferimento. Le carte a banda magnetica,

contengono un valore di verifica (CVV) che può essere verificato solo durante una transazione online. Le smart card al contrario possono essere autenticate sia offline che online utilizzando due differenti tecniche. In particolare si distinguono i seguenti parametri:

- *SDA, o Static Data Autentication:* è la più semplice e meno costosa, in questo processo la carta viene identificata dal terminale attraverso l'uso della stessa firma digitale per ogni transazione.
- *DDA, o Dynamic Data Autentication:* crea una firma digitale diversa per ogni transazione offline. Questa tecnologia è più sicura ma più costa rispetto all'altra.
- *CDA, o Combined Dynamic Data Autentication:* la carta genera l' Application Cryptogram e la firma digitale dinamica; il terminale, verificando la firma digitale, è in grado di determinare che l'Application Cryptogram sia generato da una carta genuina.

3.2 Crittografia quantistica

Passiamo ad analizzare il secondo metodo, ovvero quello della crittografia quantistica applicata al caso delle transazioni finanziarie. Il principale punto debole di ogni comunicazione cifrata, secondo la fisica classica, è che un intercettatore che abbia accesso al canale può sempre trascrivere il testo cifrato che viene inviato sul canale stesso. Con l'avvento della crittografia quantistica la disputa tra algoritmo e crittoanalisi si risolverà in favore dei crittografi, grazie alla natura intrinseca della tecnologia basata sulla fisica degli stati quantistici, che la rende teoricamente impossibile da violare e persino da intercettare.

Le leggi della fisica quantistica, applicate alla trasmissione sicura, garantiscono che l'operazione di intercettazione venga rilevata dalle parti attive sul canale. Questo conferisce a tutti gli schemi di crittografia quantistica la garanzia di perfetta sicurezza, e la certezza di non-intercettazione rappresenta il salto in avanti fondamentale di queste nuove tecnologie rispetto a diversi millenni di crittografia classica.

Questi sistemi quantistici sfruttano il principio di indeterminazione di Heisenberg, secondo il quale la misurazione di un sistema quantistico in genere lo perturba e fornisce un'informazione incompleta sul suo stato precedente alla misurazione.

La crittografia quantistica si utilizza convenzionalmente per scambiare fra mittente e destinatario le chiavi private e non il messaggio. Il messaggio verrà crittato successivamente attraverso l'algoritmo One Time Pad. In questo senso si dice che la crittografia quantistica si utilizza nella distribuzione delle chiavi. La Quantum Key Distribution (QKD) permette a due soggetti di ottenere chiavi sicure attraverso l'invio di fotoni su un "canale quantistico". Nello schema fondamentale di crittografia quantistica l'informazione associata al singolo fotone risiede nella sua polarizzazione: il fotone viene inviato con una precisa polarizzazione, che corrisponde in uno schema prestabilito ad una cifra binaria (0 e 1).

Le leggi della fisica impediscono ad un terzo soggetto di acquisire informazione sullo stato di un fotone senza disturbarlo, ovvero modificarlo irreparabilmente. E' praticamente impossibile intercettare con profitto (ovvero, senza essere scoperti) uno scambio di chiavi su un canale quantistico senza essere a conoscenza degli schemi di polarizzazione adottati. Inoltre ogni tentativo di intercettazione può essere rilevato con una semplice verifica che rivelerà un apparente incremento di errori di trasmissione.

La meccanica quantistica, diversamente dalla meccanica classica, considera un fascio di luce composto da quantità discrete di energia chiamate *fotoni*, le quali, data la natura ondulatoria della luce, hanno un proprio angolo di polarizzazione, che è definito come l'angolo formato dal piano in cui essi oscillano con l'asse di propagazione degli stessi fotoni. L'angolo di propagazione è un numero θ compreso fra 0° e 180°: non ci sono infatti distinzioni fra un fotone polarizzato a θ ed uno polarizzato a θ +180°.

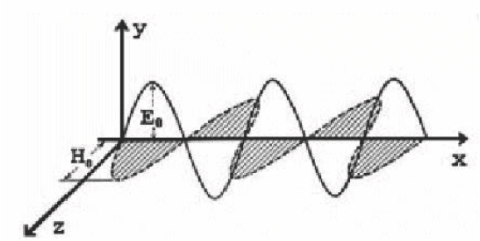

Figura 3 – Rappresentazione su assi cartesiani dell'andamento dei fotoni

I fotoni provengono da una sorgente di luce con una polarizzazione arbitraria (sono possibili tutte le polarizzazioni). Per far assumere una particolare polarizzazione ad un fotone si utilizza un filtro polarizzatore, che risponde ad una serie di proprietà. 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:

$$p_\theta(\phi) = \cos^2(\phi - \theta)$$

emergendo ovviamente con polarizzazione θ. La probabilità che lo stesso fotone sia invece "respinto" dal filtro è naturalmente:

$$1 - p_\theta(\phi) = \sin^2(\phi - \theta)$$

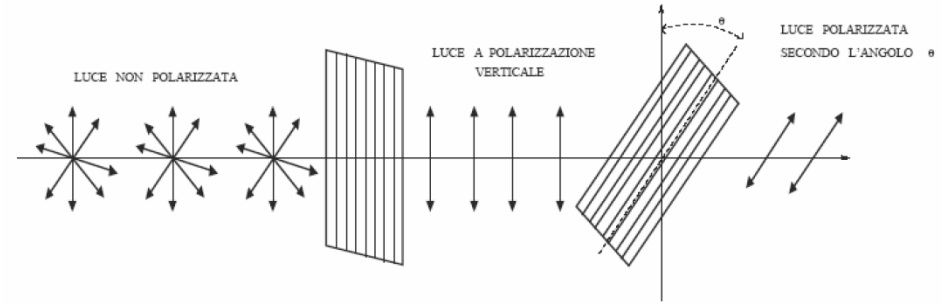

Figura 4 – Rappresentazione della composizione del sistema per l'impiego dei fotoni

A riguardo la trattazione potrebbe essere ulteriormente approfondita, ma ritengo che quanto detto sia sufficiente allo scopo di introdurre tale tecnologia nell'ambito della trattazione specifica fin qui portata avanti.

Recentemente in una banca Austriaca è stata effettuata una transizione elettronica di denaro usando fotoni *"entangled"* (correlati quantisticamente) per creare un codice di comunicazione indecifrabile. Anche se esistono già prodotti commerciali basati sulla crittografia quantistica, nessuno di questi utilizza fotoni correlati per garantire una comunicazione sicura. I fotoni "entangled" obbediscono ai principi della meccanica quantistica: disturbando lo stato di uno, si disturba automaticamente anche l'altro, non importa a che distanza si trovino. La coppia di fotoni correlati utilizzata era stata generata inviando un laser attraverso un cristallo per dividere singoli fotoni in due. Un fotone di ogni coppia correlata è stato poi inviato dalla banca al municipio attraverso una fibra ottica. Giunti a destinazione, è stato osservato il loro stato di polarizzazione. In questo modo entrambe le estremità del collegamento avevano a disposizione lo stesso dato (un 1 oppure uno 0). In questo modo è stato possibile costruire una chiave crittografica con la quale proteggere da terzi la transazione finanziaria.

Nel settembre 2004 è stato finanziato dal "*Ministero dell'Università e della Ricerca Scientifica*" un progetto del dipartimento di Fisica dell'Ateneo di Camerino teso a realizzare un Bancomat supersicuro con la crittografia quantistica.

3.3 Biometria

Il metodo che probabilmente sarà di più rapida introduzione, per il momento in appoggio agli altri sistemi esistenti, e successivamente a supporto di altre tecnologie attualmente in fase di studio e sviluppo, è certamente quello che prevede l'utilizzo di sistemi biometrici. Questi sono dispositivi automatici per la verifica di identità o identificazione di una persona sulla base di caratteristiche biologiche, che possono essere di varia natura e sono generalmente suddivise, come mostrato nella figura seguente, in *fisiologiche* (più affidabili: impronta digitale, volto, mano, retina, iride, ...) e *comportamentali* (più semplici da integrare in alcune specifiche applicazioni: voce, calligrafia, stile di battitura, ...).

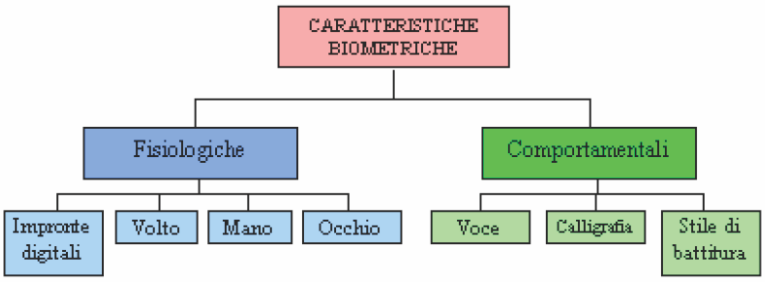

Figura 5 – Schematizzazione delle caratteristiche biometriche

Le prime sperimentazioni che sono state condotte sono riconducibili a:

- **Texas, 1999**: primo esperimento biometrico in ambito bancario; una banca ha utilizzato come tecnologia di riconoscimento la *scansione dell'iride*.
- **Tokyo, autunno 2004**: introdotto da Bank of Tokyo-Mitsubishi; sportelli ATM polifunzionali dotati di tecnologie di identificazione e autenticazione biometriche. La piattaforma utilizzata consente di riconoscere i clienti dal *percorso del sistema venoso* delle loro mani; il percorso delle vene di ogni individuo è unico e praticamente impossibile da duplicare o clonare.

Volendo accennare brevemente all'architettura di un sistema biometrico, la prima cosa che occorre notare è che questo può essere generalmente impiegato per la verifica di identità o per l'identificazione:

- *il problema della verifica di identità*: consiste nello stabilire se un individuo è veramente colui che dichiara di essere; a tal fine l'utente deve fornire al sistema, oltre alla caratteristica biometrica da esaminare, anche il proprio nome o un codice di identificazione personale che rappresenta la sua dichiarazione di identità.

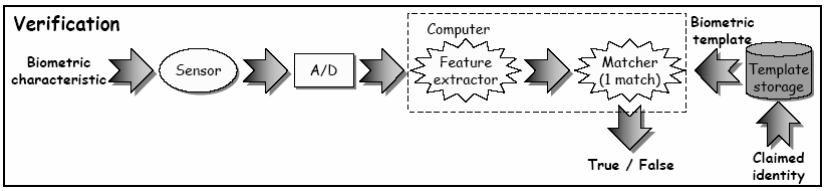

Figura 6 – Schematizzazione del problema della verifica

- *il problema dell'identificazione*: consiste invece nel determinare se una persona può essere associata (corrisponde) a una di quelle presenti in un archivio (non è richiesto all'individuo di dichiarare la propria identità).

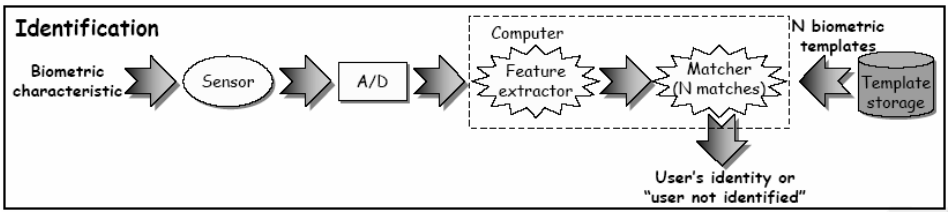

Figura 7 - Schematizzazione del problema dell'identificazione

I sistemi biometrici, per poter espletare in maniera corretta il loro funzionamento, hanno bisogno di una fase di registrazione iniziale, detta fase di "*enrolment*", durante la quale vengono acquisite una o più istanze della caratteristica biometrica.

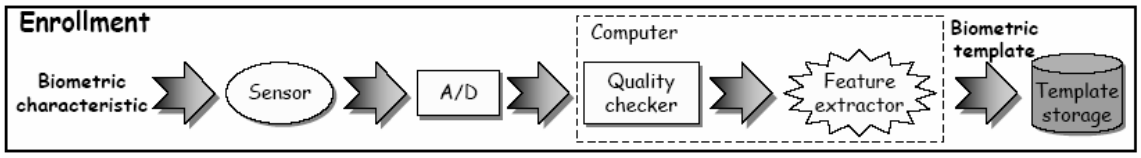

Figura 8 - Schematizzazione della fase di "enrollment"

Diversamente da un'operazione di controllo di una password, l'affidabilità del risultato di un confronto di istanze diverse delle stessa caratteristica biometrica non è del 100%. Le principali cause delle differenze tra acquisizioni successive della stessa caratteristica sono

- *variazioni sopravvenute nella caratteristica biometrica* (ad esempio tagli o escoriazioni nelle impronte digitali);
- *errato posizionamento rispetto al sensore di acquisizione;*
- *salienti modificazioni dell'ambiente di acquisizione* (variazioni di illuminazione, temperatura, umidità,...).

Non si può quindi dire che due istanze di una caratteristica biometrica coincidano, ma al più si può affermare che due istanze sono sufficientemente simili; il sistema deve essere cosciente di questo fatto, ma siccome non c'è una certezza assoluta dell'uguaglianza tra due istanze, si possono verificare errori.

Due sono i tipi di errore che un sistema biometrico può commettere; la probabilità di tali errori è espressa da due parametri (legati tra loro) che prendono il nome di FRR e FAR:

- *FRR (False Rejection Rate: frequenza di falsi rifiuti)*: specifica la frequenza con la quale il sistema rifiuta ingiustamente individui che sono autorizzati all'accesso. Nel caso in cui un utente venga ingiustamente rifiutato dovrà ripresentare nuovamente la caratteristica biometrica al sistema. Un falso rifiuto non è necessariamente indice di errore del sistema: si pensi ad esempio, nel caso delle impronte digitali, ad un incorretto posizionamento del dito sul sensore o alla presenza di sporcizia.
- *FAR (False Acceptance Rate: frequenza di false accettazioni):* specifica la frequenza con cui il sistema è ingannato da estranei che riescono a essere autorizzati, pur non avendo diritto di accesso. Questo tipo di errore è sicuramente più grave.

Il grado di sicurezza di un sistema biometrico può essere impostato dall'amministratore agendo sulla *soglia di sicurezza t*, un parametro che stabilisce quanto stringenti debbano essere i requisiti di somiglianza delle caratteristiche biometriche. FRR e FAR sono infatti funzione della soglia t:

- incrementando il valore di t per rendere più arduo il compito agli impostori (diminuisce FAR), alcuni utenti, che lecitamente tentano di accedere al sistema, possono essere talvolta rifiutati (cresce FRR).
- al contrario, diminuendo il valore di *t* per facilitare estremamente gli accessi a chi ne ha diritto (diminuisce FRR) potrebbe aumentare il pericolo di false accettazioni (cresce FAR).

Riportiamo di seguito un grafico in cui si può vedere l'andamento dei due parametri FRR e FAR in funzione della tolleranza *t*:

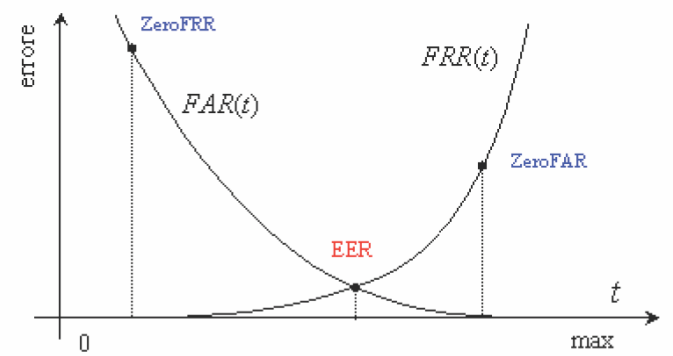

Figura 9 – Andamento di FRR e FAR in relazione alla tolleranza

Nel grafico sono stati evidenziati tre punti notevoli, denominati ZeroFRR, ZeroFAR e EER; questi tre parametri costituiscono un'alternativa per specificare le prestazioni di un sistema biometrico.

Il loro significato è il seguente:

- *EER (Equal Error Rate):* indica l'errore del sistema nel punto in cui FRR = FAR;
- *ZeroFAR*: indica FRR nel punto in cui FAR= 0;
- *ZeroFRR*: indica FAR nel punto in cui FRR= 0.

E' possibile classificare i sistemi biometrici in cooperativo/non cooperativo, evidente/velato (se il sistema è nascosto al pubblico o no), abituato/non abituato (se viene utilizzato spesso o meno dallo stesso utente), frequentato/non frequentato (se viene utilizzato spesse volte nell'arco della giornata anche se da utenti diversi), pubblico/privato (se è accessibile da un numero elevato di utenti diversi), ambiente standard, aperto/chiuso (se vi è o meno scambio di informazioni verso l'esterno).
Oltre a classificare i sistemi biometrici, si può effettuare un'ulteriore qualificazione delle caratteristiche biometriche, rilevando alcuni parametri comuni di confronto:

- *universalità*: se ogni individuo possiede o meno una determinata caratteristica;
- *unicità*: il grado con cui si può trovare la stessa caratteristica tra due soggetti diversi;
- *permanenza*: se la caratteristica varia o meno nel tempo;
- *misurabilità*: se le caratteristiche possono essere misurate quantitativamente;
- *esecuzione*: il raggiungimento preciso dell'identificazione;
- *accettabilità*: il grado con cui una persona è disposta ad utilizzare un determinato sistema biometrico;
- *insidia*: il grado di "ingannabilità" del sistema utilizzando una determinata caratteristica biologica.

Riportiamo di seguito una tabella che evidenzia il valore stimato di tali parametri per alcune caratteristiche biometriche:

	universalità	*unicità*	*permanenza*	*misurabilità*	*esecuzione*	*accettabilità*	*insidia*
Impronta	Medio	Alto	Alto	Medio	Alto	Medio	Alto
Retina	Alto	Alto	Medio	Basso	Alto	Basso	Alto
Volto	Alto	Basso	Medio	Alto	Basso	Alto	Basso
Mano	Basso	Basso	Basso	Alto	Basso	Alto	Basso
Firma	Basso	Basso	Basso	Alto	Basso	Alto	Basso
Voce	Medio	Basso	Basso	Medio	Basso	Alto	Basso

Dopo numerosi studi di caso specifici è stato dimostrato come molti di questi sistemi siano vulnerabili a diversi tipi di attacchi che hanno portato all'aumento di frodi ai danni degli sportelli

ATM. E' importante osservare che un sensore, capace di leggere e identificare i dati biometrici, è uno dei modi migliori per aumentare la sicurezza. Nei nuovi ATM è incorporato un lettore (uno scanner ad infrarossi) che "legge la mano" del cliente senza che questi sia costretto a toccare alcunché: una volta eseguita l'operazione potrà cominciare la transazione.

Per concludere la trattazione a riguardo dei sistemi biometrici, in relazione a quanto detto finora, viene spontaneo chiedersi se tale sistema sia realmente inattaccabile. In realtà molti di questi sistemi sono vulnerabili a diversi tipi di attacchi. Particolarmente efficaci risultano gli attacchi noti come *replay attack*: l'hacker, introducendosi nel sistema informatico, ruba una copia dell'immagine digitalizzata della caratteristica biometrica e se ne serve per "proiettarla" in un'altra occasione. Lo stesso si può dire per gli attacchi consistenti nella manipolazione del valore di soglia di ciascun sistema, aumentando il valore del FAR e rendendo più facili gli accessi da parte di intrusi. Ulteriori ricerche su questo tema sono state condotte anche da alcuni gruppi di studiosi dell'IBM, che hanno evidenziato i limiti e le debolezze di questi sistemi; i ricercatori hanno individuato diverse modalità di attacco possibili, tra cui la più sofisticata è quella che prevede l'inserimento nel sistema di un "cavallo di Troia" che provvederebbe a passare dati errarti al sistema informatico, in modo tale da non poter riconoscere neanche "l'utente onesto". Il punto di forza delle caratteristiche biometriche, cioè l'immutabilità nel corso del tempo, rappresenta paradossalmente un grosso limite e una forte debolezza nel caso di attacchi da parte di hacker.