Structures used in Secure Automatic Ticketing System

MARIUS POPA, CRISTIAN TOMA
Department of Computer Science in Economics
Academy of Economic Studies – Bucharest
Romana Square, No. 6, Bucharest
ROMANIA
marius.popa@ase.ro cristian.toma@ie.ase.ro

Abstract: The paper presents a solution for an automatic ticketing system and some aspects regarding security assurance of this kind of system. Some concepts and terms used in development of secure automatic ticketing system are presented. It is depicted an architecture o secure automatic ticketing system with its components and their roles in this architecture. Also, there are highlighted technical details of the cards used in implementation of secure automatic ticketing system.

Key-Words: Ticketing System, Distributed Informatics System, Informatics Security, Smart Card

1 Introduction

The chapter presents concepts that are used in the designed solution for Secure Automatic Ticketing System – SATS. SATS has several major objectives:

- to implement the secure use of the electronic cards and tags instead of paper tickets in any kind of information integrated system;
- to supervise the actions and the behavior of the subscribers within the ticketing system in order to prevent the frauds and to increase the subscribers and the clients satisfaction;
- to improve the management of the company providing complete and proper information about the components of the system;
- to improve the commercial offers to the subscribers and to the clients;
- to improve the quality of commercial services.

SATS operates with various new terms as it follows:

- *Charging* – the action of the registering and paying a service in advance at a Point of Sale – POS; for backward compatibility with the old ticketing systems, the SATS accepts the acquisition of paper tickets; for instance, in a public transportation system, the client buys 20 Euros credit and the amount is stored in E-Pocket from the E-Card; the client is free to use the amount whenever and wherever the one wants; he/she can pay the journey with the bus number 300, from the station A to B (1 Euro), and the next day, he/she can pay another journey with bus 301, from station C to D (2 Euros); after these actions the client would still have 17 Euros credit in the E-Pocket;
- *Client* – person who is gone to use the services provided by the system which choose to implement SATS as ticketing system; a client can be pre-pay (make a charging), post-pay (make a subscription) or both (he/she is also a pre-pay and a post-pay client);
- *E-Card* – the integrated circuits contactless card, memory chip card – Mifare, which is used as base for actions such as subscription and charging; it replaces the paper ticket use in the current ticketing systems;
- *E-Personal-Area* – it is a memory area in the E-Card where the information about the client is stored, such as ID Number, Social Security Number, Personal Number Identification, first and last name and so forth;
- *E-Pocket* – it is a memory area in the E-Card where the money of the client is stored; the E-Card includes the E-Pocket;
- *E-Subscription* – it is a memory area in the E-Card where the information about the subscription is stored; the E-Card includes the E-Pocket;
- *E-Validation Device* – it is combo equipment, hardware and software, which has usually an operating system (Windows CE), a LCD colour touch screen, a contact-less reader, an incorporated printer, green and red light with sound warnings for validation/invalidation and a Secure Application Module – SAM;
- *Pre-pay Client* – a client who chooses to make a subscription at a POS;
- *Post-pay Client* – a client who chooses to make a charging at a POS;

- *Point of Sale POS* – a location where a client can make a subscription or a charging;
- *Personalization* – the process of printing on the card the picture of the client's face and various identification items; the personalization process usually take place into a POS with the personalization;
- **Subscription** – the action of the registering and entering into a medium/long term billing arrangement for a service in a certain period of time; for instance, in a public transportation system, the client buys access for travelling with the buss number 300 between 15 May and 16 June this year.

SATS is inspired as terminology from the mobile applications environment and is designed for contactless integrated circuits cards with memory chip – Mifare.

2 Concepts of the Secure Automatic Ticketing System

The SATS handles pre-pay, post-pay and both types' clients. Also, SATS has the availability to works with e-Cards as well as with paper tickets. The clients who buy paper tickets are considered pre-pay clients.

For a better view about of the types of the clients it is recommended to see the definitions from first chapter of this paper and figure 1. In figure 1, the clients' types of SATS are presented:

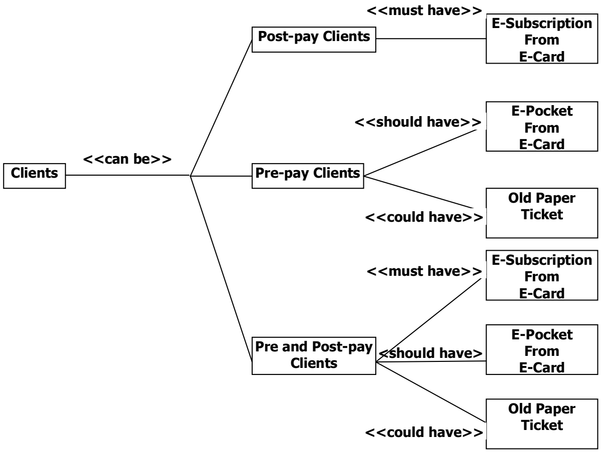


Fig.1 SATS Clients types

The ideal case all the SATS clients are not using old paper ticket. In order to validate the E-Subscription or E-Pocket, SATS uses E-Validation devices.

The SATS implements only one rating method for E-Subscription (post-pay clients). The rating method is based on checking the E-Subscription against the E-Validation Device.

The SATS implements two rating methods for E-Pocket (pre-pay clients):

- *Fixed* rate – the rate for each access to the service will be the same; for instance, in a public transportation system, the client is paying only a fixed amount no matter the station he/she is going to get in or out;
- *Variable* rate – the client is paying differentiated by the achieved service level; for instance, in a public transportation system, the client is paying a variable amount of money, depending the bus line and the area of the journey.

The priority of the E-Validation device for checking an E-Card is first to checkout if it is an E-Subscription area. After that, it checks the existence of the E-Pocket area.

The E-Pocket allows various services such as:

- The payment for any journey payment in a public transportation system;
- The payment for more than one client journey from a single card;
- The payment of other scalable systems such as:
  - Parking tickets;
  - Zoo, Cinema, Museums tickets;
  - Loyalty cards for the gym access;
- The support for clearing activities via a Clearing Authority.

The E-Card could be use also by other persons involved in a SATS. For example, in a public transportation system, E-Cards could be used also by the following categories of personnel:

- The clients: pre-pay, post-pay or both;
- The drivers: for authentication purposes and for working hours checking;
- The inspectors from the bus lines;
- The operators from the POS.

The suggested solution for E-Card is Philips Mifare memory card detailed in the below section.

3 Philips Mifare Memory Cards

The MIFARE Classic card is fundamentally just a memory storage device, where the memory is divided into segments and blocks with simple security mechanisms for access control.

They are suitable for: Public transportation; Access control; Event ticketing; Gaming and Identity. Figure 2 shows how the contact-less cards transfer data with the reader using Radio Frequency and in the same time they are charging with electrical power in order to perform various computing actions.

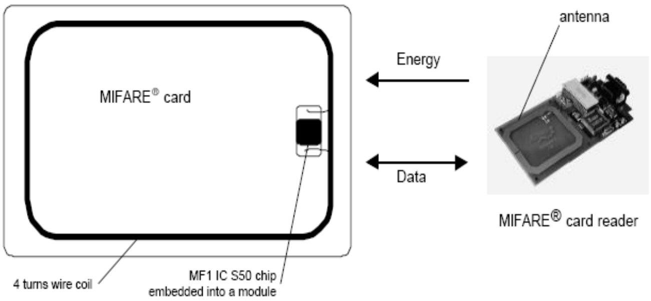


Fig.2 Mifare Card Communications [1]

The Philips Mifare card structure is shown in figure 3. As functional modules the Mifare cards have an RF Interface which is able to handle with RF – Radio Frequency – ISO/IEC 14443 specifications.

There is an Anti-collision module which is responsible with interacting with one reader but with more than one Mifare cards in the proximity area. Authentication module helps the developer to establish a set of keys for enabling only the applications that "knows" the key to be able to read/write data in EEPROM through EEPROM interface.

There is also Crypto module which is encrypting the data before going through RF interface.

Because recently the Mifare encryption of data from the card to the reader has been broken it is recommended to encrypt the data in computer application and to store the encrypted data into the memory card.

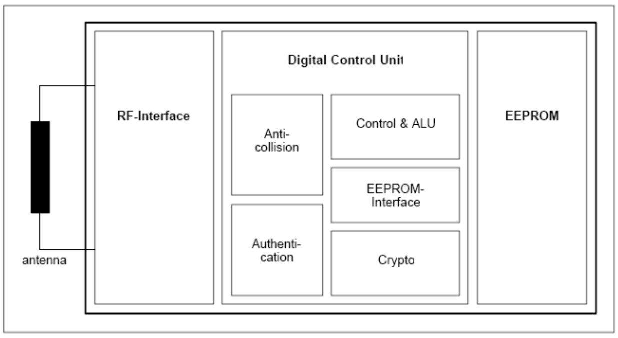


Fig.3 Mifare Card Structure [1]

The authentication and card access procedure is presented in figure 4:

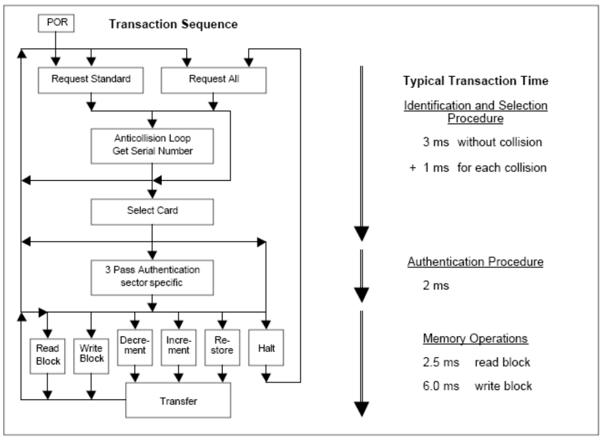


Fig.4 Card Access and Authentication Procedure [1]

The security is achieved through three mechanisms:

- Mutual three pass authentication (ISO/IEC DIS 9798-2) – after selection of a card, the reader specifies the memory location of the following memory access and uses the corresponding key for the three pass authentication procedure; after a successful authentication all memory operations are encrypted;
- Individual set of two keys per sector (per application) to support multi-application with key hierarchy;
- Unique serial number for each device.

It is recommended to encrypt the content of data in Mifare cards because the three mechanisms could have some issues regarding data security.

The mechanism how is ensured data integrity, three pass authentication sequence and memory operations are more intelligible after the memory organization is presented in figure 5.

The Philips Mifare Cards 1K have 1024 bytes of EEPROM memory. The 1024 x 8 bit EEPROM memory is organized in 16 sectors with 4 blocks of 16 bytes each. In the erased state the EEPROM cells are read as a logical "0", in the written state as a logical "1". In the memory blocks cannot be read/written a value as long as the authentication process is not successful.

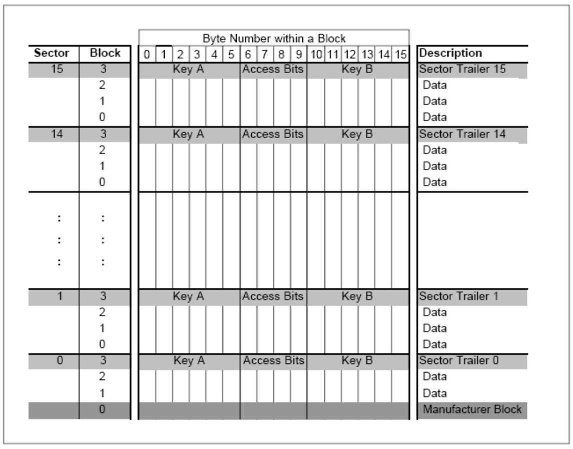


Fig.5 Memory Chip Pattern [1]

The first data block (block 0) of the first sector (sector 0) from figure 5 is “Manufacturer block”. The “Manufacturer block” is presented in figure 6.

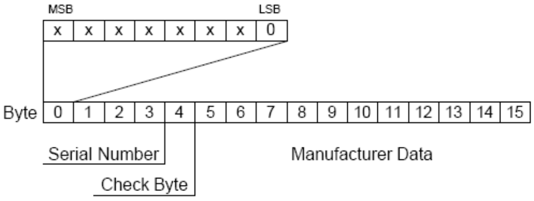


Fig.6 Manufacturer block [1]

The “manufacturer block” has 16 bytes (128 bits) and it contains the IC manufacturer data: Unique Serial Number 4 bytes, Check Byte and Manufacturer Data. Because of the security and system requirements, this block can be only read after having been programmed by the ICC – Integrated Circuits Card manufacturer at production. After the “Manufacturer Block” there are “Data Blocks” or “Trailer Blocks” organized in sectors.

All the sectors, not including the first one, have (4 blocks * 16 bytes):

- 3 “data blocks”
- 1 “trailer blocks”.

The “data block” can be can be configured by the access bits from the sector trailer block:

- to store/read binary data blocks (e.g. binary representation of ASCII data);
- value blocks (e.g. electronic purse applications), where there are additional commands like increment and decrement for direct control of the stored value.

An authentication command has to be submitted by the reader before any memory operation in order to allow further commands. A value block is represented in figure 7:

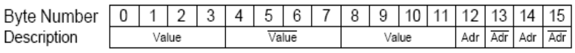


Fig.7 Value block [1]

The value blocks allow performing minimal mathematic functions as increment or decrement. The valid commands for value block are: read, write, increment, decrement, restore, transfer. A value block can only be generated through a write operation in the value block respecting the format from figure 8:

- Value – it is a signed 4-byte value; the lowest significant byte of a value is stored in the lowest address byte;
- Negative values are stored in standard 2´s complement format; for reasons of data integrity and security, a value is stored three times, twice non-inverted and once inverted;
- Adr – it is a 1-byte address, which can be used to save the storage address of a block, when implementing a powerful backup management; the address byte is stored four times, twice inverted and non-inverted; during increment, decrement, restore and transfer operations the address remains unchanged; it can only be altered via a write command.

For instance, if a program writes the value 5 in the value block, then the value block will have the following hex values (little endian bytes order and big endian bits order):

```
0x05 0x00 0x00 0x00|0xFA 0xFF 0xFF 0xFF|0x05 0x00 0x00 0x00|00FF00FF
```

The sector “trailer block” is presented in figure 8:

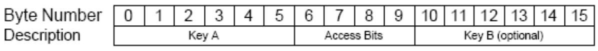


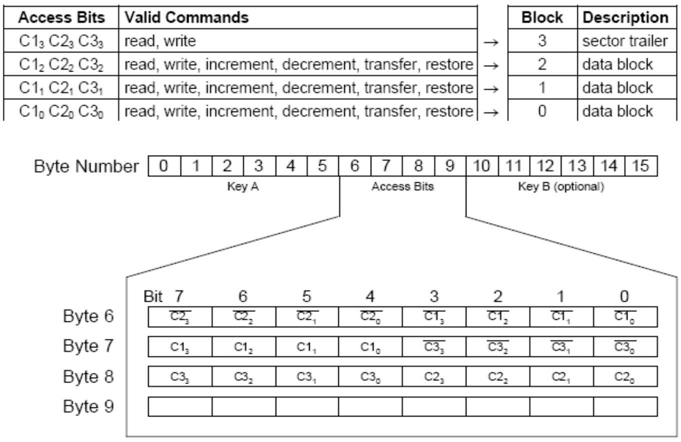
Fig.8 Sector trailer block [1]

Each sector has a sector trailer containing the:

- Secret keys A and B (B is optional), which return logical “0”s when read;
- The access conditions for the four blocks of that sector, which are stored in bytes 6...9.

The access bits also specify the type (read/write or value) of the data blocks. If key B is not needed, the last 6 bytes of block 3 can be used as data bytes. Byte 9 of the sector trailer is available for user data. For this byte apply the same access rights as for byte 6, 7 and 8.

The access conditions for every data block and sector trailer are defined by 3 bits, which are stored non-inverted and inverted in the sector trailer of the specified sector figure 10. The access bits manage the rights of the memory access using the secret keys A and B. The access conditions may be modified but with each memory access the internal logic verifies the format of the access conditions. If it detects a format violation the whole sector is blocked irreversible. In the description from figure 9, the access bits are mentioned in the non-inverted mode only. The internal logic of the Mifare ICC ensures that the commands are executed only after an authentication procedure or never.



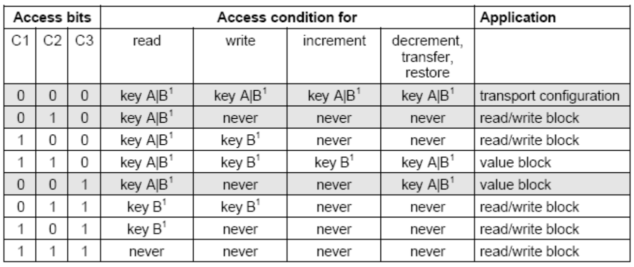
Access Bits	Valid Commands		Block	Description
$C1_3$ $C2_3$ $C3_3$	read, write	→	3	sector trailer
$C1_2$ $C2_2$ $C3_2$	read, write, increment, decrement, transfer, restore	→	2	data block
$C1_1$ $C2_1$ $C3_1$	read, write, increment, decrement, transfer, restore	→	1	data block
$C1_0$ $C2_0$ $C3_0$	read, write, increment, decrement, transfer, restore	→	0	data block

Byte Number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Key A						Access Bits				Key B (optional)					

Bit	7	6	5	4	3	2	1	0
Byte 6	$\overline{C2_3}$	$\overline{C2_2}$	$\overline{C2_1}$	$\overline{C2_0}$	$\overline{C1_3}$	$\overline{C1_2}$	$\overline{C1_1}$	$\overline{C1_0}$
Byte 7	$C1_3$	$C1_2$	$C1_1$	$C1_0$	$\overline{C3_3}$	$\overline{C3_2}$	$\overline{C3_1}$	$\overline{C3_0}$
Byte 8	$C3_3$	$C3_2$	$C3_1$	$C3_0$	$C2_3$	$C2_2$	$C2_1$	$C2_0$
Byte 9								

Fig.9 Access Bits [1]

The figure 10 specifies the access bits conditions for data blocks in value mode ($C1_i$, $C2_i$, $C3_i$ where i = 0,1,2).



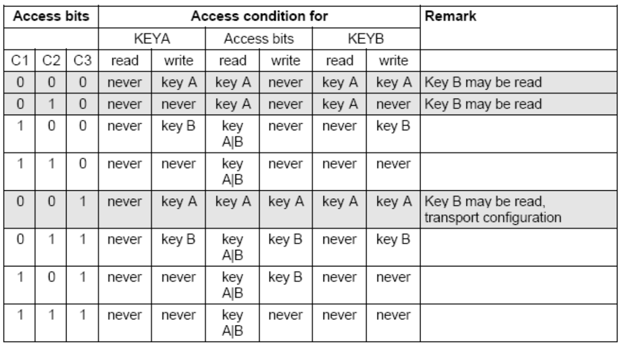
Access bits			Access condition for				Application
C1	C2	C3	read	write	increment	decrement, transfer, restore	
0	0	0	key A|B[1]	key A|B[1]	key A|B[1]	key A|B[1]	transport configuration
0	1	0	key A|B[1]	never	never	never	read/write block
1	0	0	key A|B[1]	key B[1]	never	never	read/write block
1	1	0	key A|B[1]	key B[1]	key B[1]	key A|B[1]	value block
0	0	1	key A|B[1]	never	never	key A|B[1]	value block
0	1	1	key B[1]	key B[1]	never	never	read/write block
1	0	1	key B[1]	never	never	never	read/write block
1	1	1	never	never	never	never	read/write block

Fig.10 Access Bits combination for value block access [1]

Depending on the access bits for data blocks (blocks 0...2) the read/write access is specified as 'never', 'key A', 'key B' or 'key A|B' (key A or key B). The setting of the relevant access bits defines the application and the corresponding applicable commands as it follows:

- Read/write block – the operations read and write are allowed;
- Value block – allows the additional value operations increment, decrement, transfer and restore; in one case ('001') only read and decrement are possible for a non-rechargeable card; in the other case ('110') recharging is possible by using key B;
- Manufacturer block – the read-only condition is not affected by the access bits setting;
- Key management – in transport configuration key A must be used for authentication.

Figure 11 specifies the access bits conditions for trailer blocks ($C1_3$, $C2_3$, $C3_3$).



Access bits			Access condition for						Remark
			KEYA		Access bits		KEYB		
C1	C2	C3	read	write	read	write	read	write	
0	0	0	never	key A	key A	never	key A	key A	Key B may be read
0	1	0	never	never	key A	never	key A	never	Key B may be read
1	0	0	never	key B	key A|B	never	never	key B	
1	1	0	never	never	key A|B	never	never	never	
0	0	1	never	key A	key A	key A	key A	key A	Key B may be read, transport configuration
0	1	1	never	key B	key A|B	key B	never	key B	
1	0	1	never	never	key A|B	key B	never	never	
1	1	1	never	never	key A|B	never	never	never	

Fig.11 Access Bits combinations for Key Access [1]

Depending on the access bits for the sector trailer (block 3 from each sector) the read/write access to the keys and the access bits is specified as 'never', 'key A', 'key B' or key A|B' (key A or key B).

For instance, if after personalization the Mifare 1k card will be used for writing and reading mode, the values for access bits it will be:

$C1_3$, $C2_3$, $C3_3$ - trailer block -> 1 0 0 Byte6 = 0xFF = 1111 1111

$C1_2$, $C2_2$, $C3_2$ - data block -> 1 1 0 Byte7 = 0x07 = 0000 0111

=>

$C1_1$, $C2_1$, $C3_1$ - data block -> 1 1 0 Byte8 = 0x80 = 1000 0000

$C1_0$, $C2_0$, $C3_0$ - data block -> 1 1 0 Byte9 = 0x00 = 0000 0000

So the sector trailer which have KeyA = 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF and keyB = 0x00 0x00 0x00 0x00 0x00 0x00 (default from manufacture) it will have the following structure:

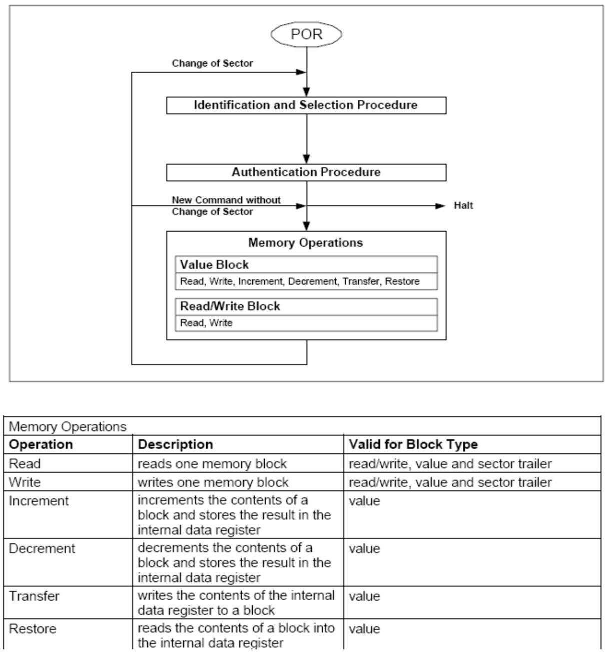
FF FF FF FF FF FF | FF 07 80 00 | 00 00 00 00 00 00

The three pass authentication sequence takes place whenever the reader wants to store or read data from a specific block. The authentication procedure:

1. The reader specifies the sector to be accessed and chooses key A or B;
2. The card reads the secret key and the access conditions from the sector trailer; then the card sends a random number as the challenge to the reader (pass one);
3. The reader calculates the response using the secret key and additional input; the response, together with a random challenge from the reader, is then transmitted to the card (pass two);
4. The card verifies the response of the reader by comparing it with its own challenge and then it calculates the response to the challenge and transmits it (pass three);
5. The reader verifies the response of the card by comparing it to its own challenge; after transmission of the first random challenge the communication between card and reader is encrypted.

Figure 12 shows the memory access for various sectors and the three pass authentication procedure place for reading/writing data blocks in value or binary mode.

After the three pass authentication procedure, any of the following operations may be performed: reading/writing binary blocks; decrementing the contents of a block and stores the result in a temporary internal data-register; incrementing the contents of a block and stores the result in the data-register; restoring the contents of a block into the data-register or transferring the contents of the temporary internal data-register to a value block.


POR
Change of Sector
Identification and Selection Procedure
Authentication Procedure
New Command without Change of Sector
Halt
Memory Operations
Value Block
Read, Write, Increment, Decrement, Transfer, Restore
Read/Write Block
Read, Write

Memory Operations		
Operation	**Description**	**Valid for Block Type**
Read	reads one memory block	read/write, value and sector trailer
Write	writes one memory block	read/write, value and sector trailer
Increment	increments the contents of a block and stores the result in the internal data register	value
Decrement	decrements the contents of a block and stores the result in the internal data register	value
Transfer	writes the contents of the internal data register to a block	value
Restore	reads the contents of a block into the internal data register	value

Fig.12 Memory Access [1]

The following mechanisms are implemented in the contact-less communication link between reader and card in order to ensure data integrity in RF transmission:

- 16 bits CRC per block;
- Parity bits for each byte;
- Bit count checking;
- Bit coding to distinguish between "1", "0", and no information;
- Channel monitoring (protocol sequence and bit stream analysis);

The Philips Mifare cards are very good for storing sensitive data but in encrypted mode because there are mechanism how to break Mifare data security over RF communication. These types of cards are considered ICC cards and are not used as 100% smart cards because they do not provide API for any kind of processing except incrementing/decrementing values. They are very strong linked with the data structures used by the host application.

4 The SATS architecture

Characteristics of distributed information systems are highlighted in [3], [8], [10] and [11].

The overview architecture is presented in figure 13. The components are described in following sections.

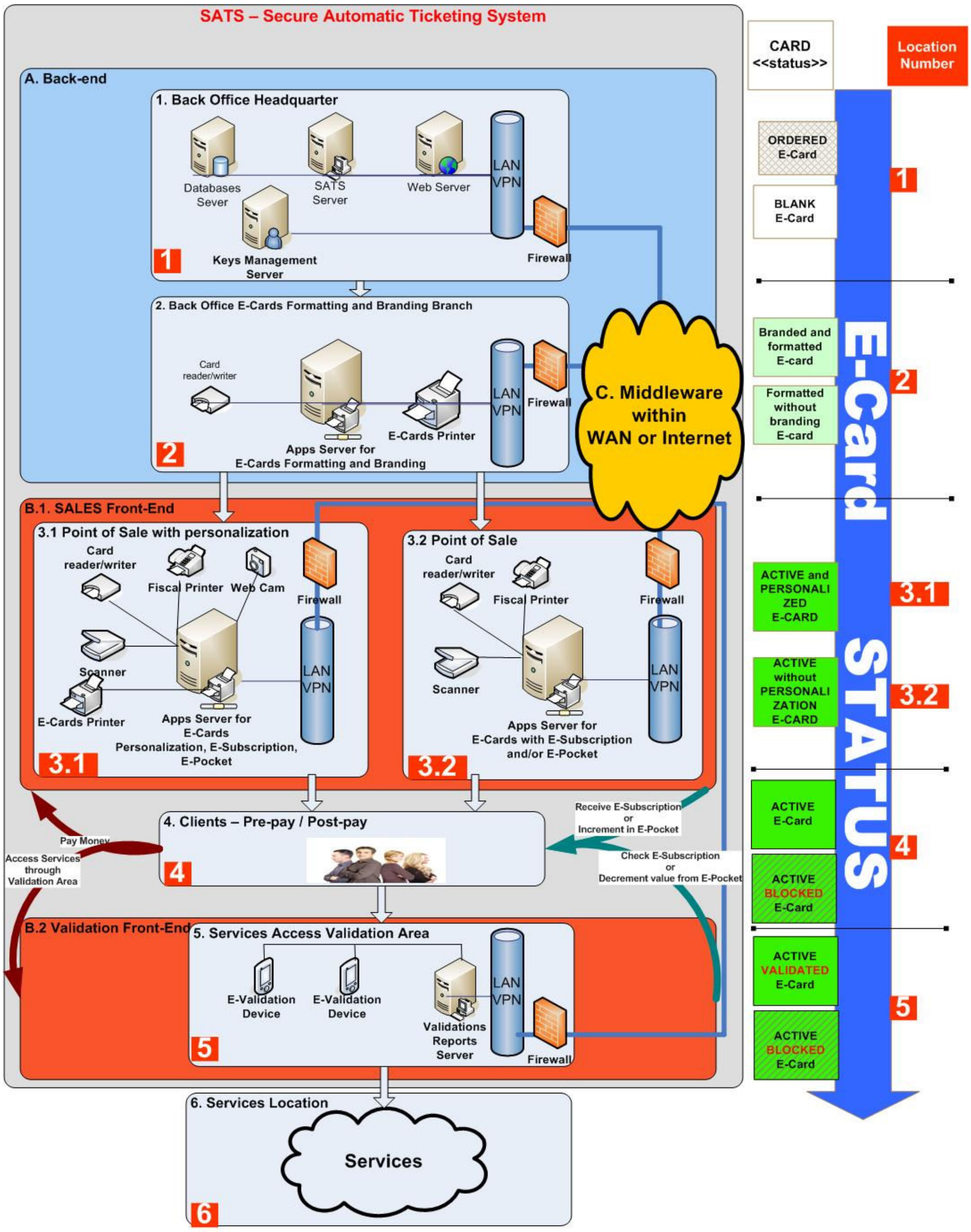

SATS – Secure Automatic Ticketing System
A. Back-end
1. Back Office Headquarter
Databases Sever
SATS Server
Web Server
LAN VPN
Keys Management Server
Firewall
1
2. Back Office E-Cards Formatting and Branding Branch
Card reader/writer
LAN VPN
Firewall
E-Cards Printer
Apps Server for E-Cards Formatting and Branding
2
C. Middleware within WAN or Internet
B.1. SALES Front-End
3.1 Point of Sale with personalization
Card reader/writer
Fiscal Printer
Web Cam
Firewall
Scanner
E-Cards Printer
Apps Server for E-Cards Personalization, E-Subscription, E-Pocket
LAN VPN
3.1
3.2 Point of Sale
Card reader/writer
Fiscal Printer
Firewall
Scanner
Apps Server for E-Cards with E-Subscription and/or E-Pocket
LAN VPN
3.2
4. Clients – Pre-pay / Post-pay
4
Pay Money
Access Services through Validation Area
Receive E-Subscription or Increment in E-Pocket
Check E-Subscription or Decrement value from E-Pocket
B.2 Validation Front-End
5. Services Access Validation Area
E-Validation Device
E-Validation Device
Validations Reports Server
LAN VPN
Firewall
5
6. Services Location
Services
6
CARD <<status>>
Location Number
ORDERED E-Card
BLANK E-Card
1
Branded and formatted E-card
Formatted without branding E-card
2
ACTIVE and PERSONALIZED E-CARD
3.1
ACTIVE without PERSONALIZATION E-CARD
3.2
ACTIVE E-Card
ACTIVE BLOCKED E-Card
4
ACTIVE VALIDATED E-Card
ACTIVE BLOCKED E-Card
5
E-Card STATUS

Fig.13 The overview of the SATS

The architecture of SATS has three layers. Each layer is responsible for certain functions and in each layer location the E-Card has a status in terms of life-cycle:

A. *Back-end layer* – it is responsible with business logic of SATS; it contains the following elements:

- *Back Office Headquarter (location 1)* – it receives the ordered cards from the supplier, and it provides the cards to the *Back Office E-Cards Formatting and Branding*; it contains the following components:
    - ○ *SATS Server* – runs applications which are receiving data from the middleware and from the Front-end applications via Web Server; it has some enterprise software components for increasing the distributed processing and scalability of the SATS;

  - *Key Management Server* – is used for generating access key for E-Cards, X509 certificates and for storing the keys; it could be assimilated with a Certificate and Registration Authority;
  - *Web Server* – contains the web applications and web services; most of the applications from Front-End are web based or they access web services;
  - *Database Server* – contains the database of the clients (pre-pay, post-pay or both), the validation device data and other kind of involved personnel data;
- *Back Office E-Cards Formatting and Branding (location 2)* – it receives the blank cards from the *Back Office Headquarter* and it formats the cards with special keys for access; a formatted E-Card may be branded with the commercial signs required by the SATS beneficiary; location 2 has the following components:
  - *Apps for E-Cards Formatting and Branding Server* – runs applications which are able to handle to format and brand the E-Cards;
  - *Card Reader/Writer for E-Card formatting* – is initializing the E-Cards with the access keys;
  - *Card Printer for E-Card branding* – prints on both faces of the plastic over E-Card various commercial signs or advertising.

B. *Front-end layer* – it is responsible about the interface with the clients; it has the following components:

- *Sales Front-end (location B.1)* – the place where the E-Cards are sold to the client; location contains:
  - *POS with Personalization (location 3.1)* – it receives the formatted and/or branded cards in order to be personalized with the proper data structures and eventually with the client picture and identification number;
  - *POS without Personalization (location 3.2)* – it receives the formatted and/or branded cards in order to be personalized with the proper data structures;
- *Validation Front-end (location B.2)* – is the place where the E-Cards are validated before the client to get access to the services or system's resources; it contains *Services Access Validation Area (location 5)* that includes the E-Validation devices which are decrementing or checking out the access rights of the clients;

C. *Middleware layer* – it is responsible with secure communication between Front-End and Back-End.

The security is ensured on the most important layers in TCP/IP stack:

- Application layer through Key Management Server and through encryption of data structure used for validation purposes by E-Card and E-Validation devices; E-Validation devices have also a Secure Application Module – SAM which is a smart card and it used for generating the session keys and the digital signatures.
- Transport layer through Secure Socket Layer – SSL with implications at application layer of using Hyper Text Transfer Protocol Secure - HTTPS for the web services access from POS locations;
- Internetworking layer through using Internet Protocol Secure – IPSec and Virtual Private Network – VPN in order to minimize the risks of eaves dropping and to support data integrity and confidentiality.

The security policies are implemented at firewall level for each location in order to filter all network packets and to accept only the proper network packets.

In right side of the figure 13 are the card's status in various locations. The architecture from figure 13 is generic and could be implemented in any kind of information system based on access cards (E-Cards). Of course, there are particularities for each system and therefore it is recommended to analyze very well the specific area of beneficiary system in order to achieve the best results.

The personal, charging, wallet, subscription and last usage data is in ASN.1 and stored encrypted information of ASN.1 DER. The choice of ASN.1 DER is made because of memory limitations of the RFID cards and the overload of information using XML for data structuring is too big.

The process of validation the information from the card consits of few steps:

- The CAD – reader/validator reads the RSA encrypted session key;
- Only the validator has the company's RSA private key stored into a SAM – Secure Access Module; Therefore only the validator is able to obtain the session key in clear;
- The session key decrypts critical information for validation process such as wallet and subscription;
- The validator checks out the information and grand or not the clients' access.

A sample of ASN.1 of the personal and wallet data structure is:

```
  PersonalData ::= SEQUENCE {
        ID_Citizen OCTET STRING,
        First_Name OCTET STRING,
        Last_Name  OCTET STRING,
        Address    OCTET STRING,
        Occupation OCTET STRING
  }

  WalletData  ::= SEQUENCE OF {
    Credit   OCTET STRING, -- 32 bits float
    Currency   IA5String -- 3 UTF-16 chars
                      --in little endian=6 bytes
  }
```

The brute force attack of RFID Mifare cards is almost impossible because of its authentication process with keys lengh of 6 bytes, 48 bits. But, the authentication and encryption algorithm is used at application level because the MIFARE communication protocol between the reader and RFID cards has been broken. The attack is described in [9]. The datails of the memory layout of Mifare cards are in [7].

5 Conclusions

The security issue of a secure automatic ticketing system is a very important one because the payments in such systems must be safe and the client must safely keep its money and personal data in the system.

The security of the suggested architecture is assured on different levels: application (session keys, the digital signatures), transport (SSL, HTTPS) and internetworking (IPSec). The security policies are implemented at firewall level for each location to accept only the proper network packets.

The security should be focused also in the architectures and structures from the back-ofice including the following sub-systems:

- The subsystem in charge with formatting and preloading the E-Cards.
- The subsystem in charge with customer relationship management.
- The subsystem in charge with managing the operator's E-cards. (bus drivers, bus validation operators)
- The subsystem in charge with the pricing scheme management.
- The subsystem in charge with customer's E-cards management.
- The subsystem in charge with all system's equipment.
- The supervising management subsystem.
- The reporting and statistics subsystem.

Regarding the impact on distributed informatics systems (mobile business solutions, mobile services), the authors are continuing this research in an important research contract with code 1838/2008, contract no. 923/2009 and the title *Implementation of the Quantitative Methods in Distributed Informatics System Audit*. The research contract is financed by The National University Research Council – Ministry of Education, Research and Innovation from Romania.

References:

[1]http://www.nxp.com/acrobat_download/other/identification/M001053_MF1ICS50_rev5_3.pdf

[2] W. Stallings, *Cryptography and Network Security, 3/E*, Prentice Hall, 2003

[3] P. Pocatilu, M. Vetrici, *Schedule Risk Management for Business M-Applications Development Projects*, WSEAS Transactions on Computers, vol. 8, April 2009, pp. 735 – 745

[4] D. Stinson, *Cryptography – Theory and Practice*, 2nd Edition, Chapman & Hall/Crc Publishing House, New York, 2002

[5] C. Toma, *Security in Software Distributed Platforms*, ASE Publishing House, Bucharest, 2008

[6] C. Toma, C. Boja, M. Popa, Solution for Non-Repudiation in GSM WAP Applications, *The 7th WSEAS International Conference on SOFTWARE ENGINEERING, PARALLEL and DISTRIBUTED SYSTEMS (SEPADS '08)*, Advances on Software Engineering, Parallel and Distributed Systems, University of Cambridge, UK, February 20-22, 2008, pp. 212 – 219

[7] C. Toma, *Secure Authentication and Encryption Scheme for E-Ticketing System, JITCS – Journal of IT&C Security, SECITC 2009, www.secitc.eu.*

[8] C. Boja, L. Batagan, *Analysis of M-Learning Applications Quality*, WSEAS Transactions on Computers, vol. 8, May 2009, pp. 767 – 777

[9] G. de Koning Gans, J.H. Hoepman, and F. D. Garcia, *A Practical Attack on the MIFARE Classic*

[10] C. Toma, M. Popa, C. Boja, *Smart Card Based Solution for Non-Repudiation in GSM WAP Applications*, WSEAS Transactions on Computers, vol. 7, Issue 5, May 2008, pg. 453 – 462

[11] M. Popa, C. Toma, *Secure Automatic Ticketing System*, Recent Advances on Data Networks, Communications, Computers*, proceedings of the8th WSEAS International Conference on Data Networks, Communications, Computers (DNCOCO'09),* 7-9 November 2009, Morgan State University, Baltimore, USA, pp. 116 – 123

Identity based Threshold Cryptography and Blind Signatures for Electronic Voting

GINA GALLEGOS-GARCÍA[1], ROBERTO GÓMEZ-CÁRDENAS[2], GONZALO I. DUCHÉN-SÁNCHEZ[1]
[1]Graduate and Research Section, [2]Department of Computer Science
[1]Instituto Politécnico Nacional, [2]Instituto Tecnológico de Estudios Superiores de Monterrey-CEM
[1] Av. Sta. Ana 1000, Sn. Fco. Culhuacan, 04430, Coyoacán
[1]Mexico City
[2] Carretera Lago de Guadalupe Km 3.5, Atizapán de Zaragoza, 52926
[2]Mexico State
gina@calmecac.esimecu.ipn.mx

Abstract: - Recently, there has been an increasing interest to improve the efficiency in election processes which has brought as a consequence a wide range of proposals for electronic voting. Electronic voting protocols are a reasonable alternative to conventional elections. Nevertheless, they are facing an evolution due to its requirements, especially the ones needed to provide full security considered to represent a democratic electronic vote. In the literature, different protocols based on public key schemes have been proposed to meet such security requirements. In this paper, we propose the use of bilinear pairings in order to provide the security requirements that an electronic voting protocol must meet, without requiring the entire infrastructure needed in a public key scheme. Proposed protocol considers two cryptographic primitives as main building blocks: threshold and blind signature schemes. It is divided in four main stages: set-up, authentication, voting and counting. Moreover, it meets privacy, accuracy and robustness by using bilinear pairings. We make a comparative analysis, which is based on its performance and the key pairs, Trust and Certification Authorities it requires.

Key-Words: - Bilinear pairings, Blind signatures, Electronic voting protocols, Identity based cryptography, Public key cryptography, Security requirements, Threshold cryptography.

1 Introduction

Electronic voting, as an e-government issue [1], has been mentioned in different media as the use of computers or computerized voting equipment to cast ballots in an election since 1964, which nowadays is a reasonable alternative to conventional elections and other opinion expressing processes. However, it must offer at least as same benefits as a conventional election does, in addition to reduce monetary costs.

Generally speaking an electronic voting protocol involves three main entities: voter, registration authorities and tallying authorities. The voter is an entity who has the right for voting. The registration authorities register voters before the election´s day and they also ensure only registered voters will be able to vote. The tallying authorities ensure cast votes are counted.

All those actors each other interact during three main phases: registration, voting and counting. In the registration phase, a citizen must be registered as an authenticated voter. In the voting phase, only authenticated voters cast their votes. Finally in the counting phase, performed by tallying authorities or a special center, cast votes are counted and tally is published.

In order to use an electronic voting protocol inside an electronic voting process, it must meet at least seven security requirements: privacy, eligibility, uniqueness, uncoercibility, transparency, accuracy and robustness.

Privacy: A vote must be kept secret from any coalition of authorities.

Eligibility: Only registered voters, who meet certain pre-determined criterion, must are eligible to vote.

Uniqueness: Only one vote for a voter must be counted.

Uncoercibility: Any coercers, even authorities, must be able to coercer a voter to cast its vote in a particular way.

Transparency: The whole voting protocol must be transparent, since the beginning until the end.

Accuracy: The votes should be correctly recorded and tallied.