UNIVERSITA' DEGLI STUDI DI
NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base
Corso di Laurea in Ingegneria Informatica

Elaborato finale in **Calcolatori elettronici I**

Smart Card e token crittografici

Anno Accademico 2018/2019

Candidato:

Valerio Volpe

matr. N46003053

A tutti coloro che mi hanno sostenuto in questo percorso, in particolare alle mie nonne che non vedono l'ora di vedermi laureato

Indice

Introduzione

Quello della sicurezza informatica è un settore molto importante e in continua evoluzione.

La crittografia gioca un ruolo fondamentale in questo campo in quanto la trasmissione dei dati viene spesso effettuata tramite canali poco sicuri e facilmente intercettabili.

I pilastri sui quali si basa la sicurezza, ovvero le problematiche che si cerca di risolvere tramite la crittografia, sono quattro:

- **Riservatezza** di un messaggio, ovvero la garanzia che un messaggio possa essere letto solo dal destinatario, evitando che possa essere intercettato e letto da entità non desiderate.
- **Integrità** del contenuto. Il messaggio inviato deve arrivare al destinatario senza subire modifiche o manipolazione da parte di terzi.
- **Autenticazione** della persona con la quale si sta comunicando. Si vuole garantire che un'entità sia effettivamente ciò che dice di essere.
- **Disponibilità dei servizi**. Questa problematica riguarda in particolare le aziende che hanno server accessibili pubblicamente e che quindi possono essere soggetti di attacchi DoS (*Denial of Service*)

Per far fronte a queste problematiche (in particolare le prime tre) sono stati sviluppati vari algoritmi di crittografia, come ad esempio algoritmi a chiave simmetrica o a doppia chiave (pubblica e privata).

Come verrà illustrato nel capitolo 1 di questa tesi le smart card rivestono un ruolo fondamentale nel campo della crittografia e sono utilizzate in molti ambiti, come quello della telefonia mobile (paragrafo 1.3.2), del settore bancario (paragrafo 1.3.3) o nella più recente carta d'identità elettronica (discussa nel paragrafo 1.3.4).

Nel capitolo 2 verrà illustrato più nel dettaglio il funzionamento di una smart card. Si fa accenno allo standard ISO 7816 al quale risponde nel paragrafo 2.1, alla programmazione delle Java Card nel paragrafo 2.2 e allo Smart Card Web server nel paragrafo 2.3.1.

Il capitolo 3 illustra altre tecnologie utilizzate nel settore della crittografica, in particolare si accennerà al *Trusted Platrofm Module* nel paragrafo 3.3.

Capitolo 1

Introduzione alle smart card

1.1 Cosa sono le smart card

Una smart card è un dispositivo hardware realizzato su un supporto di plastica in grado di elaborare e memorizzare dati, rispondendo anche ad elevati standard di sicurezza.

La prima idea di realizzare questo tipo di dispositivi è venuta nel 1968 a due inventori tedeschi: Jürgen Dethloff e Helmut Grötrupp. Da allora, grazie alle nuove tecnologie che hanno permesso di realizzare dispositivi sempre più piccoli e potenti, tale tecnologia si è diffusa proprio grazie alla sua versatilità.

Gli utilizzi delle smart card sono molteplici, dal settore bancario, a quello della telefonia, dell'identità, fino ad arrivare nel mondo del trasporto, con l'utilizzo di biglietti elettronici. [1]

1.1.1 Caratteristiche delle smart card

Ci sono due criteri che permettono di classificare le smart card. Basandoci sulle potenzialità del circuito possiamo definire le smart card a sola memoria o a microprocessore. Invece il tipo di interfaccia di collegamento ci permette di distinguere tra smart card a contatto, senza contatto e con antenna e contattiera (ovvero a doppia interfaccia).

Le smart card a sola memoria hanno la capacità di memorizzare informazioni che possono essere lette successivamente, mentre quelle a microprocessore possono effettuare elaborazioni. Infine le smart card a contatto hanno dei pin connettori tramite i quali possono essere alimentate ed è possibile inviare e ricevere informazioni, mentre quelle senza contatto hanno a disposizione un'antenna che reagisce a un particolare campo elettromagnetico emesso da un dispositivo di lettura/scrittura.

Le smart card rispondono allo standard ISO 7816 che definisce le caratteristiche che devono avere le card a contatto, mentre altri standard sono usati per le card senza contatto (ISO 14443 e ISO 15693). Lo standard ISO 7816 sarà trattato più nel dettaglio nel paragrafo 2.1.

Le smart card a microprocessore sono particolarmente utilizzate per conservare in maniera sicura una chiave privata. Grazie alla loro capacità di elaborazione sono in grado di ricevere una piccola quantità di dati (come ad esempio un hash di un documento) e restituirlo crittografato con la chiave privata contenuta al loro

interno. In questo modo la chiave non "esce" mai dal microprocessore, e quindi non può essere letta in alcun modo. Grazie a questa caratteristica le smart card costituiscono un elemento sicuro per la firma digitale di documenti. [1]

1.2 Breve storia delle smart card

Come accennato nel paragrafo 1.1, l'idea di installare un chip elettronico su una scheda in plastica è venuta a due inventori tedeschi (Jürgen Dethloff e Helmut Grötrupp) nel 1968. Due anni dopo (nel 1970) è stato registrato il primo brevetto di smart card e nel 1976 si sono avute le prime card con microprocessore e memoria. Successivamente, nel 1079, l'azienda Motorola ha sviluppato la prima carta con chip sicuro per il settore bancario.

I primi test a larga scala delle smart card sono iniziati in Francia con i primi sportelli automatici, mentre negli stati uniti le smart card sono state inizialmente distribuiti ai lavoratori del settore agricolo. Ogni agricoltore ha ricevuto una card per la vendita delle arachidi. Una volta portato il raccolto presso un punto vendita autorizzato, i computer calcolavano la quantità del raccolto e la registravano sulle apposite card dei lavoratori.

Intanto in Europa al prima distribuzione in scala delle smart card è avvenuta ad opera dei Francesi con le card per le cabine telefoniche. La società Schlumberger è stata tra le prime ad implementare una struttura a chiave di cifratura pubblica per la gestione digitale dei certificati. Inoltre nel 1992 la Francia è stata la prima a realizzare delle card che richiedevano un PIN per essere utilizzate, introducendo le così dette *Carte Bleue* che si trattavano di carte di credito.

Un anno dopo, nel 1993, è nato il cosiddetto standard EMV (Europay MasterCard Visa) per l'utilizzo dei POS quando le varie compagnie hanno deciso di definire uno standard per le carte di credito in Europa. In America MasterCard ha iniziato ad accettare questo standard solo nel 2014. [2]

1.3 Principali utilizzi delle smart card

Oggigiorno le smart card sono vastamente utilizzate in una serie di ambiti e applicazioni differenti. In questa sezione sono elencati i principali utilizzi che vengono fatti attualmente.

1.3.1 Le smart card nella firma digitale

Come accennato nel paragrafo precedente, uno degli utilizzi delle smart card più frequente è quello nell'ambito della firma digitale.

La firma digitale è un insieme di metodi crittografici che servono a garantire l'autenticità di un messaggio trasmesso su canali non sicuri, offrendo al destinatario tre garanzie:

- L'autenticazione del mittente.
- Il non ripudio di invio del messaggio da parte dell'utente.
- L'integrità del messaggio inviato.

La firma digitale si basa su un sistema di cifratura asimmetrica (ovvero che utilizza due chiavi, una pubblica e una privata). La chiave privata è utilizzata per "firmare" il file, mentre quella pubblica, disponibile a chiunque, per la verifica della validità della firma.

Il classico funzionamento di una firma digitale consiste in pochi semplici step:

1. Tramite una funzione di hash viene generata una stringa identificativa univoca al file. File uguali generano la stessa stringa (se viene usata la stessa funzione di hash) ed ogni stringa identifica univocamente un file.

2. La stringa generata con la funzione di hash viene crittografata usando la chiave privata. Una volta cifrata, la firma viene allegata al documento, che risulta firmato.

3. Per verificare l'autenticità della firma e l'integrità del documento il ricevente può calcolare nuovamente la funzione di hash del file e decifrare la firma allegata usando la chiave pubblica del mittente. Se il documento non è stato alterato e la chiave pubblica usata per decifrare la firma corrisponde alla chiave privata usata dal mittente, allora la stringa di hash calcolata corrisponderà alla stringa decifrata. Ciò garantisce i tre parametri riportati sopra.

La chiave pubblica è solitamente fornita da una Certification Autority (CA) che garantisce che si tratti effettivamente della chiave pubblica del mittente del messaggio.

Il ruolo che la smart card gioca nella firma digitale consiste nel conservare la chiave privata in un luogo sicuro. Il codice è, infatti, salvato sulla memoria del chip presente sulla card, che poi viene reso inaccessibile dall'esterno. Una volta collegata la card a un calcolatore, tramite un apposito lettore, al momento della firma viene inviato alla card la stringa identificativa del file. La scheda provvederà poi a crittografare la stringa e a restituire il risultato dell'elaborazione al PC che provvederà infine ad allegare la vera e propria firma al documento. [3]

1.3.2 Le smart card nella telefonia mobile

Un altro utilizzo molto frequente delle smart card è quello nell'ambito della telefonia mobile.

Viene detta SIM (dall'inglese Subscriber Identity Module) una smart card che viene inserita in un telefono cellulare e utilizzata dagli operatori telefonici per conservare in modo sicuro il codice identificativo dei loro clienti (l'IMSI che corrisponde alla sigla inglese International Mobile Subscriber Identity).

Le caratteristiche tipiche di una smart card SIM sono riassunte nella tabella 1.1.

L'utilizzo che viene fatto delle sim dagli operatori è quello di fornire vari servizi ai loro clienti e controllarne l'utilizzo.

La carta SIM non contiene in memoria il numero telefonico associato all'utente, ma solo il codice identificativo IMSI ed è compito dell'operatore associarlo ad un numero telefonico; grazie a ciò è possibile avere più SIM associate allo stesso numero o portare il proprio numero da un operatore a un altro.

Descrizione	64K JavaCard 2.1.1 WIB1.3 USIM
Piattaforma	Atmel AT90SC25672RU
Architettura CPU	8-bit AVR
Tecnologia	0.15uM CMOS
ROM	256KB
Memoria non volatile	72 KB EEPROM
RAM	6Kb
Frequenza operativa interna	20-30 MHz
Tempo di vita	500mila cicli di letture/scrittura

Tabella 1.1: Tabella dei parametri tipici di una smart card SIM [4].

La SIM è protetta da un codice PIN composto solitamente da 4 o 8 cifre, l'utente ha la facoltà di disabilitare la richiesta del codice ogni volta che la SIM viene alimentata oppure cambiare il codice fornito dall'operatore. Una volta che il codice PIN è stato sbagliato tre volte, la scheda si blocca e viene richiesto un codice di sblocco a 10 cifre, denominato PUK (PIN Unblocking Key) fornito dall'operatore. Se il codice PUK viene sbagliato 10 volte la scheda si blocca definitivamente e può essere sbloccata solo dall'operatore dopo aver provato di essere l'intestatario della SIM.

Il funzionamento della SIM è molto semplice. Una volta che l'operatore riconosce il codice presente sulla carta come valido e presente nel proprio database il dispositivo mobile viene agganciato alla rete e resta in attesa che l'utente richieda un particolare servizio. Una volta effettuata la richiesta, l'operatore verifica se il servizio può essere erogato controllando ad esempio le offerte attive e il credito disponibile e può decidere se accettare o rifiutare la richiesta. [5]

1.3.3 Le smart card nel settore bancario

Un altro ambito nel quale le smart card sono ampiamente utilizzate è quello del settore bancario nel quale sono utilizzate per la realizzazione di carte di credito.

Le carte di credito sono dispositivi molto utilizzati per l'elaborazione di pagamenti e/o il trasferimento di denaro in quanto in grado di identificare l'utente utilizzatore e la banca emittente.

Inizialmente le carte di credito utilizzavano una banda magnetica per memorizzare il codice identificativo, tuttavia, per far fronte a esigenze sempre più stringenti sulla sicurezza e l'affidabilità di questi dispositivi, si è scelto di utilizzare un chip integrato rendendole, a tutti gli effetti, delle smart card. Inoltre un chip, oltre a conservare le informazioni della card in modo più sicuro, ha una memoria molto più ampia di quella che poteva avere una banda magnetica. Grazie a ciò, le card a micro chip possono offrire maggiori funzionalità delle card a banda magnetica, infatti vengono chiamate card "multi-applicazione".

Le dimensioni di una carta di credito sono definite dallo standard ISO/IEC 7810 ID01 e valgono 85,60 x 53,98 mm con uno spessore di 0,76 mm.

Alle carte di credito è sempre associato un codice PIN, che viene inserito ogni volta si voglia effettuare un pagamento.

Le carte di credito vengono principalmente utilizzate per effettuare pagamenti o prelevare denaro. Una transazione effettuata con carta di credito è un processo di autorizzazione tra tre enti:

- L'Ente emittente, ovvero la società che emette la carta di credito. Questa società stipula un contratto con il titolare della carta che viene considerato suo cliente a tutti gli effetti.
- L'Ente esercente, ovvero il commerciante che aderisce a un circuito di pagamento offrendo ai propri clienti un metodo di pagamento alternativo al contante. Per aderire al circuito, l'esercente di rivolge a una società di gestione terminali che offre la vendita o l'affitto di dispositivi POS (Point of sale) tramite i quali l'esercente può elaborare i dati della carta di credito del proprio cliente.
- Il circuito di pagamenti, invece, è l'azienda che crea e gestisce una propria rete di comunicazione alla quale sono collegati i POS degli esercenti tramite la quale viaggiano le richieste di pagamento e le eventuali autorizzazioni.

Quindi la transazione avviene attraverso tre attori:

1. Il *titolare* della carta che si impegna a restituire i soldi della spesa all'emittente della carta secondo quanto stabilito nel contratto.
2. Il *fornitore* che eroga i servizi o i beni richiesti dal titolare della carta.
3. L'*istituto* emittente che si impegna a pagare il fornitore il prezzo stabilito drante la transazione a meno di eventuali canoni concordati in fase di stipula del contratto con il fornitore.

Tra le principali tipologie di carte di credito la più comune è sicuramente la *Carta di credito "a saldo"*, fornita dalle banche a seguito dell'apertura di un conto corrente. Questa card da la possibilità di pagare tutte le spese effettuate in un mese solare in un'unica soluzione il mese successivo.

Un'altra tipologia di carta molto comune, sempre emessa dalle banche, è la cosiddetta *Carta di credito rateale o rotativo* che permette di rateizzare i pagamenti della merce acquistata.

Le principali carte di credito sono:

- American Express
- Diners
- Visa
- MasterCard
- JCB
- Discover Card
- China UnionPay

Inoltre la prima cifra del codice della card serve per identificare il circuito di appartenenza e per questo è sempre la stessa per ogni carta appartenente al medesimo circuito.

- Diners: 3
- American Express: 3
- Visa: 4
- MasterCard: 5
- Discover Card: 6

L'unico rischio in cui si può incorrere quando si ha una carta di credito consiste nella possibilità di smarrimento e/o clonazione dei codici usati per l'autenticazione e l'autorizzazione del pagamento. In caso di furto o smarrimento è sempre possibile bloccare la carta effettuando la richiesta presso il proprio fornitore. [6]

1.3.4 Le smart card nell'identificazione

A luglio 2016 è partita, in Italia, la sostituzione della carta d'identità cartacea con quella elettronica.

Oltre a cambiare le dimensioni e il materiale dal quale è costituito, il documento è stato anche munito di un microprocessore a radio frequenza, rendendolo di fatto una smart card contactless.

I dati contentuti all'interno del chip sono i seguenti:

- Comune emettitore
- Nome del titolare
- Cognome del titolare
- Luogo e data di nascita
- Sesso
- Statura
- Cittadinanza
- Immagine della firma del titolare
- Validità per l'espatrio
- Fotografia
- Immagini di 2 impronte digitali (un dito della mano destra e un dito della mano sinistra)
- Nome e cognome del padre e della madre (nel caso di un minore)
- Codice fiscale

- Estremi dell'atto di nascita
- Indirizzo di residenza
- Comune di iscrizione AIRE (per i cittadini residenti all'estero)
- Codice fiscale sotto forma di codice a barre

L'accesso ai dati presenti sul microprocessore è effettuato attraverso la tecnologia NFC (Near Field Communication), ciò permette alla carta di essere letta non solo da appositi lettori, ma anche dai più moderni smartphone dotati di lettori NFC.

L'applicazione tramite la quale viene effettuata la verifica del documento è la stessa presente anche nei passaporti moderni, denominata "ICAO MRTD" (International Civil Aviation Organization - Machine Readable Travel Documents). L'applicazione contiene tutti i dati anagrafici dell'utente che sono firmati digitalmente dal Ministero dell'Interno prima che la carta venga prodotta presso l'Istituto Poligrafico e Zecca dello Stato.

La sicurezza dei dati è garantita dal fatto che la loro lettura è consentita solo a chi può leggere le informazioni stampate fisicamente sulla carta. Questo perchè sulla card è presente una chiave di accesso richiesta per la lettura elettronica (nel CAN – Card Access Number o nell' MRZ – Machine Readable Zone). Invece, per l'accesso ai dati più sensibili, come le impronte digitali, sono richieste ulteriori autorizzazioni date a specifici organi come le forze di Polizia. Questo, insieme al fatto che la conversazione tra lettore e carta è cifrata, fa sì che le informazioni non possono essere lette all'insaputa dell'utente e blocca anche tentativi di intercettazione della comunicazione. [7]

Esempio di scansione di una carta d'identità

In questa breve sezione si mostrerà come venga utilizzata la tecnologia NFC per la lettura della smart card presente all'interno di una carta d'identità tramite l'applicazione *IDEA - Identity Easy Access* disponibile gratuitamente sul Google Play store per tutti gli smartphone Android dotati di lettore NFC.

Come primo passaggio, l'applicazione ha bisogno di leggere il codice di accesso presente sul retro della carta, ciò può essere fatto manualmente o mediante l'utilizzo della fotocamera. Questo processo è mostrato nella figura 1.1

Una volta fatto ciò è possibile avvicinare la carta al lettore NFC per far avvenire la richiesta e la comunicazione dei dati. Come mostrato nella figura 1.2.

Una volta terminata l'operazione è possibile vedere i dati che possono essere letti elettronicamente. Come si vede nella figura 1.3, i dati che possono essere letti sono: il numero del documento, il nome, il cognome, il sesso, la data di nascita, la scadenza del documento e la foto del titolare, è inoltre possibile anche verificare l'autenticità del documento.

Figura 1.1: Lettura del codice di accesso tramite la fotocamera.

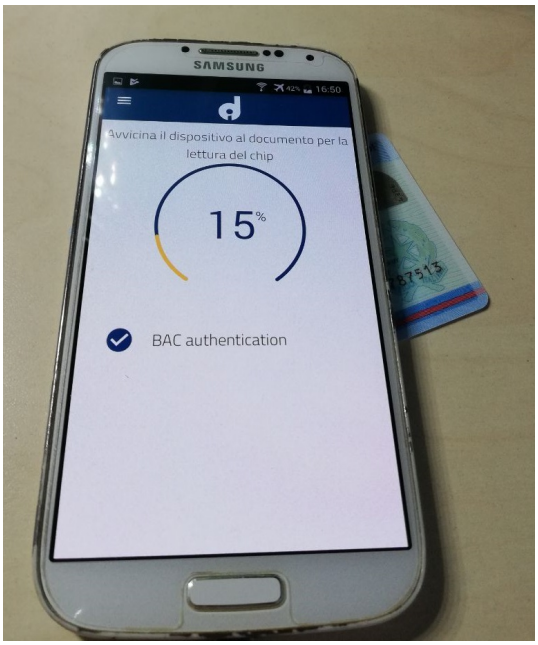

Figura 1.2: Lettura della carta tramite NFC.

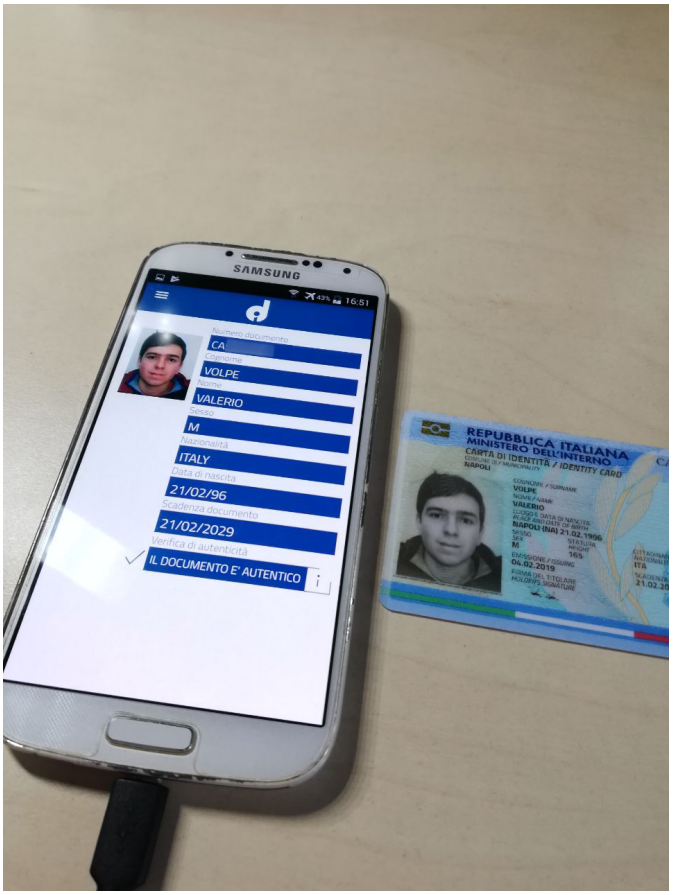

Figura 1.3: I dati letti elettronicamente dalla carta.

Capitolo 2

Smart card

2.1 Lo standard ISO 7816

Lo standard ISO 7816 è uno standard internazionale gestito dalla ISO (International Organization for Standardization) e dalla IEC (International Electrotechnical Commission).

Lo standard è composto da varie parti, ognuna delle quali serve per descrivere un dato aspetto delle card, le più importanti sono le prime 4 che descrivono le caratteristiche fisiche ed elettroniche della carta nonché l'organizzazione dei file, la sicurezza e i comandi per lo scambio di informazioni. [8]

2.1.1 ISO 7816-1 e ISO 7816-2

La prima e la seconda parte dello standard descrivono le caratteristiche prettamente fisiche della card, come la dimensione della carta e dei contatti, la resistenza a flessione e piegamento nonché i materiali da utilizzare per la loro fabbricazione. Questo per garantire elevati standard di sicurezza e un tempo di vita adeguato.

Inoltre lo standard definisce anche i limiti di esposizione a raggi X, luce ultravioletta, campi elettromagnetici e temperatura che la carta deve sopportare.

Tutte queste informazioni servono ai costruttori per la fabbricazione delle carte. [9]

2.1.2 ISO 7816-3

La terza parte dello standard definisce le caratteristiche elettriche dei contatti e i protocolli di comunicazione con la card. Queste informazioni sono di fondamentale importanza per chi fabbrica lettori o per sviluppatori che vogliono stabilire una comunicazione con il chip presente sulla scheda. [9]

In particolare ci sono tre classi di carte a seconda della tensione di alimentazione alla quale lavorano (VCC): Classe A (VCC = 5V), classe B (VCC = 3V) e classe C (VCC = 1.8V).

Sulla card sono presenti 7 contatti, il primo e il quinto sono rispettivamente l'alimentazione (VCC) e il ground (GND), il secondo serve per inviare un segnale di reset, il terzo serve per inviare gli impulsi del clock. Il sesto contatto viene riservato per un utilizzo standard o proprietario come seconda porta I/O. Infine l'ultimo contatto serve per un I/O dei dati in maniera seriale. Il sesto contatto dal 1990

viene spesso utilizzato per fornire alla scheda una tensione di programmazione. [10]

2.1.3 ISO 7816-4

La quarta parte dello standard è sicuramente la più interessante per i programmatori che utilizzano un linguaggio più ad alto livello (come ad esempio il Java - paragrafo 2.2) per realizzare applet, salvare informazioni o far eseguire alcune operazioni dal processore della scheda.

In questa parte dello standard sono indicati i comandi che può ricevere una smart card insieme alla struttura del file system e l'architettura di sicurezza che definisce i diritti di accesso ai dati presenti sulla memoria della card.

Per comunicare con la carta va inviato un comando e attesa una risposta, come indicato dalla terza parte dello standard i bit sono inviati e letti in maniera sequenziale. Un comando è composto da un header e da un corpo. L'header è composto da tre campi: 1 byte di classe denotato CLA, 1 byte di istruzione denotato INS e 2 byte per i parametri denotati P1 e P2 rispettivamente.

All'header seguono una serie di byte per l'invio di eventuali dati alla card. Il primo byte indica il numero di byte che saranno inviati, questo è denominato come L_c. Questo campo può non essere presente se non vengono inviati dati, oppure occupare fino a 3 byte. Ovviamente subito dopo si hanno gli N byte indicati dal campo L_c usati per inviare i dati necessari alla carta. Infine il comando si chiude con un campo L_e che indica il numero massimo di byte che ci si aspetta come risposta dalla carta. Anche questo campo può essere assente o occupare fino a 3 byte.

La risposta è molto più semplice e contiene solo due campi: il primo è un numero di byte al più uguale a quanto indicato dal campo L_e che contiene gli eventuali dati inviati dalla risposta. Il secondo campo è formato da due byte di stato denominati rispettivamente SW1 e SW2.

File system

Come specificato dallo standard ISO 7816-4 i file presenti sulla smart card si dividono principalmente in due categorie: i file dedicati (DF - dedicated files) e i file elementari (EF - elementary files). I primi sono i file delle applicazioni e delle strutture dati, mentre i secondi sono file che contengono dati. I DF possono essere imparentati tra di loro, mentre ciò non è possibile per gli EF. Gli EF sono a loro volta divisi in due categorie gli EF di lavoro e gli EF inteni. I primi sono utilizzati "dal mondo esterno" mentre i secondi sono dati interni utilizzati dalla carta.

I file sono organizzati in una struttura albero il cui file DF che si trova alla radice viene chiamato master file (MF), come mostrato in figura 2.1.

Sicurezza

L'accesso ai file è, naturalmente, protetto. Lo standard fa riferimento a uno stato di sicurezza ovvero uno stato che è possibile ottenere al seguito di una procedura di autenticazione nella quale viene richiesta, ad esempio, la conoscenza di una password o di una key.

Vengono considerate quattro tipologie di stato di sicurezza:

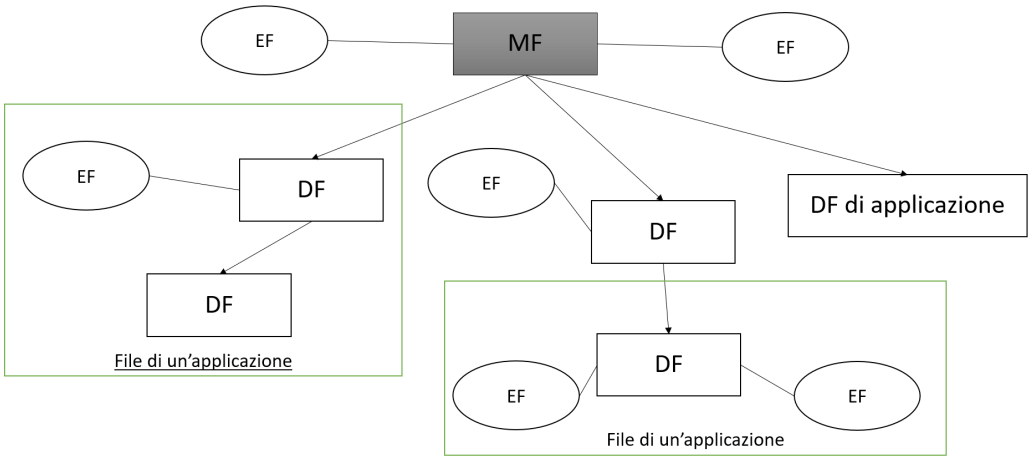

Figura 2.1: Esempio di una gerarchia di file in una smartcard.

- **Stato di sicurezza globale**, è solitamente legato a una procedura di autenticazione risiedente nel MF e quindi serve per proteggere l'intera struttura del file system.
- **Stato di sicurezza specifico per un'applicazione**, può essere modificato al seguito del completamento di una procedura di autenticazione basata su un'applicazione, può essere mantenuto, perso o recuperato quando durante la procedura di selezione di un'applicazione. Questo stato è rilevante solo per l'applicazione alla quale fa riferimento.
- **Stato di sicurezza specifico per un file**, il funzionamento è simile allo stato relativo a un'applicazione, tuttavia lo stato si basa su un DF specifico e può essere mantenuto, perso o recuperato quando viene selezionato un file. Uno stato di sicurezza globale può essere visto come un caso particolare.
- **Stato di sicurezza relativo a un comando**, quest'ultimo stato esiste solo quando viene processato un comando usando una comunicazione sicura che richiede un'autenticazione.

Una volta stabilito lo stato in cui si trova chi sta comunicando con la carta, è possibile definire gli attributi di sicurezza, questi definiscono le azioni permesse e le condizioni che devono essere verificate per poterle effettuare, inoltre differiscono tra DF e EF e si basano su parametri opzionali presenti in un dato file o nel suo "padre". In particolare questi attributi specificano in quale stato bisogna trovarsi per poter accedere a un dato file e le funzioni (ad esempio *sola lettura*) che sono disponibili quando ci troviamo in un dato stato.

Ultimo aspetto da considerare sono i meccanismi di sicurezza per l'autenticazione. Prima di tutto troviamo l'autenticazione tramite password o tramite key, per la prima la card compara dati inviati dall'esterno con dati segreti presenti al suo interno, mentre per la seconda viene richiesto a chi vuole accedere alla card di mostrare le conoscenza di una chiave privata.

Un altro meccanismo di sicurezza è quello sull'autenticazione dei dati che consiste nell'utilizzo di una chiave segreta o privata per controllare i dati ricevuti dall'esterno. In alternativa la card può usare sempre una chiave segreta o privata per calcolare una checksum o firma digitale da inviare al lettore. Infine l'ultimo meccanismo disponibile e quello della crittografia dei dati, secondo il quale la scheda decifra o cifra dati scambiati con l'esterno usando una chiave segreta o privata.

Se richiesto dall'applicazione i risultati di un'autenticazione possono essere salvati su un file di log interno (EF). [10]

2.2 Java Card

La tecnologia JavaCard permette di realizzare applicazioni in linguaggio Java, dette applet, da far girare sui chip delle smart card in sicurezza. Questa teclogogia è ampiamente utilizzata nel settore delle SIM (di cui un esempio è riportato nel paragrafo 2.3.2) e delle carte bancarie. [11]

Con l'ausilio del manuale ufficiale pubblicato da Oracle [12] verrà illustrata sinteticamente la piattaforma **Java Card 3** ovvero un kit di sviluppo per applet Java Card.

2.2.1 Architettura della piattaforma

L'architettura classica della piattaforma è costruita su una classica macchina virtuale java.

Il kit di sviluppo include anche un simulatore (*Java Card RE*) che simula l'ambiente che si avrebbe su una carta fisica e implementa le specifiche dello stadard ISO:7816-4:2013 che è trattato più approfonditamente nel paragrafo 2.1. Il simulatore supporta anche venti canali logici e le estensioni APDU (*Application Protocol Data Units*) definite nello standard ISO 7816-3.

Java Card TCK

Nel kit di sviluppo fornito da Oracle viene anche fornita una suite di test automatica e configurabile chiamata *Java Card Technology Compatibility Kit* atta a verificare la compatibilità tra l'applet che si sta sviluppando e le specifiche della carta sulla quale la si vuole far girare.

2.2.2 Sviluppo degli applet

Lo sviluppo delgi applet può essere effettuato tramite l'ide Eclipse installando l'*Eclipse Java Card Plug-in*.

I passaggi per lo sviluppo di un applet sono i seguenti:

- Installare e impostare l'ambiente di sviluppo usando l'IDE Eclipse e il Plug-in
- Sviluppare l'applet.
- Fare il debugging dell'applet.

- Creare il file CAP che può essere scaricato sul simulatore o su una card compatibile. Il file viene inviato tramite APDU. Il kit offre un convertitore utilizzabile per generare file CAP da inviare alla scheda (vedere il paragrafo 2.2.3).

2.2.3 Utilizzo dei file CAP

Per essere installato su una smart card un applet deve essere convertito in Converted Applet (CAP). Per generare questo tipo di file il kit mette a disposizione un convertitore.

Un file CAP utilizza il formato JAR (Java Archive) e, oltre a varie informazioni sul pacchetto Java, contiene un manifesto (*META-INF/MANIFEST.MF*) che fonrnisce una serie di informazioni riguardanti il file. Queste informazioni possono essere usate per facilitare la distribuzione del file.

Le informazioni presenti nel file sono presentate con lo standard *nome:valore* e sono riportate nella tabella 2.1.

Nome	**Descrizione del valore**
Java-Card-Creation-Time	Indica quando è stato generato il file CAP
Java-Card-Converter-Version	Indica la versione del convertitore utilizzata
Java-Card-Converter-Provider	Indica il fornitore del convertitore
Java-Card-File-Version	Versione del file CAP secondo la convenzione *maggiore.minore*
Java-Card-Package-Version	Versione del pacchetto file CAP secondo la convenzione *maggiore.minore*
Java-Card-Package-AID	AID (*JADE Agent Identifier*) del pacchetto
Java-Card-Package-Name	Il nome del pacchetto
Java-Card-Applet-<n>-AID	AID (*JADE Agent Identifier*) dell'applet *n*
Java-Card-Applet-<n>-Name	Il nome breve della classe *n* implementata dal CAP
Java-Card-Import-Package-<n>-AID	AID (*JADE Agent Identifier*) del pacchetto *n* importato
Java-Card-Import-Package-<n>-Version	Versione del pacchetto *n* importato secondo la convenzione *maggiore.minore*
Java-Card-Integer-Support-Required	Può assumere valori *TRUE* o *FALSE*. Il valore vero indica che il pacchetto richiede l'integer support

Tabella 2.1: Tabella dei parametri presenti nel manifesto del file.

Generazione di un file CAP

Per generare un file CAP è possibile utilizzare il tool *capgen* fornito dal kit.

Per avviare il tool è possibile usare il comando

capgen.bat ***[opzioni] nome_del_file.***

Le opzioni disponibili sono:

- **-help** stampa un messaggio di aiuto.
- **-nobanner** sopprime i vari messaggi.
- **-o *nome_del_file*** permette di specificare un file di output.

2.2.4 Fare il debugging delle applicazioni

Per effettuare il debugging delle applicazioni il kit offre due tool che lavorano insieme per alleggerire la procedura, altrimenti troppo impegnativa per la virtual machine del simulatore.

Il primo tool, *cref* può essere lanciato direttamente da Eclipse o da riga di comando e ha la possibilità di simulare la memoria persistente (EEPROM) nonché di salvare e recuperare i dati salvati sulla memoria o su file presenti sull'hard disk. Inoltre il tool può eseguire operazioni di I/O tramite un'interfaccia socket che simula il collegamento tra il lettore di carte e il computer.

Per il debugging l'IDE utilizza il Java Debug Wire Protocol (JDWP) che, come accennato, è troppo pesante per la piccola VM utilizzata dal simulatore fornito dal kit. Per questo per il debugging viene utilizzato un protocollo proprietario più leggero.

Il secondo tool offerto è il *debugproxy*, esso ha il compito di tradurre comandi e risposte tra l'IDE e il simulatore *cref* utilizzando il protocollo appropriato.

Dato che i tool *cref*, *debugproxy* e l'IDE comunicano tramite socket, essi possono girare su host diversi.

Da Eclipse è possibile far partire il debug proxy per impostare breakpoints, leggere o impostare variabili e fare il debug di una libreria.

2.2.5 Distribuire le proprie applicazioni

Il kit permette di scaricare un Java Card technology package, effettuare il collegamento con la card, eliminare applets e pacchetti dalla smart card e impostare le applet di default per i vari canali logici.

Per installare l'applicazione su una card bisogna prima convertire i file .class in file .cap utilizzando il convertitore fornito dal kit, successivamente il tool *scriptgen*, ovvero l'off-card installer converte il file .cap in uno script .scr che consiste in una serie di comandi APDU che vengono eseguiti dall'APDUtool. Infine l'on-card installer processa il file CAP e invia gli APDU di risposta all'APDUtool con lo stato ed eventuali dati.

Il file .scr contiene comandi e C-APDU che sono terminati da un punto e virgola. La sintassi di un C-APDU è la seguente:

<CLA> <INS> <P1> <P2> <LC> [<byte 0>...<byte LC-1>] <LE>;

dove

- <CLA> è il byte della classe definito dallo standard ISO 7816-4.
- <INS> è il byte di istruzione definito dallo standard ISO 7816-4.
- <P1>, <P2> sono i parametri P1 e P2 definiti dallo standard ISO 7816-4.

- <LC> è il numero dei byte inviati (1 in modalità non estesa, 2 in modalità estesa).
- <byte 0>...<byte LC-1> sono i byte per i dati di input.
- <LE> è la lunghezza che ci si aspetta per l'output (1 in modalità non estesa, 2 in modalità estesa).

Il protocollo utilizzato per installare un applet è composto da una sequenza di comandi ben precisa. Per prima cosa viene inviato un comando di selezione utilizzato per invocare l'on-card installer, segue un comando di CAP Begin. Successivamente viene ripetuta una serie di 3 comandi per ogni componente presente nel file CAP (Component ## Begin, Component ## Data, Component ## End). La sequenza viene conclusa dai comandi CAP End e Create Applet. Ogni comando viene inviato alla card e riceve una risposta che varia a seconda del comando.

2.3 Funzionalità avanzate

La tecnologia della smart card si è evoluta nel tempo, grazie anche ai nuovi sviluppi tecnologici che hanno portato alla realizzazione di circuiti integrati sempre più piccoli e veloci. Ciò ha permesso lo sviluppo di smart card sempre più potenti capaci di far girare programmi sempre più complessi.

2.3.1 Smart Card Web Server

Uno *Smart Card Web Server* (SCWS) è un server HTTP implementato all'interno di una smart card, di solito integrata in uno smartphone (SIM - vedi paragrafo 1.3.2). Questa tecnologia permette agli operatori di telefonia mobile di offrire determinati servizi utilizzando il diffusissimo protocollo *HTTP/1.1*.

Il principale obiettivo di questa tecnologia è quello di creare una comunicazione interna al dispositivo tra un WEB browser che gira nello smartphone e un server presente sulla card. Permette, inolte, un'amministrazione remota della smart card da un'entità autorizzata (ad esempio il fornitore del servizio di telefonia e/o della card stessa).

L'interfaccia HTTP offerta dalla card si trova a un livello logico separato dall'interfaccia di comunicazione definita dallo standard ISO 7816 (vedi paragrafo 2.1) e permette ad applicazioni HTTP di comunicare con la card in maniera autonoma.

Questo canale di comunicazione è utilizzato dai fornitori per offrire maggiori servizi ai loro clienti.

L'URL utilizzato per la comunicazione deve corrispondere alla definizione data dal protocollo *HTTP/1.1*, ovvero

http_URL = "http:" "//" host [":" port] [abs_path ["?" query]]

La $< query >$ opzionale deve essere una sequenza di uno o più termini della forma $< name >=< value >$ separati da un '&'. Il server deve poter rispondere perlomeno a url di 1024 caratteri e *abs_path* di 128 caratteri.

Sicurezza della comunicazione

Per la sicurezza della trasmissione dei dati con la card viene utilizzato il protocollo *Transport Layer Security* (TLS). Questo protocollo fornisce un meccanismo sicuro ed affidabile per il trasporto dei dati tra due entità, con un controllo dell'integrità e della confidenzialità delle informazioni scambiate. Fornisce anche dei meccanismi di autenticazione per una o entrambe le parti.

Il TLS è pesato per un paradigma Client-Server dove il client è chi inizia la comunicazione (o invia una richiesta) e il server fornisce una risposta. Solitamente il client può autenticare il server utilizzando un certificato a chiave pubblica. Un'auntenticazione mutua può essere effettuata utilizzando dei certificati a chiave pubblica pre-condivisi (*Pre Shared Keys-TLS* o PSK-TLS) [13]

2.3.2 Applicazione SIM

La SIM card (introdotta nel paragrafo 1.3.2) oltre a conservare il codice identificativo dell'utente offre allo stesso anche alcune funzionalità, come ad esempio il salvataggio di contatti telefonici, utile quando si cambia cellulare e non si vuole perdere la lista di contatti salvata in rubrica.

Inoltre alcuni operatori offrono delle semplici applicazioni presenti all'interno delle loro SIM card, che presentano un'interfaccia molto semplice (a menù) con alcune funzioni che possono essere richieste dall'utente. Queste applicazioni possono anche aprire URL, mandare SMS, avviare chiamate e reagire a particolari eventi come l'arrivo di una chiamata o la disconnesione di una chiamate (per inviare, ad esempio, un SMS all'utente con il credito residuo o i minuti ancora disponibili nell'offerta). Inoltre c'è la possibilità di interagire con altre card, nel caso di telefoni dual SIM.

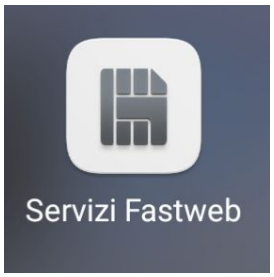

Figura 2.2: Icona dell'applet della SIM di Fastweb.

Nella figura 2.2 è possibile vedere l'icona dell'applet fornito dall'operatore Fastweb, mentre nella figura 2.3 è possibile vederne la schermata principale.

Il modo più semplice per realizzare applicazioni che girano su smart card (più propriamente dette applet) è quello di utilizzare il liguaggio Java e le Java Cad, di cui si parlerà più nel dettaglio nel paragrafo 2.2. [4]

2.3.3 Applet firewall

All'interno di una smart card con sistema operativo Java card è possibile installare più applet diversi che convivono sulla card. In base alla funzionalità implementata

Figura 2.3: Schermata dell'applet della SIM di Fastweb.

da ogni applet è possibile che debba salvare e manipolare dati sensibili, come una valuta virtuale, delle impronte digitali o chiavi di crittografia.

Per questo motivo bisogna fare in modo che un applet non possa accedere ai dati delle altre applicazioni che convivono sulla scheda. Un applet firewall ha il compito di confinare ogni applet in una porzione della memoria e bloccare eventuali accessi ad aree designate alle altre applicazioni.

Il firewall permette, comunque, un meccanismo di condivisione di alcune risorse in maniera controllata e sicura; questo meccanismo influisce sulla programmazione di un applet e va tenuto conto dagli sviluppatori.

Il meccanismo di firewall permette di evitare che errori della programmazione o del design di un applet possano portare alla "fuoriuscita" di dati sensibili. Inoltre il firewall perviene anche tentativi di hacking, in quanto se un'applicazione malevola viene installata su una smart card e riesce ad ottenere il riferimento pubblico di un oggetto che appartiene ad un'altra applicazione, nel momento in cui tenta di accederci il firewall la bloccherà. In questo modo non viene impedita la regolare esecuzione dell'applet "lecito".

Il firewall divide i vari oggetti in *contesti*, ovvero aree di memoria protette. Il bordo di ogni contesto è costituito dal firewall stesso. Quando un'applicazione viene eseguita, riceve un contesto (detto *group context*). Se ci sono più istanze di applicazioni che fanno parte dello stesso package, allora hanno in assegnazione lo stesso contesto e possono condividere gli oggetti che vi appartengono.

Inoltre il JCRE (*Java Card Runtime Environment*) ha il proprio contesto, ovvero un contesto di sistema con privilegi speciali. Il JCRE può accedere, ad esempio, a tutti gli altri contesti, ma il viceversa è proibito.

La figura 2.4 mostra un esempio di una possibile divisione dei contesti tra i vari applet. [14]

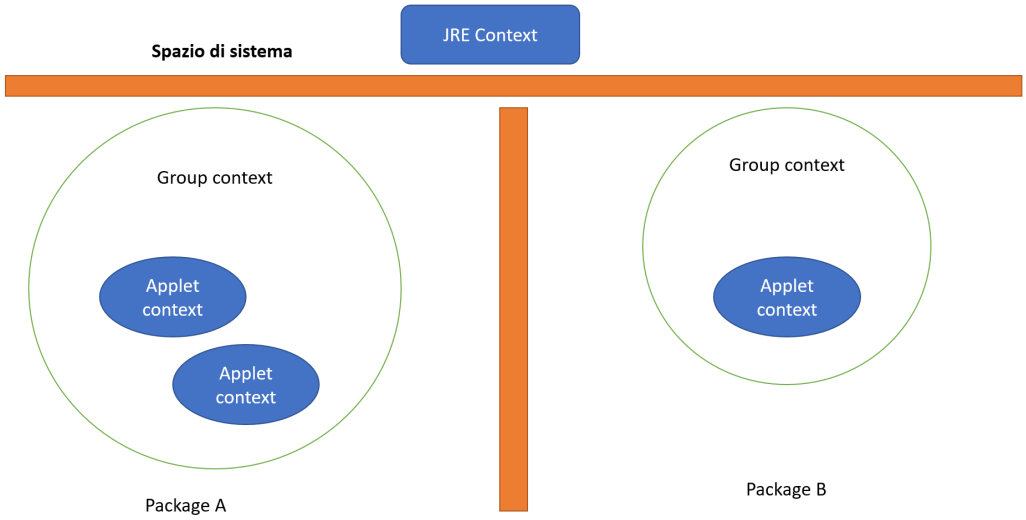

Figura 2.4: Schema logico delle divisione dei contesti in una Java Card.

Capitolo 3

Secure Access Module

3.1 Introduzione

Un SAM (*Secure Access Module*) è un circuito integrato che offre funzionalità simili alle Smart Card. Sono usati principalmente per conservare chiavi crittografiche che vengono utilizzate dal dispositivo nel quale sono integrati.

Come per le Smart card, una volta salvata una chiave in un dispositivo SAM durante, ad esempio, la sua produzione, essa non uscirà più dal dispoditivo. Ciò garantisce che la chiave non possa essere letta, ma solo utilizzata (se si dispone dell'autorizzazione).

I SAM posseggono abbastanza potenza da poter effettuare operazioni non troppo complesse, come generare chiavi di sessione o crittografare una piccola mole di dati (tipicamente un hash).

La Smart card è un esempio particolare di SAM, tuttavia la differenza sostanziale è che, mentre una Smart card può essere inserita in lettori diversi, un circuito SAM è integrato in un dispositivo "più grande". In questo senso non è un dispositivo *stand alone* (ovvero che sta "da solo") come una Smart card.

I SAM, oltre ad essere utilizzati come posti sicuri per conservare chiavi di crittografia possono anche essere utilizzati come proxy per ottenere una comunicazione più sicura con una smart card.

Quando un'applicazione deve accedere alla smart card per comunicare può utilizzare il SAM come tramite per fare in modo che i dati che vengono scambiati con la card siano criptati, come mostrato in figura 3.1.

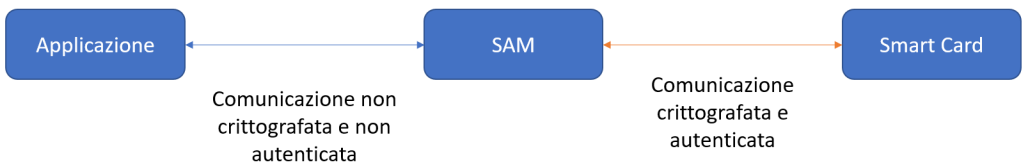

Figura 3.1: Schema della comunicazione utilizzando un SAM.

3.2 Point of sale

Il POS (*Point of sale*) è un dispositivo elettronico utilizzato dai commercianti per dare la possibilità ai loro clienti di acquistare merci e/o servizi tramite l'utilizzo di moneta virtuale (carte di credito o bancomat).

Il funzionamento di un POS è molto semplice, prima di tutto il terminale deve leggere la carta del cliente e verificarne la validità. Per la lettura vi sono varie tecnologie disponibili, dalla ormai inutilizzata banda magnetica, alla lettura della smart card a contatti presente sulla carta, alla più recente tecnologia contactless che prevede l'avvicinamento della carta al dispositivo senza effettivamente toccarlo. Quest'ultima opzione può essere anche sfruttata per pagare con il cellulare tramite i servizi offerti da Google e Apple.

Una volta letta la carta, va autorizzata la transazione. Per farlo vi sono due possibili modi:

- Tramite firma. Il POS stampa due scontrini, uno dei quali va lasciato al cliente come ricevuta, mentre l'altro va firmato dallo stesso. Se la firma presente sullo scontrino non è la stessa presente sulla carta, allora la transazione viene considerata fraudolenta.
- Tramite PIN. Il cliente inserisce un PIN segreto associato alla carta. Ciò aumenta di molto la sicurezza della transazione.

[15]

3.3 Trusted Platform Module

Un Trusted Platform Module (o TPM) è un microchip elettronico utilizzato per la sicurezza informatica e implementato nelle schede madri o in altri dispositivi elettronici, può essere considerato un SAM in senso lato. Questi chip sono dotati di una coppia di chiavi e un modulo che implementa la crittografia asimmetrica (RSA) dei dati. Essendo le chiavi diverse per ogni dispositivo, esse permettono di identificarlo univocamente.

3.3.1 Specifiche del TPM

Le specifiche di questo tipo di dispositivi sono state pubblicate dal *Trusted Computing Group*. Le funzionalità che questo dispositivo deve poter offrire sono:

- La generazione di numero pseudo-casuali.
- La generazione e memorizzazione di chiavi crittografiche asimmetriche.
- La cifratura e de-cifratura di dati mediante l'algoritmo RSA.
- La generazione e verifica di hash SHA1

Per realizzare queste funzionalità il chip deve avere una serie di componenti che comunicano utilizzando un bus interno e in grado di comunicare con il bus della scheda madre sulla quale il dispositivo è installato.

Dispositivo I/O

Come accennato, il dispositivo I/O deve essere in grado di far passare le informazioni dal bus interno del chip a quello esterno e viceversa.

Coprocessore Crittografico

Il coprocessore crittografico deve essere in grado di garantire le funzionalità che sono state elencate in precedenza. Questo dispositivo può anche utilizzare la crittografia asimmetrica per scambiare dati internamente. Inoltre per la firma digitale si usano chiavi a 2048 byte anche se il modulo deve comunque supportare chiavi a 512 e 1024 byte.

Generatore di Chiavi Crittografiche

Questo dispositivo deve semplicemente generare una coppia di chiavi crittografiche secondo L'Algoritmo RSA. Non ci sono specifiche sul tempo necessario per il calcolo, che può essere arbitrariamente lungo.

Motore HMAC

Si tratta di un dispositivo preposto alla verifica che i dati di identificazione siano corretti e al tempo stesso non siano stati manipolati. Per ottenere ciò si utilizza l'algoritmo HMAC con chiavi di 20 byte e blocchi di dati di 64 byte. Questo algorimo, utilizzando una parte del messaggio originale e della chiave per la crittografia dei dati, garantisce massima sicurezza.

Generatore di Numeri Pseudocasuali

Il generatore di numeri pseudocasuali serve per introdurre della casualità all'interno del TPM. Viene utilizzato per generare le chiavi asimmetriche per la firma digitale. Questo dispositivo deve essere composto da un componente in grado di ricevere dati imprevedibili e un coprocessore in grado di generare i numeri utilizzando una funzione non invertibile. Ogni volta che viene attivato, il generatore deve fornire 32 byte di dati casuali.

Quando viene prodotto il TPM il generatore viene inizializzato con dati casuali tramite disturbi termici o via software. Una volta inizializzato nessuno (nemmeno il produttore) deve poter essere in grado di modificare lo stato del componente. Durante il suo funzionamento il generatore utilizzerà altri dati imprevedibili come ad esempio il movimento del mouse o i tasti premuti sulla tastiera. Infine la funzione di output riceve un numero minore di input per produrre i dati necessari, in modo da creare ambiguità e garantire l'invertibilità della funzione utilizzata.

Motore SHA-1

Tale componente viene utilizzato per la firma digitale dei file e deve poter gestire hash SHA-1 da 160 bit.

Gestore di alimentazione

Questo dispositivo non solo deve alimentare il TPM, ma deve anche informare i suoi dispositivi interni dello stato dell'alimentazione. Ciò per evitare che possano essere collegati altri dispositivi (operatori esterni) non autorizzati al chip.

Opt-In

Questo componente ha il compito di gestire l'accensione, spegnimento, attivazione e disattivazione del TPM non ché la presenza di operatori esterni collegati al dispositivo. L'Opt-In deve garantire che le operazioni di accensione/spegnimento e attivazione/disattivazione del TPM sono effettuate da utenti abilitati al controllo del TPM (*TPM-Owner*).

Motore di esecuzione

Il compito del motore di esecuzione è quello di eseguire il codice che il chip riceve dal dispositivo di Input/Output garantendo la trasparenza delle operazioni eseguite e la protezione dei dati sensibili.

Memoria non volatile

L'ultimo componente che troviamo nel TPM è una memoria non volatile utilizzata per memorizzare i dati che identificano il TPM. Questa memoria deve poter essere letta solo dal proprietario del modulo per poter conservare i dati in maniera sicura.

3.3.2 Utilizzo del TPM

Il TPM, per essere utilizzato, richiede un software particolare, ovvero uno specifico *Trusted Software Stack*, definito anch'esso dal Trusted Computing Group.

Utilizzando il software giusto il TPM è in grado di attestare da remoto l'identità della macchina su cui è montato e di cifrare e decifrare i dati che viaggiano da e verso la memoria di massa (quest'ultimo processo è denominato *data sealing* e *data binding*).

Il processo di data sealing consiste nella cifratura dei dati che vengono salvati sulla memoria di massa del PC tramite l'utilizzo di una chiave che dipende dallo stato del computer (ovvero dall'hardware e il software utilizzato in quel momento). Ne consegue che la de-cifratura dei dati può essere fatta solo calcolando la medesima chiave e quindi avendo un PC il cui hardware e software in esecuzione corrisponda a ciò che è stato utilizzato per cifrare i dati.

Il binding dei dati, invece, consiste nella cifratura delle informazioni usando una chiave RSA detta *Endorsment Key* (chiave di approvazione) che identifica univocamente il TMP, garantendo, quindi, che i dati possono essere decifrati solamente dal chip utilizzato per cifrarli.

L'attestazione della macchina viene fatta con apposite chiavi dette *Attestation Identity Key* (AIK) generate dal TPM. Tuttavia dalla versione 1.2 è possibile anche utilizzare un protocollo che preserva la riservatezza dei dati del sistema operativo, denominato *Direct Anonymous Attestation*. In sostanta il chip ha particolari registri detti *Platform Configuration Register* (PCR) che utilizza per salvare un

hash dell'evoluzione dello stato del sistema, effettuando un controllo sull'hardware e software installato e misurandolo secondo la formula 3.1.

$$PCR[i]_{t1} = SHA1(PCR[i]_{t0} + informazioni\_prelevate_{t1}) \tag{3.1}$$

In altre parole, il contenuto dell'i-esimo registro viene aggiornato (al tempo *t1*) con l'hash del contenuto presente nel registro al tempo precedente (ovvero *t0)* più le informazioni prelevate al tempo *t1*. Quando il sistema è evviato il contenuto del PCR è azzerato.

Disponendo delle informazioni conenute nel PCR, il software può autenticare la macchina e decidere quali operazioni intraprendere. [16]

Conclusione

Il mondo delle Smart Card e dei Token crittografici è enorme e in continua evoluzione, data l'importanza e la delicatezza dei problemi che si vogliono risolvere con queste tecnologie.

In questa tesi è stata fatta una breve panoramica su concetti molto importanti, ognuno dei quali richiederebbe un libro a parte per essere trattato in maniera sufficientemente approfondita.

Si spera che chi è interessato al settore della sicurezza informatica abbia trovato utili gli argomenti esposti e, aiutandosi con la bibliografia, riesca a trovare dei riferimenti validi per ampliare la sua conoscenza.

Bibliografia

[1] Smart card - wikipedia. https://it.wikipedia.org/wiki/Smart_card. 31/07/2019.

[2] The history of smartcards. https://www.hospitalityupgrade.com/. 4/08/2019.

[3] Firma digitale - wikipedia. https://it.wikipedia.org/wiki/Firma_digitale. 1/08/2019.

[4] Karl Koscher Eric Butler. *The Secret Life of SIM Cards*.

[5] Carta sim - wikipedia. https://it.wikipedia.org/wiki/Carta_SIM. 1/08/2019.

[6] Carta di credito - wikipedia. https://it.wikipedia.org/wiki/Carta_di_credito. 2/08/2019.

[7] Carta d'identità elettronica. https://www.cartaidentita.interno.gov.it. 2/08/2019.

[8] Iso/iec 7816 - wikipedia. https://it.wikipedia.org/wiki/ISO/IEC_7816#7816-4:_Organizzazione,_sicurezza_e_comandi_per_interscambio. 11/08/2019.

[9] Iso 7816 smart card standard - cardwerk. https://cardwerk.com/iso-7816-smart-card-standard/. 11/08/2019.

[10] *International standard*. ISO/IEC, 2006.

[11] Java card - wikipedia. https://it.wikipedia.org/wiki/Java_Card. 4/08/2019.

[12] *Java Card 3 Platform*. Oracle, United States, 2017.

[13] *Smartcard-Web-Server*. Open Mobile Alliance, 2013.

[14] Zhiqun Chen. *Java Card Technology for Smart Cards: Architecture and Programmer's Guide*. Sun microsistem, 2004.

[15] Pos: costi, funzionamento, obbligo in italia e novità 2018-2019. https://www.pagamentidigitali.it/carte/pos-costi-funzionamento-obbligo-italia/. 30/08/2019.

[16] Trusted platform module - wikipedia. https://it.wikipedia.org/wiki/Trusted_Platform_Module. 17/08/2019.

Licenza