The System for Secure Mobile Payment Transactions

Master Thesis in Information and Communication Systems Security

Behzad Pouralinazar

Stockholm, Sweden 2013
TRITA-ICT-EX-2013:6
The System for Secure Mobile Payment Transactions

Behzad Puralinazar

Abstract

Secure mobile payment methods are typically needed in mobile access channels, it is vital to apply them when using mobile payment applications, which have arisen as a result of the increase of e-commerce. Currently, there are some payment methods that most of them can be used when conducting transactions but, they differ in terms of functionality, usability, costs or security.

What makes mobile payment system more useful and necessary are the features and functionalities which are essentially based on customer needs that may differ in different market places, social categories and regions. In addition to payment system functionality, what mainly matters is securing payment methods with sufficient security protections.

In this thesis, it is been tried to introduce some features currently lacking in mobile payment systems to propose a security architecture and then to present a solution for prospective problems with some current mobile payment systems and applications. The solution mainly implements what has been proposed and required to fulfill system requirements.

The proposed security architecture supports already designed mobile payment system (SAFE) which typically interacts with the merchant and the customer for performing secure payment transactions. Mobile payment security architecture is designed, implemented and analyzed to be highly scalable and modular in addition to providing the desired security requirements set to it in this thesis work. The architecture is enhanced with mobile applications, so called SAFE apps, enabling securely authorizing e-commerce transactions.

This report begins with introducing mobile payment system technologies, concept and architecture as the foundation for this work. The systems relevant to the one proposed in this thesis are briefly discussed in terms of functionality and security issues. System architecture and design of SAFE Mobile Payment System are briefly discussed as the core component of the settling e-commerce functions in a holistic view. The functionality and security issues of the system design and implementation are presented and finally the work is analyzed to determine how the project objectives were met. The future development and work are finally summarized in the conclusions chapter.
Acknowledgement

This thesis work would not have been possible without the guidance and the help of several individuals who contributed and extended their valuable assistance in the completion of this study. I must express my first and foremost gratitude to Prof. Sead Muftic, whose persistence, encouragement and patience I will never forget. His guidance and assistance was a considerable inspiration in facing various obstacles in completion of this thesis work.

In my daily work I have been blessed with a friendly and cheerful group of PhD students: Abdul Ghafoor Abbasi, Aron Kondoro, Matei Ciobanu Morogan and Feng Zhang as well as many of my friends in Information and Communications Systems Security program, have provided good arguments in different areas from technical assistance up to some sort of project management assistance.

Also, my special thanks to my dear friend Saman Abbasian who was a good companion and advisor on exhausting and disappointing situations in my personal life during this work.

Last but not least, exceptional thanks to my parents for supporting me throughout all my studies from the first year up to graduate studies and for giving me the strength to pass over most obstacles so far.
Table of Contents

1. Introduction .................................................................................................................. 6
   1.1 Research Problems ................................................................................................. 7
   1.2 Purpose ..................................................................................................................... 8
   1.3 Delimitations ............................................................................................................ 9
   1.4 Methodology .......................................................................................................... 9
       1.4.1 Methodology for Establishing System Requirements ........................................ 9
       1.4.2 Methodology for Design and Implementation .................................................. 10
       1.4.3 Methodology for Data Acquisition .................................................................. 11
       1.4.4 Methodology for Evaluating the Results ......................................................... 11

2. Background and Basic Concepts ............................................................................... 12
   2.1 Basic Concepts ....................................................................................................... 12
       2.1.1 Mobile Payment .............................................................................................. 12
       2.1.2 Mobile Payment Models ................................................................................ 13
       2.1.3 Mobile Payment Transactions ....................................................................... 14
       2.1.4 Mobile Payment Applications ....................................................................... 14
   2.2 Related Work ......................................................................................................... 15
       2.2.1 Architecture for Secure Two-Party Mobile Payment ....................................... 16
       2.2.2 A Lightweight and Secure Protocol for Mobile Payments via Wireless Internet in m-commerce ................................................................. 18
       2.2.3 Secure Mechanism based on Concurrent Signature for Mobile Payment Services 20

3. System Design and Architecture .............................................................................. 22
   3.1 Functionality .......................................................................................................... 22
   3.2 Security ................................................................................................................... 23
   3.3 System Architecture .............................................................................................. 24
   3.4 The Purpose and related Work ............................................................................. 25
   3.5 System Components .............................................................................................. 26
       3.5.1 SAFE System .................................................................................................. 26
       3.5.1.1 Mobile Banking .......................................................................................... 27
       3.5.1.2 Mobile Shopping ....................................................................................... 29
       3.5.2 Mobile Applications ....................................................................................... 31
       3.5.3 Security Components and Architecture ............................................................. 32
       3.5.3.1 Client Security Module .............................................................................. 35
       3.5.3.2 IDMS server ................................................................................................ 36
       3.5.3.3 CA server .................................................................................................... 36
3.5.3.4 Location-based Authentication (LBA) ......................................................... 37

4. Implementation ......................................................................................... 41

   4.1 The Environment, Platforms and Tools .................................................. 41
   4.1.1 Hardware and Software for Development ........................................... 41
   4.1.2 Android SDK ..................................................................................... 41
   4.1.3 SAFE Administration Station ............................................................... 42
       4.1.3.1 Registration of Entities ................................................................. 42
       4.1.3.2 Transactions Management ............................................................ 43
   4.1.4 SAFE Mobile Applications .................................................................... 43
       4.1.4.1 Digital Certificates Management .................................................. 49
           4.1.4.1.1 Key pair generation ................................................................. 50
       4.1.4.2 Secure Element ............................................................................. 53
       4.1.4.3 SAFE Agent .................................................................................. 53
       4.1.4.4 SAFE Wallet ................................................................................ 54
       4.1.4.5 SAFE Merchant ............................................................................... 59
       4.1.4.6 Secure Mobile Transactions ........................................................... 62
           4.1.4.6.1 Over-The-Counter Payment Transaction .................................... 64
           4.1.4.6.2 Over-The-Air Payment Transaction ........................................... 69

5. Analysis and Discussions of the Solution ................................................. 71

   5.1 Meeting the Goals .................................................................................. 71
       5.1.1 Functional .......................................................................................... 72
       5.1.2 Technical ........................................................................................... 72
       5.1.3 Security ............................................................................................. 73
   5.2 Future Work and Development .............................................................. 73
List of Figures

FIGURE 1: CONCEPTUAL SCHEMA OF THE MOBILE PAYMENT SYSTEM [73] ................................................................. 12
FIGURE 2: ARCHITECTURE FOR SECURE TWO-PARTY MOBILE PAYMENT [26] ................................................................. 16
FIGURE 3: SECURITY MECHANISM FOR SECURE TWO-PARTY MOBILE PAYMENT [26] ................................................... 17
FIGURE 4: DIGITAL SIGNATURE KEY MANAGEMENT [26] ..................................................................................................... 18
FIGURE 5: FIRST METHOD WHICH CUSTOMERS USE FOR PAYMENT [27] ................................................................. 19
FIGURE 6: MESSAGE EXCHANGE BASED ON CONCURRENT SIGNATURE PAYMENT PROTOCOL [28] .................. 21
FIGURE 7: FUNCTIONAL MODEL OF A MOBILE TERMINAL DESIGNED FOR M-COMMERCE APPLICATIONS ......... 25
FIGURE 8: THE ARCHITECTURE FOR SECURE MOBILE BANKING ................................................................................. 28
FIGURE 9: SYSTEM SECURITY ARCHITECTURE ................................................................................................................. 35
FIGURE 10: LOCATION-BASED AUTHENTICATION MECHANISM WITHIN THE SYSTEM ARCHITECTURE ......... 38
FIGURE 11: LOCATION-BASED AUTHENTICATION AND AUTHORIZATION PROTOCOL [37] .................. 39
FIGURE 12: THE FUNCTIONS OF REGISTRATION OF DIFFERENT ENTITIES [32] .................................................. 42
FIGURE 13: SAFE MESSAGE FORMAT [32] ......................................................................................................................... 44
FIGURE 14: SAFE MESSAGE FLOW [32] ......................................................................................................................... 44
FIGURE 15: SAFE SECURITY OPTIONS .............................................................................................................................. 45
FIGURE 16: MESSAGE INTEGRITY PROTECTION FLOW [32] ....................................................................................... 46
FIGURE 17: MESSAGE CONFIDENTIALITY PROTECTION FLOW [32] ........................................................................... 47
FIGURE 18: MESSAGE CONFIDENTIALITY PROTECTION FLOW [32] ........................................................................... 47
FIGURE 19: MESSAGE CONFIDENTIALITY AND INTEGRITY PROTECTION FLOW [32] .............................................................. 48
FIGURE 20: SECURITY OF MESSAGES [32] .......................................................................................................................... 49
FIGURE 21: REQUESTING AND OBTAINING A CERTIFICATE FROM SAFE CA ................................................ 50
FIGURE 22: SELF-REGISTRATION SEQUENCE .............................................................................................................. 53
FIGURE 23: TRANSACTION CONFIRMATION RESULT .................................................................................................. 54
FIGURE 24: SAFE WALLET MAIN MENU ......................................................................................................................... 54
FIGURE 25: SAFE WALLET MOBILE PAYMENT FUNCTIONS .......................................................................................... 55
FIGURE 26: SAFE WALLET ACCOUNT AND TRANSACTIONS FUNCTIONS ......................................................................... 56
FIGURE 27: SAFE WALLET M-BANKING MENU .............................................................................................................. 57
FIGURE 28: SAFE WALLET M-CARDS AND M-CASH FUNCTIONS .................................................................................. 57
FIGURE 29: SAFE WALLET M-SHOPPING FUNCTIONS ..................................................................................................... 58
FIGURE 30: M-SECURITY FUNCTIONS .......................................................................................................................... 59
FIGURE 31: SAFE MERCHANT MAIN MENU ................................................................................................................... 59
FIGURE 32: SAFE M-MARKETING FUNCTIONS .............................................................................................................. 60
FIGURE 33: SAFE M-BUSINESS FUNCTIONS ................................................................................................................ 61
FIGURE 34: SAFE M-POS FUNCTIONS .......................................................................................................................... 61
FIGURE 35: MOBILE CLIENT APPLICATION STRUCTURE .................................................................................................. 62
FIGURE 36: SMARTCARD API ARCHITECTURE [43] ............................................................................................................ 63
FIGURE 37: BLUETOOTH PERMISSION REQUEST .......................................................................................................... 65
FIGURE 38: BLUETOOTH DISCOVERING .......................................................................................................................... 65
FIGURE 39: DISABLED PAYMENT FORM ......................................................................................................................... 65
FIGURE 40: ENABLED PAYMENT FORM .......................................................................................................................... 66
FIGURE 41: FILLING IN THE PAYMENT FORM BY MERCHANT .................................................................................... 66
FIGURE 42: APPROVING PAYMENT ............................................................................................................................... 66
FIGURE 43: FILLING IN THE PAYMENT FORM BY CUSTOMER .......................................................................................... 67
FIGURE 44: TRANSACTION RESULT ............................................................................................................................... 67
FIGURE 45: DELIVERING TRANSACTION RESULT ........................................................................................................... 67
FIGURE 46: ASK FOR RECEIPT ............................................................................................................................................ 68
FIGURE 47: APPROVING THE RECEIPT ............................................................................................................................ 68
FIGURE 48: SENDING THE RECEIPT ............................................................................................................................... 68
FIGURE 49: FILLING IN THE PAYMENT FORM BY MERCHANT .......................................................................................... 69
FIGURE 50: TRANSACTION RESULT ............................................................................................................................... 70
1. Introduction

Recent developments of communications technologies and business models raised concerns about mobile payment systems in terms of usability and security. Rising smart mobile devices with variety of usage and privacy and easy access to communication protocols have provided the potentials for growing development of mobile commerce. Furthermore, new business models in daily activities have increased the need of comprehensive mobile e-commerce system.

Apart from business models of m-commerce, identical with today’s current business behavior, mobile payment is a core player affecting the whole scenario of m-commerce. There are significant number of mobile users with various incomes and spending power who can affect the adaptability of mobile payments with current business models. In fact, the main desire in mobile payments is providing a convenient way of payment so that a customer can perform payment anytime, anywhere for any available services. Also, there is a variety in kinds of services, value of the transactions, customer habits which are not developed equally in each market. Hence, there was always been a challenge in adopting users’ experience with current and upcoming mobile devices. In other words, mobile devices can offer users many conveniences, such as portability and availability in spite of small screen, lower power supplement, less memory capacity and processing power. On the other hand, there is a lack of a specific mobile payment policy to deal with larger number of customers and low-value transactions. Based on these aspects and limitations, mobile payment methods will be provided with considerable potential for a creative development. Several payment methods are designed or developed in m-commerce area but still new customized payment methods are required to fulfill customers’ needs in special markets.

Naturally, m-commerce is highly regional. Many regions may prefer customized and aboriginal service providers. In other words, a desired m-commerce environment is required to establish strategic relationships with regional financial rules and institutions. Here, payment methods are being introduced in which even the customers with special limitations can operate. The widespread use of mobile devices and consequently electronic commerce assisted us to develop a new integrated system to provide m-commerce services that extends e-business using latest technologies considering special circumstances.

Basically, depending on interaction model, m-commerce applications could be classified into three types: client to server, client to proxy server, and peer-to-peer. Also, m-commerce applications inevitably require essential underlying connectivity features, mobile access adaptation, mobile user profile and mobile security [1]. On the other hand, new technologies usually bring new risk and challenges despite new capabilities and services. In order to design a desired payment method, the inherited risks of new technologies must be overcome in order to leverage their capabilities in handling existing obstacles of payment transactions in
corresponding markets. Mobile payment methods have always been critical, since they are dealing with credits or money. So, providing an adequate security would be mandatory and an inevitable aspect of mobile payments. On the other hand, there has been an issue to preserve a trade-off between usability and security of mobile payments, so that providing maximum security can affect or even violate the usability of mobile payments in practice. The art of work will be combining security and usability to provide a smooth, fast and comprehensive mobile e-payment for end-users. This artifact would be able to evolve into a financial system supporting transaction environments which eliminates or minimizes physical cash handling, as a potential in eliminating criminal activities.

Basically, m-payment process may be implemented in different scenarios, but it includes some fundamental steps: registration, payment submission, authentication and authorization of parties by system service provider, and the final confirmation. In order to provide a secure and comprehensive m-payment, the payment scenario should be designed so that it performs fast and simple for the end-user, but secure and comprehensive for the provider. An efficient payment scenario takes efficient steps in performance. The critical items in each step are the payment messages containing critical information being transferred between participating parties. These messages are objects to which security should be applied. Applying security to messages, to transfer and process payment messages should be done to fulfill a desired fast, secure, integrated and comprehensive transaction. Authentication, secure communication, including confidentiality and integrity of messages, authenticity of sender and recipient, key exchange protocols and non-repudiation should enhance an atomic payment transaction with security.

To establish the desired implementation of convenient m-commerce operations, a need of a mobile application is being felt. With rising new smart phones available in markets, the facilities of smart mobile devices could be exploited to develop an application to perform required m-commerce operations. Current smart phones with different operating systems provide an extensive environment to develop desired applications based on business and users needs.

This project basically followed the potential to overcome specific financial access problems in some developing markets by accommodating unbanked users. Then, design and implementing a mobile application is conducted according to some defined business and professional needs. This report brings a comprehensive overview about its potentials starting form supporting back-end systems, up to design and implementation of account-based payment scenarios used in m-commerce and traditional transactions and evaluating and analyzing of the results against the essential and probable business needs, keeping in mind a perspective for further development and research potentials.

1.1 Research Problems

With development of m-applications, there is an inevitable need of security for all functions of the applications. In this research, the issues regarding payment transactions are considered.
Based on what is designed, implemented and tested, some potential points of development for essential security enhancement could be detected. The most sensitive part of application, the payment function as a mobile payment operation, is the process of exchanging financial values done by two or three parties using mobile devices.

In this project, according to particular business needs and underlying technologies, a specific security policy should be defined to cover privacy and security issues of payment and even functions of mobile application. Apart from security of the complete application in a mobile device, every aspect, from initial point of payment up to final delivery of payment report for involved parties, should be considered.

The following security services are required to establish a secure, comprehensive and smooth payment. There are potentials where the security is an essential property.

- **Authentication** of all parties and objects involved in a transaction.
- **Confidentiality** of messages and information transferred in a transaction.
- **Integrity** of messages and transactions.
- **None-repudiation** of payment parties and objects.

Applying security services and constraints will assist payment functions in performing secure and integrated transactions. The next issue will be design of a system for secure mobile payment transactions. Extensive cryptography, key handling protocols and several packaging and repackaging the payment messages will reduce the usability level of the payment function.

### 1.1.1 Research question

Considering the research problems, there are research questions that need to be answered during the research work.

- How payment’s end parties can make sure about their connection if it is really continued in the secured way behind the TTP proxy?
- How to manage decryption and encryption of the connection content in TTP’s proxy not to make it vulnerable to probable attacks both internally or externally?
- How to provide an end-to-end authentication based on our security architecture design?

### 1.2 Purpose

The ultimate purpose of this thesis is design of a system for secure mobile payment transactions using new methods to provide efficient security for payment messages using existing mobile payment models. In fact, upon considering general characteristics of mobile payment models, a typical m-commerce system involved with a specific payment model is being chosen. Then, based on underlying protocols and methods of the system, a method is proposed to provide security of payment messages during payment procedures. To reach the ultimate purpose of this thesis, there were some other objectives which had to be addressed.
First of all, identifying general and standard characteristics of mobile commerce systems is necessary to understand the applied methods and protocols, so that the payment models could be understood comprehensively in terms of efficiency, usability and security.

Next, whatever the required characteristics of m-commerce systems are, a typical architecture for mobile commerce and financial transactions should be adopted. This architecture should comprise components, protocols and interfaces to provide various services to various mobile applications: registration, security of users at different levels, and protection of its own components. The architecture is required to be modular, integrated, extendible and scalable. This thesis briefly describes design of the architecture and its integration with proposed models and methods besides future research and development plans.

Finally, all objectives above will be reached by proposing methods for secure payments in order to enhance all payment and commerce services with security.

1.3 Delimitations

This thesis deals only with mobile payment security and evaluates only current mobile commerce models and systems.

A general analysis of mobile commerce systems is being mentioned and it is evaluated in terms of security. In fact, having a holistic approach to mobile payment models, a detailed analysis of mobile payment message security has been considered.

There are some general concepts about mobile commerce which are explained briefly and then a short, but efficient and comprehensive description of underlying mobile financial system will be provided.

The underlying mobile financial system is described, so explanation of other components of the system will be discarded and left as an optional source of reference for audiences of this thesis.

The information security concepts are described as they have been applied to the implementation and evaluation of thesis artifacts.

1.4 Methodology

1.4.1 Methodology for Establishing System Requirements

This section describes system requirements being considered in order to evaluate if the purpose of the thesis is successfully met.

Initially, technical, business, and user requirements should be considered for a payment system presenting an interoperable, modular, integrated, extendible and mobile payment architecture that provides the potentials for deploying security extensions.
Secondly, according to the specified system requirements, a financial system appropriate for mobile applications has been chosen. Considering fundamental system requirements of mobile payment systems, the architecture of the system has been evaluated as well as the interactions between internal components and external components of the system.

The third step will be identifying potentials of the payment model of the system for security enhancement along with preserving system behavior including protocols, services, transactions and message structure.

Next, an interface has been designed which is used to interact with the adopted financial system. Development of mobile applications is required to provide required mobile commerce services for corresponding users.

Identifying potentials points of interactions between mobile applications and back-bone system in applying security constrains was the starting point to employ security arrangements.

According to all information related to evaluating system security potentials, security requirement specifications have been determined, so that the system can proceed persistently along with predicted security circumstances.

Based on possible security requirement specifications, some of them should be adopted which are feasible in terms of design, implementation and deployment in an efficient way.

Finally, a methodology for security design and implementation will be planned.

### 1.4.2 Methodology for Design and Implementation

This section describes a methodology for design and implementation of the system described in this thesis report.

A qualitative case study methodology has been conducted in order to provide tools to study existing phenomena within the research context, for revealing and understanding multiple facets of the phenomenon.

According to [2] a case study design is considered in this report, since this study focuses on answering “how all steps of mobile payments should be considered in applying security” and “why it is required to design and employ a security method of payment transactions”; Also the behavior of involved components in the study could not be manipulated. Moreover, it is required to include contextual conditions relevant to the phenomenon under study. Here, the contextual conditions could be financial system infrastructure and mobile interfaces.

Next, in order to determine the case of analysis, it is required to analyze the process of mobile payment in adopted environment. So, there might be a question how the payment process would be carried out between external mobile interfaces and internal components of a financial system. Then, it is essential to evaluate the payment process in terms of security issues. So, it needs to be found out how to design a secure payment system which will
preserve the usual behavior of the backbone financial system. Accordingly, security of payment messages will be considered independent of communication protocol security.

Also, the case study of this thesis could be considered as “Descriptive”, since it describes the real behavior of the mobile payment transactions while applying security constraints.

1.4.3 Methodology for Data Acquisition

This section describes the complimentary methodology for data acquisition in order to make conclusion of the current work.

Using different data sources enhances data credibility for this research project. Potential data sources in this thesis include extensive study and analysis of related works and technology, direct observations, and participant-observation. By data acquisition in integration with qualitative approach, within a case study research, data integration and collection can facilitate reaching a holistic understanding of the phenomenon under study. In this project, multiple related works and experiences of the financial system are then converged in the analysis process. Each work and practice experience contributing to understanding of requirements of the design and implementation method.

1.4.4 Methodology for Evaluating the Results

Having designed and implemented the proposed payment security system, the performance of the system considering the messages security and payment transactions integrity was evaluated in a pattern-matching analysis. In other words, according to the most relevant and successful empirical works, there are certain standard and recognized factors which could be considered as a merit to evaluate this project, so that it could be declared as the most efficient, desirable and promising solution proposed for the common research problem.
2. Background and Basic Concepts

Sufficient background information is given to the reader to understand the context and significance of the problem of the undertaken research.

2.1 Basic Concepts

2.1.1 Mobile Payment

As mobile devices have been transforming into personal trust devices, mobile payment is recognized as interactions between parties in a e-payment system with specific context (e.g. business models, player relationships) and capabilities (mobile device capabilities) so that there is at least one party as a mobile user [3, pp. 30-31] [4]. Basically, the context of m-payments includes any payment in which a mobile device is used in order to “initiate, activate, and confirm” the payment [3, pp. 30-31]. The m-payment services are often carried out through a none-bank party (such as financial and credit institutions) independent of pre-existing bank accounts [5]. Mobile payment systems evolve with new technologies, since they are free of limitations usually applied to bank-anchored services. There are three initiatives that could be considered to best suit mobile payments. First, a mobile device is the most convenient and possible payment technology for mobile context and service purchases. Second, the diminishing use of cash provides the potentials to develop new substitute payment approaches for low value transactions using financial service stations. Third, need of a cost-effective means to charge macro-payments in m-commerce environment [6].

As Figure.1 shows, m-payment system is merely registering and forwarding the authorized and validated payment transactions [7] [8, pp. 200-212]. Payment system life-cycle includes payment request creation, payment request authorization, and payment request committal [6].

![Figure 1: Conceptual Schema of the Mobile Payment System](73)
Principally, m-payments occur between four stakeholders: mobile consumers subscribe to a service, merchants, who provide product or service to consumers, payment service provider, which controls the payment process and the trusted third party that administers the authentication of other players and the authorization of payment settlement. Note that different roles can be merged into one party and act as one player. For example, payment service provider, which controls payment process and trusted third party, can act as the same stakeholder [9, p. 12].

**Bluetooh:**

Bluetooth, as a popular short-range communication technology, enables mobile devices to communicate with each other using 2.45 GHz frequency at the distance of up to 80 meters, although distances of up to 10 meters are more common [10] [11, p. 26]. BT traditionally supports data transfer rates up to 3 Mbps, though BT 3.0, which incorporates 802.11 standards, can support transfers even up to 24 Mbps [12, p. 285]. Also, Bluetooth technology provides a better connection, since data transfer dispatches signals in all directions [11].

The impressing aspect of Bluetooth is its impact on increasing development of peer-to-peer use for Bluetooth devices. In addition to file-sharing tasks, a more common task is using Bluetooth-enabled devices to interact with other BT-enabled devices in the intermediate proximity. Other benefits of BT communication protocol is that the messages can be sent to other phones without knowing phone number. These messages could later be used or shared with other users or exchanged for commercial purposed, like exchanging digital goods, tickets or coupons [13, pp. 126-128].

### 2.1.2 Mobile Payment Models

As already explained, transaction of digital value basically includes three phases: payment initiation, payment authorization, and payment settlement. Mobile payment models can be characterized based on some important features, such as: payment amount, payment settlement mechanism, and the technologies which support the complete m-payment system [14, p. 84].

With respect to monetary value of a payment, the substance of value will be digital or paper cash. Digital cash can be used as equivalent to the paper cash. Basically it preserves user’s anonymity and mainly enables off-line transactions. While in other models payment value should be verified by third-party operators [15, p. 2].

Considering the payment amount, the payment will be either macro or micro payment. A macro-payment usually involves amounts more than $10 especially for credit card payments. A typical micro payment scenario basically will be settled by third-parties for authorization and verification request by card issuing bank or financial institutions and in other words, banks pay mobile merchant for the user [16, p. 414]. While micro-payments normally deal with less than $10 amounts and usually are charging users facilitated by mobile network operator through the billing system [17].

Mobile payments can be classified into three major types in terms of payment settlement mechanisms:

- **Account-based payment systems**, which are based on mobile phone numbers, smart card or credit cards. In account-based systems, the transaction amount is charged by the mobile subscriber’s account or credit/debit card [17] [18, pp. 343-345].

Generally, account-based payment model includes four parties: customer, merchant, issuer or customer’s financial service provider, and acquirer or merchant financial institution. In some occasions, there might be another party called payment gateway or proxy acting as an interface between issuers and acquirers in the network of banking side and customer and merchant at the Internet side [19]. In this payment system, each user owns a specific account.

With respect to the type of the supporting technology, m-payments are classified into contactless and remote. ‘Contactless’ or ‘proximity’ payments are performed with the physical presence of a customer at the point of sale and actually ‘face-to-face’ or ‘machine-to-machine’. For example: buying a product from a vending machine. Contactless payments use a radio-frequency interface between the mobile device and the beneficiary’s payment device [18, p. 345]. In fact, remote mode transactions are performed over the network or ‘over-the-air’ (OTA). OTA services utilize mobile device facilities that transmit data via GPRS, 3G or WiFi. Using OTA m-payment, consumers can initiate transactions directly from their devices to the payment service provider. In the OTA mode, the confidentiality and integrity of data will be satisfied due to strong encryption and integrity circumstances [20, p. 376]. Contactless or proximity payments can be fully or partially initiated or settled ‘over-the-counter’ (OTC) according to available proximity communication facilities in mobile device.

### 2.1.3 Mobile Payment Transactions

As already explained, there are two main classes of mobile payment systems with different natures of mobile payment transactions.

In OTC payments, the customer is physically present at the point-of-sale, and mainly the transaction is conducted using a wireless device using proximity communication protocols. While, in OTA payments, payment transactions are performed where the consumer is physically remote from the point-of-sale and closer to Internet payment gateways. OTA payments usually require a more sophisticated infrastructure for wide acceptance of payment requests [21, p. 154].

To complete a mobile payment transaction, three steps must be successfully performed in sequence: payment request creation, payment request authorization and, payment request settlement [22].

### 2.1.4 Mobile Payment Applications

Mobile payment applications can basically an interface handling financial functions which allows a user to perform various payments between the customer account, merchants and bank and also to purchase goods, coupon and offers by different merchants using Smart-
phones. Hence, mobile payment applications are mainly required to provide a secure route for payment request from customers to merchants and vice versa when being used as POS on a merchant side.

2.2 Related Work

Various security services must be provided for mobile payment services. Also, there are specific threats for mobile payment services. There are previous research results about providing essential security services and probable threads corresponding to the payment services occasions. On the other hand, based on different deployment methods of financial mobile applications, there are different architectures of payment methods. There are some common issues in mobile payment standards [9].

- Security: Considers the usage and trust of customers and merchants in the integrity of the payment network, it is vital to increase security level by applying security services: confidentiality, authentication, integrity, authorization, availability and none-repudiation.

- Interoperability: It is preferred that any typical payment method can be used at any participating mobile commerce system.

- Usability: It is required to consider users’ consumption behavior and habits to make the payment system more user-friendly.

- Privacy: It is required to protect collected information of the participants of transactions took place over the Internet or stored locally on client side. This information can be useful outside of the transactions, so that any of that information may be linked in a way the participants without their knowledge or consent. Privacy requirement prevents use or disclosure of any personal information and keeps them secure according to defined privacy policy and obligations.

In order to design desired payment options, some issues such as regional support, consumer preferences and customer base should be considered [22]. So, the desired goal of a typical m-commerce system is to establish a balanced trade-off between main issues of mobile payment standards for mobile users to perform convenient e-commerce transactions in a simple and secure way [23].

For developing markets there are many security concerns for m-payment systems. For instance, there are mobile applications environments that include kind of security model, where mobile payment parties connect to each other in two independent secure approaches. Actually, depending on m-payment architecture, there might be one secure connection either between service provider and mobile operator or between mobile operator and user’s mobile terminal or even a secure connection between user’s mobile terminals. Hence, to assure a secure transaction, there is an implication that every secure transaction involving its entities (service providers, mobile peers and mobile operator) requires a trust level between all entities. All of transaction players should be sure that its connection is extended in the secure
way to other transaction participants. Also, if the content of connection is being encrypted and decrypted by transaction peers, that may make it vulnerable to potential threats for each transaction entity. So, end-to-end authentication is a desired feature. Moreover, in developing markets, m-payment service providers rely on agents for customer acquisition and payment verification. They use customer’s sensitive information and credentials for identification and authentication purposes. These agents are vulnerable to be compromised to customers’ information leakage [24]. Furthermore, mobile devices are potentially vulnerable with malwares performing unauthorized operations, such as sending sensitive user information using available connectivity features.

In the following, the most relevant research results are described in order to emphasize different aspects of payment scenarios and security arrangements corresponding to security threats and vulnerabilities, as well as interoperability and usability. In general, these payment protocols comprise transaction parities including customer, merchant, agents, service provider, and mobile operator network. Then, for all the discussed payment methods, the security issues of payment messages will be reviewed and a candidate payment protocol with transaction messages security will be proposed as this thesis purposes and potential security threats will be discussed as well as a supporting potential future work.

### 2.2.1 Architecture for Secure Two-Party Mobile Payment

The research performed by J.E.Rice and Y.Zhu proposes architecture for secure two-party payment model for mobile payments. The involving parties are a customer and a payment service provider. This architecture focuses on applying security at the application layer. They considered the application layer’s security isolated from security protocols in the lower layers. The architecture is designed in such a way that it handles all the security-related functions free of any modifications to the existing communication infrastructure and protocols.

![Architecture for Secure Two-Party Mobile Payment](image)

*Figure 2: Architecture for Secure Two-Party Mobile Payment [26]*
They proposed a secure architecture for two-party mobile payment based on application-layer security architecture to provide end-to-end security, implementing a digital signature module [26].

**Security Mechanism**

As Figure 4 illustrates, during a payment transaction, the system transfers the transaction message attached with the digital signature’s public key over an unsecured network link. In order to protect transaction messages from third party eavesdropping, both signature and encryption layers are used to process messages. In this architecture, digital signature layer ensures that the message is sent from the right client to the right server. Hence, they combined the SIM (Subscriber Identifier Module Number), PHID (mobile phone serial number) and ACCID (user’s bank account number) as the Client ID, then signed all that and appended to the message. Client → Bank: Encrypt {Sign {message, Client ID}}

![Figure 3: Security Mechanism for Secure Two-Party Mobile Payment](image)

Due to the J2ME limitations, ECDSA has been adopted because of its low computational cost, higher performance, a fast signature generation, and short key size. [26] Elliptic Curve Digital Signature Algorithm (ECDSA) is adopted to implement Digital Signature Algorithm (DSA).

In this architecture, both of two key pairs are used for encryption and digital signature generation. As Figure 5 illustrates, private key generated on a mobile device is used for generating digital signature while other party uses the corresponding public key to verify the digital signature. To keep the key pair (including private key) secret, they are generated and stored in the mobile device.
Basically, when Java applications are being compiled, class files are generated in machine language so; this process makes it difficult to understand details of the private key. The RMS (Record Management System) APIs provides the ability to manipulate records between different applications and shares records within an application, so that access to these records is strictly prohibited. The key pair will eventually expire, and the banking server detects if any renewal of the key-pair in needed then, initiates the renewal of a key pair by notifying mobile device to generate a new one.

### 2.2.2 A Lightweight and Secure Protocol for Mobile Payments via Wireless Internet in m-commerce

A.ATabandehjooy and N. Nazhand mentioned some major problems with the previous method, as follows [27]:

- Participation of the merchant and acquirer in wallet payment protocol may raise processing rates and make the payment process to take longer.

- Cryptography algorithms, hash and digital signatures have been highly used.

- Limiting users to use particular devices supporting special software.

In addition to solving these problems, they proposed a new method making the security mechanism simpler.
Security Mechanism

- Initially, the customer initiates the payment using her mobile device by sending a request to her issuer. This is basically withdrawal of money from his/her account and transfer of it to merchant’s account. This request contains the following information: account number or credit card number from which customer needs to withdraw, a complex key (being generated from users’ secrets) and, the destination account or credit card number along with the amount of money.

- Then, the customer can commit the transaction through a payment gateway provided by the issuer. All the information being transferred via gateway between the issuer and customer, is encrypted/decrypted with WTLS protocol.

- Issuer performs all transactions between himself and the acquirer in a secured tunnel.

- Depending on some established agreements, acquirer sends a text message or an email to merchant after transferring money.

Issuer rolls back all these transactions in case of any problem during any phase of transactions.

![Figure 5: First Method which Customers use for Payment](27)

During payment process, the customer generates a complex key using a private key given by the issuer and a number associated with him. This complex key in spite of its simplicity is different for each payment transaction. So, only the issuer and the customer know about its details and how it is been generated, so that it guarantees authentication and non-repudiation.

Also, during payment process acquirer needs to create payment authorization response. A complex key will be generated. When this information is transferred to the issuer having it checked for their validity, commit or roll back the transaction. Actually, there is an agreement
established between the issuer and acquirer about the algorithms. This agreement can be used for generating the complex using agreed algorithms. In all steps of exchanging data, the complex key is being used for cryptography to provide confidentiality and integrity.

2.2.3 Secure Mechanism based on Concurrent Signature for Mobile Payment Services

This research has been performed a review over a mobile payment system and its security issues and proposes a payment model and protocol based on concurrent signature scheme. The research earlier evaluates a payment model then, proposes a payment protocol to resolve probable weakness points. The payment model is the following [28]:

Initially, client starts the payment from the wallet application to the payment gateway (PG); next, PG exchanges messages with associated banks and merchants and sends the result back to clients. Actually, Wallet using the security module embedded in the application software performs the encryption of all exchanged messages. Also, all users having mobile access interfaces can connect to a CA for the assignment of an authentication key. In the above model, the security of the model is based on PKI. All entities of the payment service have their own digital certificates.

In the above model, the merchant and the customer need to trust each other. So, an exchange protocol is required to ensure that no one can take an advantage over the other party by misbehaving the protocol [14]. PKI could be an alternative scheme for establishing fair information exchange. But, the relatively high cost of PKI for mobile devices with low computing power caused to compensate this. So, an algorithm named “Concurrent Signature” excluding any third party has been presented to decrease the complexity especially for mobile payments’ computations [29] [30]. To resolve problems mentioned above, a mobile payment protocol based on Concurrent Signature has been proposed, involving only clients and merchants, which guarantees the confidentiality, authenticity, non-repudiation.

Security Mechanism

Figure 6 features the details of the exchange protocol for mobile services using “Concurrent Signature”. The details of the message content, hash and encryption functions can be found in [28].
As the client/Alice chooses a digital product by her mobile devices, she receives the shopping order. Then, the order will be encrypted by the Alice’s signature, attached to it is an unsigned check, and will be sent to merchant/Bob.

Merchant receives the Client’s message and verifies of all received messages in terms of validity. Based on the verification result, Bob sends the hint messages back to Client. Next, Merchant chooses a secret key \( k \) to encrypted digital services and then computes hash of \( k \) (\( f=\text{Hash}(k) \)) and encrypts the message, sales service commitment and \( f \) by \( k \). Merchant encrypts \( k \) by public key of Client and sends back all these messages to Client.

Client decrypts messages received from Merchant by its private key and retrieves the \( k \), then decrypts service messages to retrieve message content, sales service commitment and \( f \). Client checks the sales service commitment and its signatures check by \( f \) and if it was satisfied with the result, it sends \( S \) back to Merchant otherwise; it rolls back the whole transaction:

\[
S= (w, h, f, \text{Client’s public key, Merchant’s public key, C})\]

in which \( f \) stands for \( \text{Hash}(k) \) and \( h \) for a cryptographic hash function and \( w \) stands for a hash function over Client’s private key.

When Merchant receives message \( S \), it performs signature check and further validation. If validation process was successfully validated confirmed, the check is confirmed. Then, merchant sends shared secret \( k \) to Client, and sends the signature check attached with \( k \) to the corresponding bank servers for value transfer.

Bank, after successful validation of the legitimacy of Merchant as the receiver of check by \( k \), performs the transfer. Otherwise, it rejects the transfer request.

Finally, Client decrypts message using key \( k \) which has been received from the Merchant and gets the digital goods.
3. System Design and Architecture

In this thesis, different mobile payment systems have been considered and evaluated relevant to our proposed system design. There are two groups of criteria that ought to be considered relevant: functional and architectural. The functional criteria basically should enforce the system policy and what the system should be able to do to satisfy the system requirements listed in section 1.2, and the architectural criteria, i.e. Interoperability, Usability, Simplicity, Security, Privacy, Trust, Cost and, Availability define how the system should be constructed.

3.1 Functionality

The purpose of this thesis is to construct a system that enables payment using available Wi-Fi, GPRS, 3G, and Bluetooth compliant mobile terminals which are equipped with mobile devices. The following criteria are being fulfilled by the system in terms of its functionality:

1. Implementing a secure means of initiation, authorization and, settlement of payments, using either credit or bank accounts.

2. Implementing an integrated and secure registration of consumers.

3. There are three actors that interact with the payment system: the customer, the merchant, and the agent.

4. Providing an authentication service to authenticate the end-users in efficient ways in a flexible manner by any available and required means and also, to provide protection for data exchange and authorization.

5. Enabling customers to perform payment transactions directly with the payment system and also, should enable merchants to register payment transaction requests as well as agents to confirm mobile payment transactions, with details about the payment transaction into the payment system.

6. Keeping the record of payment information including registration status of users (customers, merchants and agents) and the status of payment transaction requests and payment messages received by the system.

7. Providing the consistency and integrity of payment transactions initiated by consumers and committed into the payment system.
8. Committing the registered payment transactions, only when the payment scenario has been successfully performed and all engaged consumers have authorized the transaction and the authorization is verified by the system.

### 3.2 Security

The system’s design in terms of security should be in line and compatible with other involved components of the system. Hence, the system design and implementation is dependant to security modules and issues related to key lengths, key generation, certificate issuance, distribution and, revocation and, security module implementations. All these issues are considered in order to fulfill the following required security requirements and accomplish the desired performance of the payment system:

1. Provisioning mobile payment applications including preparing and loading mobile applications into user’s mobile device with personalized keys and also deployment of unique personalized keys to protect information store and retrieval and the transactions made by the m-applications, so that, user’s profile and data exchange must be secured to ensure that no data is compromised. This functional security requirement can provide the security service of “non-repudiation”.

2. The keys should be generated compatible with key generation formats and standards of other modules and components of the system. Also, key generation functions should store and retrieve generated keys from hardware security modules installed in mobile devices, that they cannot be recovered by any means.

3. All mobile applications should follow PKI standards defined by the CA of public key certificates.

4. All configuration files and data should be encrypted in secure elements of mobile devices using compatible and standard encryption modules of the system.

5. Access to the system and the system database has to be provided only through the specific interfaces provided by the system.

6. Access to the files and the mobile application database has to be provided only through the specific interfaces provided by the application.

7. All communications between mobile applications and the back-end gateways and servers must be encrypted.

8. OTC transactions should be performed only by authorized users and mobile applications.
3.3 System Architecture

The preliminary research effort led me to propose an architectural design for Secure Mobile Payment System with certain design considerations in order to provide and satisfy recently introduced requirements for mobile commerce and financial transactions. This architecture will solve research problems and fulfill requirements listed in section 1.1 to propose a method of secure payment to enhance all payment and commerce services in terms of security. This architecture should comprise components, protocols and interfaces to provide various services to various mobile applications: registration, security of users at different levels, and protection of its own components. The architecture is required to overcome the following challenges: Interoperability, Usability, Simplicity, Security, Privacy, Trust, Cost, Availability, and Cross-border payments in order to operate as widely adopted mobile payment architecture [31, pp. 44-66].

The system architecture is designed in a modular way so that, it is possible to plug new services and components compatible with the system standards into the system without interfering with other modules, services and components.

Basically, the proposed architecture design is based on a regulatory environment where every mobile payment transaction goes through the bank accounts of payment involved parties in flow form of consumer-bank-consumer. The mobile devices mainly act as an interface to access the bank through back-end payment gateways systems.

For the design of a system for secure mobile payment, there are some important security aspects in mobile environments which should be considered. In order to provide efficient security, it is required to secure all participating components of the architecture, as well as communication between these components. One of the main components of the system architecture is an infrastructure for support and settlement of financial transactions. A service-oriented security infrastructure is adopted suitable for mobile financial transaction. In this infrastructure, a system exists which is called SAFE (Secure Applications for Financial Environment) which is designed and implemented to provide a secure, convenient and reliable large-scale infrastructure for mobile financial transactions.

The components of this system architecture are secure mobile applications for different kind of consumers (Agent, Merchant and, Customer), Location-based authentication service and, three SAFE servers: Communications (Gateway) Server, IDMS (Identity Management System) Server, and Payment Server. All these components are integrated through a secure messaging system and can provide a number of secure financial services and for this thesis specifically payment services.

This chapter mainly presents necessary design and guidelines for implementation of the system. According to requirements and criteria specified in previous chapters, a high level modular model for the system will be specified to describe the functionality and the processes that the system impalements.
In below, the proposed architectural design for secure mobile payment system will be illustrated and every component of this architecture will be described, as much as it is required for this research.

3.4 The Purpose and related Work

As already introduced, the ultimate purpose of this thesis is proposing a system for secure mobile payment transaction using a new method to provide efficient security for payment information messages using existing mobile payment models. In order to design desired payment options, some issues, such as regional support, consumer preferences and customer base, should be considered [22]. So, one of the desired goals of a typical m-commerce system is to provide a balanced trade-off between main issues of mobile payment standards for mobile users to perform convenient e-commerce transactions in a simple, and secure way anytime and anywhere [23]. Figure.9 features an abstraction of all these in a functional model of a mobile terminal designed for m-commerce applications.

![Functional Model of a Mobile Terminal designed for m-commerce Applications](image)

As explained in section 2.2, J.E.Rice and Y.Zhu proposed architecture for secure two-party payment model for mobile payments where the involving parties are assumed to be a customer and a payment service provider. Some current security solutions at the communication layer in mobile environments are not adequate. Therefore, it is concluded that security at the application layer must be added in order to achieve end-to-end protection for mobile financial environments. The architecture design in this report, similar to their architecture design, focuses on applying security on the application layer. In fact, the application layer’s security is considered independent of the lower layers’ security protocols, so that the architecture is designed such that the application handles all the security-related functions free of any modification to the existing communication infrastructure and protocols.
3.5 System Components

The components of this system architecture are secure mobile applications for different kinds of consumers (Agent, Merchant and, Customer), Location-based authentication service and, three SAFE servers: Communications (Gateway) Server, IDMS (Identity Management System) Server, and Payment Server comprising the SAFE system. All these components are integrated with secure messaging system and can provide a number of secure financial services and for this thesis specifically payment services.

3.5.1 SAFE System

SAFE (Secure Applications for Financial Environment) is a system capable of performing various financial transactions with mobile clients or any kinds of mobile devices. SAFE system supports transactions with multiple banks, direct client-to-merchant payments, peer-to-peer transactions, and other non-banking mobile applications [59, pp.21].

To follow specifications of the system design, account-based payment model is adopted as one of the payment models supported by the SAFE system. As already explained, this model includes four parties: customer, merchant, issuer or customer’s financial service provider, and acquirer or merchant financial institution. In the SAFE model, as the main features of the system, there are mobile pre-paid accounts (PPAs) used to deposit and withdraw cash and also for various mobile payments, so that a consumer can pay with an account associated with its mobile phone number through communication networks. When the customer intends to make an m-payment transaction, he/she can access this wallet, select from which account they want to pay and the beneficiary account number. Then the value is debited from the account of the customer and is transferred to the merchant account [32].

SAFE system can provide the following services based on the system design [32]:

- Management and registration of pre-paid accounts (PPAs) for system actors: Agents, Customers and business entities (content providers and merchants);
- Use of those PPAs for financial transactions: over-the-counter (OTC) and over-the-air (OTA) payments, cash deposits/withdrawals, and account transfers.
- Issuance and management of biometrics smart cards for system administrators, SAFE agents, customers, and merchants for authentication, authorization and payment against PPAs.

SAFE system supports various types of transactions. In this system architecture design, mobile banking and mobile commerce are the main financial applications for payment transactions.
3.5.1.1 Mobile Banking

Mobile banking, at an advent of mobile payment technologies, refers to provision and settlement of financial services using mobile devices. SAFE system, offers mobile banking services with its business model, actors and use cases.

SAFE mobile banking service can be said to comprise three concepts:

- Mobile accounting
- Mobile brokerage
- Mobile financial information services

Mobile accounting and mobile brokerage can provide transaction-based use cases while non-transaction-based use case can be considered as mobile financial information services vital for conducting transactions such as, balance inquiries which might be required prior to credit transfer [33]. All these services are provided in combination with information services which in contrast, maybe provided as an autonomous unit.

The participants or actors in m-Banking transactions are the following:

**Banks**: perform registration and certification of individuals and provision of financial services,

**Consumers**: individuals initiating or receiving transfers as the result of financial transactions,

To accomplish the described SAFE mobile banking model, large-scale, federated security architecture is designed which comprises two general types of servers in a mobile banking system:

**Gateway Servers**: specialized servers that support various secure communication functions, used as the front-end proxies to bank servers.

**Bank Servers**: internal servers in banks, performing standard banking applications and transactions.

Meanwhile, client mobile stations including mobile devices enhanced with secure applications and used as an interface interacting with the system to perform financial transactions from mobile locations.

The functions of these components and transactions between them for individual mobile applications are illustrated in Figure.10.
In the following, money transfer transaction is illustrated:

“Money Transfer” transaction may be performed between two personal accounts or between a personal and a corporate account. In any case, one customer is the sender or initiator of the transaction and the other customer is the recipient. This transaction as a primary and basic transfer function may be used for all kinds of payments and money transfer affairs.

If the two parties of money transfer have accounts in the same bank, then the sender initiates the transfer of certain amount of money indicating his/her account and the account of the recipient.

Transfer_Request message, including all required parameters, is sent from the sender’s mobile application to the SAFE servers, which upon successful verification and commission of transfer by the bank, informs the recipient about the transaction result status.

Using the SAFE system, users can perform m-commerce transactions either using Over-the-Counter (OTC) or Over-the-Air (OTA) protocols. For OTC transactions, users basically use proximity protocols supported by available their mobile devices (Bluetooth or NFC), while merchants use specialized PoS terminals or mobile devices capable of performing proximity protocols. For OTA transactions, users can use their mobile devices containing secure wallet applications in order to perform full transfer cycle through wireless protocols.

Moreover, secure wallet application can also perform standard debit/credit card payments. All the information of the credit cards can be entered into customer’s mobile device either during registration or during the process equivalent to credit cards issuance and then stored in a secure element of a mobile device through the provided interfaces of the wallet application. Merchant PoS terminals and mobile devices containing wallet application will be capable to accept such data through proximity protocol. Wallet application user interface is efficiently designed to follow the same steps in today’s debit/credit cards transactions.
Payment using smart card in OTC mode includes using mobile device to reach merchant PoS terminal or mobile device to provide cards number and other data for merchant through available proximity protocols. While in OTA mode, initiating payment by selecting an already registered card will be possible through wireless protocols. While merchant needs to connect to SAFE server to verify the authorization of transaction and upon receiving authorization result and termination of transaction, merchant sends back the transaction result.

Also, “Digital Cash Dispensing and Micropayments” are supported by SAFE system. In fact, it is possible to load cash from recently registered bank accounts or credit cards to your mobile device and digitally store it. So, instead of cash you can use stored “digital cash” for later micro-payment transactions. Hence, you can use the type of payment using corresponding mobile devices equipped with hardware and software supporting appropriate proximity protocol and required mobile application.

In order to load digital cash into customer’s mobile device, customer needs to send “Cash_Request” message via available interfaces through mobile application to the SAFE server. Then, after successful validation, “digital cash” is debited from customer’s account, loaded and stored in his/her mobile wallet. When the customer needs to initiate a micro-payment using OTC mode, he/she initiates the payment using proximity tools to PoS side or merchant mobile device then the payment amount is reduced form customer’s “digital wallet” and transferred to merchant PoS terminal or mobile device. Finally merchant sends “Cash_Reclaim” message to the SAFE server which contains merchant’s bank account number to make deposit into it. For OTA mode, customer simply can choose its pre-loaded digital wallet as the source account for payment and the rest of transaction goes on the same as with OTC mode through wireless protocols. Moreover, it is possible to unload digital cash from digital wallet back to any registered bank account or credit card.

### 3.5.1.2 Mobile Shopping

Mobile shopping feature of SAFE system encapsulates the following functionalities to cover new e-commerce trends.

**M-Promotions** is a functionality that has been considered in our system design to make a convenient way for merchants to present information to customers as well as others and increase demands of their products and services and also for customers to receive the latest popular special offers on their mobile device. Basically, promotions can be uploaded by merchant through available interface on their corresponding mobile application available on the market visible for customers through their wallet applications. Customers can select and download the available promotions filtered by user’s location into their wallet application, so that they can view stored promotions all in wallet offline mode.

**M-Coupons** is a functionality that has been considered to provide digital coupons containing financial discounts or rebate issued by manufacturers of consumer package goods and services, to be used in retail stores as part of promotions. M-coupons functionality not only can enable prices conscious customers to use this coupons and a form of price discrimination,
but also, can enable merchants to offer coupon with lower price offers, targeted selectively to regional markets specifically where regional markets with great price competitions. SAFE system has enabled merchants to upload their coupons into m-marketing server using provided function of the merchant smart device application. Also, customers can view available coupons in the market filtered by customer’s location and select those that have been uploaded by merchants close to the customer’s location and store them in wallet application.

**M-Tickets** function enables customers to inquire, pay for, obtain and validate tickets conveniently using mobile devices. This functionality can reduce the production and distribution costs of traditional paper-based tickets by providing simple ways to purchase tickets conveniently. Moreover, this functionality provides merchant to upload any kind of ticket to the SAFE system and customers, on the other side, can view available tickets in the m-market server of SAFE system via designed corresponding interfaces of mobile application. These tickets can be purchased and then downloaded into wallet application. Tickets on the m–Marketing Server can be filtered by customer’s location. Tickets may be paid using OTC/OTA payment method – SAFE accounts, bank accounts, bankcards, or digital cash and upon successful payment be downloaded and stored in wallet application.

**M-Parking** provides four functionalities:

- **Search Parking**: users may search available parking place at the local garage / parking location by giving its registration number
- **Pay Parking**: Users may pay for parking by specifying parking place, parking time and selecting method of payment.
- **Extend Parking**: The system will warn users about expiration of their parking time. In that case, users may extend parking by specifying parking place number and additional parking time.
- **Pay Ticket**: users may use SAFE system to pay tickets for parking violations. The system will notify users if the ticket has been issued for parking violation. In that case, users may pay the ticket using SAFE system, by specifying ticket number, parking place number, amount to pay and by selecting the method for payment.

All mobile parking functions require SAFE system to be integrated with the parking system of some Parking Authority.

**M-Gift Cards** functionality enables merchants to upload their gift-cards as a restricted monetary equivalent issued by retailers or banks as an alternative to non-monetary gifts to SAFE m-Marketing server via mobile application interfaces. Customers can list gift cards loaded into the m–Marketing Server. These gift cards can be purchased using OTC and OTA payment methods and then downloaded and stored in the wallet application. Gift card can be transferred to another SAFE user and may be used as one of the payment options with designed payment interfaces in wallet mobile application.
3.5.2 Mobile Applications

SAFE Wallet application supports three groups of mobile functions:

- Various types of mobile payments using mobile pre-paid accounts, standard bank accounts, bankcards and digital cash payments in OTA and OTC mode. Wallet application provides an interface for customers to perform various types of payment using different types of accounts that SAFE system supports. Using Wallet application, customers can use a specific type of payment based on the situation and available capability of host mobile device.

- Wallet application enables its user to register and create associated pre-paid SAFE accounts, as well as credit/debit card registration in order to provide most of the usual payment methods.

- Users can perform deposit, withdraw and transfer transactions using their pre-paid SAFE mobile accounts between all SAFE account owners and then list all launched transactions filtered by defined constraints.

SAFE Wallet will be described and demonstrated in detail in Chapter 4.

SAFE Merchant application supports three basic mobile functions:

- Mobile payments with subscribers using over-the-air (OTA) SMS and wireless protocols and over-the-counter (OTC) Bluetooth protocols. Using merchant application, OTC payment is performed using Bluetooth and NFC proximity protocols and OTA using wireless protocol, either provided by the Internet providers or telecom carriers.

- Various types of mobile marketing functions: creating and uploading promotions, mobile coupons, mobile gift-cards, and mobile tickets into the SAFE system marketing server.

- Various types of mobile business functions: accepting discounts based on promotions, mobile coupons, mobile gift-cards, and verification of mobile tickets. Also, providing mobile transactions at locations using PoS systems, then mobile services for the owners/drivers of vehicles, and various mobile commerce transactions and accepting and clearing mobile vouchers from customers, uploaded by merchants using m-Marketing services.

- Mobile security functions: registration of a merchant and administration of and access to SAFE mobile pre-paid accounts, configuration of the SAFE system, detection of locations, management of Merchant’s local and SAFE system PINs, selection of cryptographic options, and managing X.509 digital certificates.

SAFE Merchant will be described and demonstrated in detail in Chapter 4.
SAFE Agent application is being used by agents, merchants or any other member of the SAFE system, authorized by the system administrator to perform the functions of an agent.

- Self-registration and registration of subscribers and merchants, including registration of locations for merchants
- Cash-in and cash-out financial transactions with subscribers and merchants.
- Mobile security functions: registration of an agent, management of agent’s local and SAFE system PINs, selection of cryptographic options, and managing X.509 digital certificates.

SAFE Agent will be described and demonstrated in detail in Chapter 4.

### 3.5.3 Security Components and Architecture

As mentioned earlier in section 3.3, the goal of this research is designing a secure system for mobile transactions. Also, the system design and architecture support various financial mobile applications and transactions. Since all the functions and transactions are basically financial operations, the main concern must be their security. Therefore, one of the most distinguished features of the whole system architecture is its comprehensive security. In fact, system architecture is designed in such a way that existing components can be enhanced with security countermeasures, so that the integrity and availability of the whole system would be preserved. Here, the security infrastructure of the system architecture will be described.

Fundamentally, according to the inherent characteristics of financial transactions, the item of trust should be established between participating parties. There are security requirements that must be supplied in the security architecture design to provide the sufficient trust:

**Authentication**: User identification verification and approval is essentially required in system design and must be efficiently provided. Each participant in the system, including all functions and transactions, needs to make sure that counterparty is the one he/she is interested to communicate with. There are some factors as the “basic instruments available to a human user to authenticate her in order to convince a computing system of her true “identity”, as is known or registered in the system” [34]. These factors are called authentication factors and are generally classified into three categories:

- **What an entity knows**: It is something secret that ideally only the valid subject should know, for example, password, PIN, answers to security questions.

- **What an entity has**: Identity cards and licenses or any physical tokens which imply identities may make entities recognizable. [35]. These factors are usually something physical that the user owns it and only the user who owns the correct token can be successfully authenticated.
• What an entity is: basically based on physical characteristics of the user which are uniquely associated with him/her, such as fingerprint, the pattern of user’s voice or face. [35]

**Integrity**: in this context specifically means that transaction contents must be created or modified only by authorized parties or only in authorized ways to assure all participants that the received messages have not been altered in any way from the original message. Generally a message digest of the original message is attached with the message for the recipient to verify the integrity. The cryptography prevents changing the data block (the plaintext) and also changing the checksum value (the cipher-text) to match [35].

**Confidentiality**: ensures that the transaction contents are accessed only by authorized parties. Basically access can be reading, viewing, printing or knowing that a particular asset exists. In this context, encryption and decryption are the methods to achieve confidentiality. Both types of crypto systems are used to provide confidentiality: symmetric and asymmetric cryptography [35].

**Non-repudiation**: as a security requirement it provides and maintains evidence, so that the participants of transactions or interactions cannot deny their participation in that transaction. There are some factors required to provide non-repudiation qualified enough as one of the system security requirements [36, pp. 390-400]:

• Capture information about the actions that a participant did in transactions or system events. Here, the information is required which are explicit enough to help assign accountability [36, pp. 390-400].

• Preserve and protect all information required to achieve non-repudiation associated with an event. It is quite important to keep all non-repudiation data uncorrupted in order to enforce accountability. Increasing non-repudiation service quality provides a degree of confidence about integrity and reliability of information.

• Preserve availability of system services in addition to non-repudiation service. Non-repudiation service should be accomplished so that system participants get discouraged due to transaction complexity and duration [36, pp. 390-400].

**Privacy**: function of protecting the collected information of the participants of transactions that were performed over the Internet. This information can be useful outside of the transactions, so that any of that information may be linked in a way the participants without their knowledge or consent. Privacy requirement prevents use or disclosure of any personal information and keeps them secure according to defined privacy policy and obligations.

**Availability**: as an integral requirement each security infrastructure needs the complete security design in order to provide expected services available enough for its intended users.

Security design architecture provides necessary protection for all data both on a client side (mobile device) and at a server side to achieve an end-to-end security. The following components comprise design of the proposed security architecture:
Local security module: providing local (mobile device) security using encryption procedures and secure elements in order to protect sensitive local data stored in a mobile device.

Identity Provider (Registration) Servers: providing registration services to create, maintain and manage of identities and their associated information.

Certification Servers: issuance, management and distribution of X.509 digital certificates for system participants based on PKI.

Authorization Servers: providing authentication and authorization services for system authorities based on secure web services and, for consumers, based on location-based authentication services.

Based on the architecture design illustrated in Figure 11, users establish transaction flows both with other users and with various service providers via SAFE Gateway Server. The SAFE Gateway Server, as the core component of the SAFE system, provides communication with users at the front-end through any available kind of wireless communication protocols and connects with security providers and service providers at the back-end through stable TCP/IP connections. It receives various requests from clients in SAFE messages format and upon interpretation dispatches these request messages to different requested service providers [32].

Security providers basically provide security services synchronized with service providers’ needs and client-side standards, so that they can accomplish the most efficient access control and transaction security countermeasures. They are playing key role in providing security services needed to accomplish security requirements. Communication protocol between security providers, service providers, and front-end components are all standardized and designed to achieve efficient system functionality.
3.5.3.1 Client Security Module

Front-end side clients communicate with the system using smart mobile devices. In fact, users may communicate with back-end servers through various mobile applications. Mobile client structure is illustrated in Figure 12.
Mobile device client internal structure comprises four independent modules. In other words, mobile device includes mobile application and removable secure element. SAFE Mobile application logically comprises:

- Communication module: for establishing communications with other nodes using available and supported communication protocols and protecting communication using application layer security circumstances.

- Business Logic module: containing business logic of the application function and transaction logics and procedures.

- Security module: providing required security capabilities required for application functions and transaction messages security as well as communication module.

Secure element being used with the described mobile applications, are removable SD cards. SE basically provides a secure environment for storing sensitive information and cryptographic keys.

### 3.5.3.2 IDMS server

IDMS server provides services for registration of SAFE system participants and their authentication, authorization, roles and privileges within the system boundaries in order to increase security and productivity of the entire security infrastructure. Registration of users may be performed either via registration interfaces in mobile applications or by SAFE Agents under supervision of a bank, a telecom operator or any other independent ID services provider, so that all the system entities have reliable and verifiable registration data used for all SAFE transactions.

IDMS server in our security architecture provides directory and access control services supporting identity management and single sign-on for other components of security infrastructure.

### 3.5.3.3 CA server

CA Server provides management of digital certificates of the system entities. Since in our security architecture transaction security is based on PKI, CA server performs all the activities related to digital certificates. CA server issues digital certificates to the owners of public keys generated by the SAFE mobile application and stored in secure elements of mobile devices. CA server issues X.509 certificates based on registration data provided by IDMS server.

Digital certificates contain public key generated in customers’ mobile devices using specified key generation functions. Generated public key and private key will be stored in secure element of the mobile device through provided interfaces of the SAFE mobile applications.
Customers can request their digital certificates using mobile application functions upon supplement of required information associated with their generated public key. Then, after successful issuance of certificates by the CA server, they may be stored in mobile phones secure elements. After reliable and verifiable registration, certification, and issuance of smart cards, an instance of the SAFE system is ready to support various secure financial transactions.

3.5.3.4 Location-based Authentication (LBA)

As a compliment to the authentication mechanism of our system design, LBA is a special asset to prove system participants’ identities and authenticity by detecting their presence at a distinct location. Location verification mechanism has exploited different specific techniques and uses the advantages of their strengths in order to achieve a sufficient level of confidence. This a “strength-in-depth” method in which different validation and verification methods add their own verification steps and complement each other so that, if one method is bypassed the others can detect and prevent intrusion attempts [37].

Basically, system participants using their smart mobile devices equipped with GPS chips can be capable of determining the coincidence of their identities at their distinct location. Also, there was an attempt for distinctiveness of locating using precise proximity of the participating individuals associated with their location.

LBA system being incorporated in the system architecture is an authentication system that incorporates the location factor using various technologies, such as GPS, Wi-Fi built in a smart mobile device. This system will help to improve security of the authentication process by limiting locations from which an entity is able to successfully authenticate.

LBA system is capable of getting integrated with different compatible system architectures. Here, there are some other components needed to complete system design, which are actually a combination of various servers and related applications [37]:

- Location-based ID (LBID) Server – stores location information and provides registration and verification services handling reliable user identification and location data;

- CA Server – provides management of digital certificates of the system entities. CA server issues digital certificates to the owners of public keys generated by SAFE mobile application and stored in secure elements of mobile devices. CA server issues X.509 certificates based on registration data supplied by IDMS server.

- Authentication Server – provides accurate location-based authentication service for all participants;

- Authentication Server – provides authorization services based on security policies for system authorities and consumers, using secure web services along with accurate location-based authentication service and enforces location-based authorization service based on already defined policies for all participants;
- Service Provider (SP) Server – provides various mobile services based on location-based policies;

LBA system incorporated in our system architecture and its components is shown in Figure 13. System administrators perform system setup in order to establish location-based authorization policies and to certify system components. Merchants are the entities who initially trigger the system and register their locations associated with their products. On the other side, users inquire the system when using mobile applications on their GPS-enabled mobile devices interacting with LBID Server and Authorization Server. Users request access to Authentication Server interacting with LBID Server and Authorization Server about request verification based on the information in LBID Server and authorization policies in the Authorization Server. Then, the targeted SP Server finally allows or denies access request based on the received result.

Security and privacy issues of the incorporated LBS are supported with existing security components of the incorporating system. PKI is mainly used as the security infrastructure of the whole system along with appropriate cryptographic modules. The details of certification issuance and management in a mobile environment (m–PKI) are not described here. As already briefly described, every entity being registered in the system should request its certificate from the CA server in order to perform the required security functions. All certificates are issued by CA Server in a standard way. However, certificates of users have special extension containing client’s location, which is obtained from the LBID Server [37].

![Diagram](image-url)

**Figure 11: Location-based Authentication Mechanism within the System Architecture**

Authentication and authorization protocol is activated every time when a user requests some location-based service from the system. Initially, user sends service request to the gateway
server. GW server directs the request to the SP Server. SP Server checks the request and requests authentication status of the client from the Authentication Server. Next, it sends an authentication challenge to the user. Then, the user provides his/her security credentials to the Authentication Server. If this verification was successful, Authentication Server sends location verification request to the LBC and prompts the user to enter PIN. LBC authorizes location response, determines user’s location and sends it back to the AS within a response message signed by LBC’s private key, encrypted by LBID Server’s public key.

Next, the Authentication Server sends all location information to the LBID Server. LBID Server using its private key decrypts the message and verifies user’s digital signature. After successful verification, LBID server compares retrieved location information with the user’s location records stored during registration through IDMS server and sends back verification result attached to user’s location information to the Authentication Server. Successful authentication triggers Authentication Server to send authorization request, containing user’s access request and user’s verified location information, to the Authorization Server. Then, Authorization Server verifies user’s access request by comparing subject, object, operation and user’s location with associated privileges in the authorization policies and consequently sends authorization result to the SP Server. Finally, the SP Server grants/denies the access for user’s service request based on the authorization result. The authentication and authorization protocol is illustrated below (Figure 11).

![Figure 12: Location-based Authentication and Authorization Protocol](image-url)
The following parameters are considered for sufficient location verification [37]:

- Location coordinates - longitude and latitude and the range of accuracy (in meters) from two different location APIs: operating system and third party location service provide APIs to calculate two sets of location estimates; longitude, latitude and accuracy. These estimates can be obtained by the combination of GPS, Wi-Fi and 3G signals to represent the location of the client within circles with the coordinates as the centers and the accuracy measurements as the radii.

- IP address of the client: Using the IP address of the client’s mobile device, the server can calculate general location of the client using the IP2Location positioning service [38, p. 104].

- MAC address of an access point with the strongest signal: this address can be fetched and reported by the residing client to the server.

In order to verify user’s location, the protocol performs the following steps/checks [37]:

- Whether the two sets of location coordinates from the OS and third party APIs overlap
- Whether any of the location coordinates overlap with the location obtained from the IP address
- Whether the MAC address received matches the one that was previously detected

The location claim is verified only if all the checks above are successfully satisfied. If any failure happens with any one of them, the process will fail and the claim will be rejected.
4. Implementation

In this chapter all details of the implementation of the described system architecture is presented. In Chapter 5 the results of the proposed solution in terms of design and implementation matters are also presented, so that they would be useful in analyzing the overall results of the thesis work.

4.1 The Environment, Platforms and Tools

This section describes the tools, platforms and development environment used to implement the proposed Mobile Payment System. As mentioned earlier, the proposed system architecture includes two groups of components, back-end servers and front-end client applications. The implementation of back-end components does not fit into this report, so here mainly the front-end interfaces are described and including how they interact with back-end components.

4.1.1 Hardware and Software for Development

The development was carried out in Windows workstation environment, with the Eclipse development environment 3.6.x for Java developers enabled with Android ADT 18.0.0 and Eclipse JDT plug-ins. The tool used was Android SDK 3.2 platform.

The installation of the development tools for testing and demonstration purposes was installed on a 2.4 Intel core2Due machine with 4 GB of RAM running Windows 7 Professional.

The real devices being used for debugging and testing Android applications were two Samsung Galaxy S Android Smart phones with 1 GHz ARM "Hummingbird" processor, 16 GB internal flash memory, Wi-Fi and Bluetooth connectivity with Android 2.3 OS. Also, a Samsung Galaxy Tab with features the same as the described smart phones.

4.1.2 Android SDK

Android comprises “an operating system, middleware and key applications” [39]. Android SDK provides necessary tools and APIs to develop applications Android platform using Java programming language. In order to build mobile applications for Android devices, Android is used which provides a SDK. These tools can be accessed through an Eclipse plug-in called ADT (Android Development Tools) or from the command line. The ADT plug-in for Eclipse provides a smooth Java development environment with advanced features for build, test, on-device debug, and package the desired Android apps. Upon setting up the development environment it was possible to create Android Virtual Devices (AVDs) on which applications can be installed.

When building mobile applications, they always were being tested and debugged both on simulator and real mobile devices before releasing them to users. Actually, the emulator basically does not provide testing every device feature (such as Bluetooth, GPS and the accelerometer).
4.1.3 SAFE Administration Station

SAFE Administration station performs management of SAFE system entities in connection with back-end databases and directories. The primary functions of SAFE Administration station are the following:

4.1.3.1 Registration of Entities

SAFE Administration station provides registration function for registering all entities and management entity’s data. There are different kinds of entities in the system, such as banks, customers and accounts. Hence, there are specific registration processes for them. Registration of Customers and Account is also possible through SAFE mobile applications [32].

Figure 13: The functions of registration of different entities [32]
4.1.3.2 Transactions Management

SAFE Administration station also supports transaction management services which keep the logs containing all entities’ operations involved in SAFE transactions and messages. This feature could be useful for bank administrators to audit and trace customer transactions and message flows [32].

4.1.4 SAFE Mobile Applications

As mentioned above, as one of the components of the SAFE system, mobile applications are considered as front-end interfaces representing secure, stable, convenient and user-friendly experience for end-users, pre-loaded into their mobile devices. This section describes design, implementation and demonstration of SAFE mobile applications.

There are three versions of SAFE mobile applications. Each one provides specific services for its associated users: SAFE Wallet, SAFE Merchant and SAFE Agent. Overview and demonstration of these applications are given below.

SAFE mobile applications must be initially configured for smooth functioning and message passing. For this purpose in an initial sequence, the authentication credentials, including his/her mobile number and SAFE system configurations, are required to be set on initial lunch of applications to personalize the applications and activate the configuration of the SAFE Server.

SAFE mobile applications interact with back-end components via SAFE commands for every specific function. Every command is encapsulated in a common message format containing all information required to settle the initiate transaction.

<table>
<thead>
<tr>
<th>Object</th>
<th>Function</th>
<th>Amount</th>
<th>Target</th>
<th>Sample</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account(a)</td>
<td>Open(o)</td>
<td>N/A</td>
<td>[Currency]</td>
<td>ao EUR</td>
<td>Request to open an Euro account</td>
</tr>
<tr>
<td>Airtime(at)</td>
<td>Transfer(t)</td>
<td>[Amount]</td>
<td>[Target ID]</td>
<td>att 100 1234567890</td>
<td>Transfer 100 unit airtime to the target account</td>
</tr>
<tr>
<td>Money(m)</td>
<td>Deposit(d)</td>
<td>[Amount]</td>
<td></td>
<td>md 100</td>
<td>Deposit 100 units in cash-over-the-counter transaction</td>
</tr>
</tbody>
</table>

Table 1: SAFE commands and their descriptions [32]
Client Message: transaction command
Wallet Message: client message added with client’s mobile phone number
SAFE Message: wallet message added with security fields
GW Message: Gateway Message that is SAFE message added IP header, ready to send to Gateway Server
Phone Number: client’s mobile phone number
DS [Hash]: Digital Signature of the hash value
PHL (2): phone number length
WML (2): wallet message length

**Figure 14: SAFE Message Format** [32]

Security of messages and consequently transactions is provided by three layers. The first security layer is provided by SAFE system indicated by users’ preferences. The second security layer could be accomplished using digital certificates issued by SAFE authorized CA servers and, finally, security modules provided by SAFE mobile applications and security hardware of mobile devices.

**Figure 15: SAFE Message Flow** [32]

Users may select various security options for protection of their messages: No Security, Data Integrity only, Data Confidentiality only, or both – Data Integrity and Data Confidentiality.

By selection of either of these options, users indicate a level of security for their mobile transactions. Users’ preferences affect the SAFE messages format and notify the SAFE gateways to enforce requested security measures.
**Integrity only** – by selecting this option the security header is set to INTEGRITY. Actually you indicate that you need to provide the integrity of the outgoing and incoming messages. The procedure is described and illustrated as the following: A Message Digest of a typical message is created using available hash functions (eg.SHA-1). Then, the Message Digest is signed with the sender’s private key and packaged to be sent to the purposed recipient. At the receiving end, the recipient will inspect security header to check “security option”. Next, recipient calculates Message Digest using mutually agreed hash functions and compares it with the received Message Digest created upon decrypting “digital signature” with sender’s public key [32].
Encryption only – by selecting this option the security header is set to ENCRYPTION. Actually you indicate that you need to provide the confidentiality of the outgoing and incoming messages. The procedure is described and illustrated as the following: A Message encryption key (MEK) is randomly generated to encrypt message content in order to provide message confidentiality. Then MEK is encrypted with recipient’s public key and attached to the encrypted content and finally, it is packaged and sent to the recipient. At the receiving end, recipient will inspect security header to check “security option”. With ENCRYPTION mode, recipient extracts the incoming security package and decrypts MEK with its private key and consequently decrypts encrypted message using the decrypted MEK [32].
Fully Protected — by selecting this option the security header is set to FULLY_PROTECTED. Actually you indicate that you need to provide both the confidentiality and integrity of the outgoing and incoming messages. The procedure includes two steps: to sign and to envelop. In the first step, message is signed by calculating Message Digest and then signing it using the sender’s private key to get Digital Signature, and in the second step, Message Encryption Key (MEK) is generated to encrypt both the message content and digital signature. Then MEK is encrypted with recipient’s public key, packaged.
and sent to the recipient. At the receiving end, recipient checks security header for “security option”. With FULLY_PROTECTED mode, recipient decrypts MEK using its private key and using MEK decrypts the received message containing digital signature of the sender and actual message content. Finally, recipient verifies digital signature using sender’s public key in order to confirm message integrity and authenticity [32].

In addition to the first layer security provided by the SAFE system, an end-to-end security is also provided using asymmetric crypto functions. In fact, users need to request their digital certificates from the SAFE systems and then load and store them in their devices’ secure element using SAFE applications. In this way, users would be able to prevent data from being revealed to the Gateway Server. It is working like this: user first encrypts sensitive data with
his/her own randomly generated secret key, then envelopes that key using public key of the service provider and attaches to complete package an unique identification (IP address or mobile number), so that it can be verified by the Gateway Server and then sends the message to the Gateway Server [32].

![Diagram](https://via.placeholder.com/150)

**Figure 22: Security of Messages [32]**

### 4.1.4.1 Digital Certificates Management

As mentioned in earlier chapters, our architecture includes use of mobile PKI as a potential to provide authenticity of SAFE entities and messages in a secure user-friendly manner. This PKI implementation employs a typical secure element (SE) present in existing smart phones. This SE contains key information which could be used primarily to authenticate an entity of the SAFE system. SAFE applications enable users to make use of X.509 certificates to authenticate themselves and to digitally sign SAFE messages when accessing SAFE services.

In addition to the secure elements and smart phone standards, there are also some standards related to authentication and digital signatures. Such standards require cryptographic processing and corresponding key management methods. In fact, PKI basically employs X.509 certificates according to the RSA’s Public Key Cryptography Standards (PKCS). Digital certificates can be used for cryptographic and authentication purposes and digital signatures. With this m-PKI, each mobile device will contain a private key only in its secure element, so that it can be used for protecting of transactions; and a public key that is encapsulated in user’s corresponding digital certificate. SAFE Certificate Authority (CA) issues these certificates and the CA’s signature can be verified through a certificate chain by anyone using the CA’s root certificate.

Based on the type of the CA in the SAFE system and limited computation capabilities of mobile devices, X.509 certificates could be obtained in a way as described below. Following standard approach, SAFE CA issues certificates typically by receiving corresponding subject identifiers and submission of certificate signing request (CSR). This CSR follows RSA standard specification PKCS #10 [40]. Actually, the “PKCS#10” implementation produces a DER-encoded PKCS #10 certificate request. For this purpose, Oracle crypto modules have been used (oracle.security.crypto.cert.CertificateRequest) [41].

The request basically contains subject identifiers, public key chosen by a mobile user, and it is signed by the user's private key. Public key in the request would be used to verify the signature, while the signature of the request is verified on the request submission. Upon the public/private key pair generation, the private key should be stored in mobile device’s secure element. Access to the key is solely restricted to user’s SAFE application interfaces, so that it
supports non-repudiation security service. After successful certificate request handling, the CA will send back a certificate within a “PKCS#7” message format [42] that has been signed with the private key of the CA which is verified both on proxy and mobile user sides. Then, proxy extracts the received certificate form PKCS#7 message and forwards the certificate to corresponding subject owner. Finally, upon receiving certificate on a mobile user’s side, it is stored in the SE for required security purposes of SAFE applications.

Since the SAFE CA is internal to SAFE system domain, there are some additional options that reduce overheads required to process requests both on server side and client side. Hence a proxy server is considered between the mobile user and SAFE CA, so that it receives all the information for CRS from user in a compromised format and protocol and then, after extracting required information for CRS, it continues with handling the PKCS messages to the SAFE CA. In other words, proxy acts as an intermediate component, which translates the request and response between mobile user’s and SAFE CA. Figure 17 illustrates the process of a subject requesting and issuing X.509 certificates with SAFE CA that processes certificate requests.

The following code snippets are parts of security module which actually implements certificate handling described in the Figure above, initiating from key pair generation up to to receiving the issued certificate from the SAFE CA server.

4.1.4.1.1 Key pair generation

Key pair generation is simply done by implementing JCE class libraries as follows:

```java
// Here we want a RSA key generator using its algorithm provided by JCE
KeyPairGenerator keyPairGen = KeyPairGenerator.getInstance("RSA");
// Initialization of the generator with a random
SecureRandom rnd = SecureRandom.getInstance(rndAlg);
keyGen.initialize(1024, rnd);
// Generate a new key pair
KeyPair kpair = keyPairGen.generateKeyPair();
```
Next, both generated keys are being saved in a file and finally stored in the SE. Then Registration Request will be composed as the following. Actually, Registration request uses the generated RSA public key’s exponent and modulus required for Proxy Server. The reason for extracting exponent modulus is that Proxy Server needs to reproduce public key with a different representation to create a required certificate structure and PKCS#10 certificate request.

```java
Socket socket = new Socket("proxy ip address", port);
DataInputStream in = new DataInputStream(socket.getInputStream());
DataOutputStream out = new DataOutputStream(socket.getOutputStream());

// Composing Certificate registration request for proxy
String strBuffer = strBuffer + "RequestPKCS10" + "|";
String userDN = "US|MD|Silver Spring|SETECS Inc|IT|Security Administrator|localhost|";
strBuffer = strBuffer + PrintableCoding.encode64(userDN.getBytes()) + "|";
String userDN = "US|MD|Silver Spring|SETECS Inc|IT|Security Administrator|localhost|";

// Fetching the public key and corresponding Exponent and Modulus required for proxy and PKCS#10 request
KeyFactory factory = KeyFactory.getInstance("RSA");
RSAPublicKeySpec pub = factory.getKeySpec(kpair.getPublic(), RSAPublicKeySpec.class);
pubkExponent = pub.getPublicExponent();
exponentBytes = pubkExponent.toByteArray();
pkMudulos = pub.getModulus();
java.math.BigInteger m = new java.math.BigInteger(1, pkMudulos.toByteArray()).toByteArray();

pkMudulosstr = PrintableCoding.encode64(pkMudulos.toString().getBytes());
pubkExponentstr = PrintableCoding.encode64(exponentBytes);
strBuffer = strBuffer + pkMudulosstr + "|" + pubkExponentstr;

// -- receive the data from server to encrypt
String dataToEncrypt_str = in.readUTF();
dataToEncrypt = PrintableCoding.decode64(dataToEncrypt_str);

// padding
byte[] paddedData = new byte[128];
paddedData[0] = 0x00;
paddedData[1] = 0x01;
```
for (int i = 0; i < 128 - 3 - dataToEncrypt.length; i++) {
    paddedData[2 + i] = (byte) 0xFF;
}
paddedData[128 - 1 - dataToEncrypt.length] = 0x00;
for (int i = 0; i < dataToEncrypt.length; i++) {
    paddedData[128 - dataToEncrypt.length + i] = dataToEncrypt[i];
}
// -- encrypt the data--------
dataEncrypted = encrypt(dataToEncrypt, kpair.getPrivate());
// -- send encrypted signature back to server to be packaged
out.writeUTF(PrintableCoding.encode64(dataEncrypted));
// --- read pkcs10request
pkcs10request = in.readUTF();
pkcs10requestStr = PrintableCoding.encode64(pkcs10request.getBytes());
CertReq = "RequestCertificate|" + pkcs10requestStr + "|136";
out.writeUTF(CertReq);
// --- read Certificate
in.readUTF();

Finally, after fetching the certificate, it will be stored in a file in SE.

Certificates.this.startActivityForResult(intent, GET_FILE_LOCATION);

int duration = Toast.LENGTH_LONG;
Toast toast;
File[] roots = File.listRoots();
File root = roots[0];
File sdcard = new File("/sdcard");
String[] sub = sdcard.list();
CharSequence text = "";

for(int i =0; i < sub.length; i++){
    text = text + "\n" + sub[i];
}
toast = Toast.makeText(context, text, duration);
toast.setGravity(Gravity.CENTER, 0, 0);
toast.show();

Intent intentBrowseFiles = new Intent(Intent.ACTION_GET_CONTENT);
inentBrowseFiles.setType("/*");
inentBrowseFiles.addCategory(Intent.CATEGORY_OPENABLE);
Certificates.

```java
this.startActivityForResult(intentBrowseFiles, GET_FILE_LOCATION_REQUEST);

String filename = "filename";
FileOutputStream file;
file = new FileOutputStream(filename);
file.write(currentCert.getEncoded());
file.close();
```

4.1.4.2 Secure Element

Secure Element is basically a tamper-resistant secure storage of security secrets and extra sensitive data needed in order to maintain the security of the containing system in such way that it is very difficult to compromise. This is needed to be achieved for mobile payments in accordance with security rules and requirements ideally specified by a set of well-identified rusted authorities. So, Secure Element is required to be securely in connection with security management module of mobile applications. This actually ensures security between mobile application and users’ information stored on a mobile device. Secure Element could be placed in different possible places but, here it is in memory card that can be inserted in a smart phone. In the Implementation Chapter, it is described how SE would be used with mobile applications.

4.1.4.3 SAFE Agent

SAFE Agent application is being used by agents, merchants or any other member of the SAFE system, authorized by the system administrator to perform the functions of an agent. Those functions are:

- Self-registration and registration of subscribers and merchants, including registration of locations for merchants

  ![Figure 24: Self-registration Sequence](image)

- Cash-in and cash-out financial transactions with subscribers and merchants. When performing cash-in and cash-out transactions, user receives four digits authorization code and tells it to the agent and Agent enters the code to confirm cash transaction.
Mobile security functions: registration of an agent, management of agent’s local and SAFE system PINs, selection of cryptographic options, and managing X.509 digital certificates.

If an individual, who is the customer of the SAFE system, acts also as an agent, he/she must use SAFE Wallet application for its own financial transactions and use SAFE Agent application when performing agent’s transactions. The same with merchants: if they perform their own transactions, they use SAFE Merchant application, while if they act as an agent, they use SAFE Agent application.

4.1.4.4 SAFE Wallet

SAFE Wallet application supports three groups of mobile functions:

- Various types of mobile payments using mobile pre-paid accounts, standard bank accounts, bankcards and digital cash payments in OTA and OTC mode. Wallet application provides an interface for customers to perform various types of payment using different types of accounts that SAFE system supports. Using Wallet application, customers can...
use a specific type of payment based on the situation and available capability of host mobile device.

Figure 27: SAFE Wallet Mobile Payment Functions

Wallet application enables its user to register and create associated pre-paid SAFE accounts as well as credit/debit card registration to provide most of usual payment methods.

Users can perform deposit, withdraw and transfer transactions using their pre-paid SAFE mobile accounts between all SAFE account owners and then list all launched transactions filtered by defined constraints.
Upon any successful deposit or withdrawal, customer receives an authorization code and tells that to Agent, who uses it to confirm the transaction.

Wallet application provides also management of user’s SAFE accounts. Users can perform the following actions to manage their SAFE mobile accounts:

- List SAFE Accounts – lists all user’s SAFE accounts registered in the system
- SAFE Account Balance – gets balance of the selected SAFE account
- Suspend Account – temporally disables the use of the account
- Activate Account – enables the use of an account after being suspended
- Terminate Account – permanently removes the account from the system

![SAFE Wallet Account and Transactions Functions](image.png)

**Figure 28: SAFE Wallet Account and Transactions Functions**

- Various types of mobile banking functions having bank’s IT Server connected to SAFE system directly or through some banking switch. Users can perform the following transactions using bank accounts:
  - Register Bank Account – register bank account in the SAFE system
  - Pay Bill/Invoice – pay bill (individuals) or pay invoice (companies) using bank account
  - Transfer Funds – transfer funds from one bank account to another
  - Account Balancer – inquire the balance for the selected account
  - List/Delete Bank Accounts – list all bank accounts registered in the SAFE system
  - List Transactions – list the last five transactions performed with the selected bank account
Various types of bank cards functions having some card transaction processing gateway connected to SAFE system directly. In fact, payments can also be performed by selecting one of the registered bankcards in the m-Pay function. Users can perform the following transactions using bank accounts:

- Register Card – register bankcard.
- Check Card Balance – inquire balance for the selected credit card
- List/Delete Cards – list registered bankcards and remove those that are no longer valid
- List Transactions – list the last five transactions performed with the selected bankcard

Digital cash enables users to perform the following transactions with stored e-cash and view e-cash stored in the Wallet:

- Load Cash into Wallet – transfer cash from the SAFE mobile account into Wallet
- Unload Cash from Wallet – transfer e-cash stored in the Wallet back to the SAFE mobile account
- View Stored Money – view the amount of e-cash available in the Wallet

Various types of mobile shopping functions: mobile promotions and advertisements, mobile coupons, mobile gift-cards, mobile parking and mobile tickets. Users can perform the following functions:
- **m-Advertisements** – select and download announcements. Mobile advertisements can be loaded into Mobile Marketing Server of the SAFE system by merchants, using m-Marketing functions of the SAFE Merchant mobile application. Advertisements loaded to the Mobile Marketing Server can be filtered by user’s location.

- **m-Promotions** – select and download promotions with various discounts and benefits. Mobile promotions can be loaded into Mobile Marketing Server of the SAFE system by merchants, using m-Marketing functions of the SAFE Merchant mobile application. Promotions have QR bar code which is used to present it to the merchant’s SAFE Merchant application for mobile clearance, using merchant’s mobile device as mobile PoS device.

- **m-Coupons** – select and download mobile coupons

- **m-Telecom** – inquire, select, purchase air-time. Through mobile telecom menu, telecom providers will be able to upload their airtime plans and in that way providing possibility to users to review various airtime plans, search for plans, select one plan and pay for it. This group of mobile services requires on-line connection to telecoms offering airtime plans.

- **m-Tickets** – inquire, select, purchase, transfer and use mobile tickets. Tickets can be paid using any SAFE payment method – SAFE accounts, bank accounts, bankcards, or e-cash. Downloaded tickets have QR bar code which is used to present it to the merchant’s SAFE Merchant application for mobile clearance, using merchant’s mobile device as mobile PoS device.

- **m-Parking** – search for parking places, pay parking using SAFE account and receive expiration notifications. It is required that SAFE system is integrated with the parking system of some Parking Authority. m-Parking supports the following functions for mobile parking:
  - **Search Parking** – users may search available parking place at the local garage/parking location by giving its registration number
- Pay Parking – users may pay for parking by specifying parking place, parking time and selecting method of payment.

- Extend Parking – the system will warn users about expiration of their parking time. In such case, users may extend parking by specifying parking place number and additional parking time.

- Pay Ticket – users may use SAFE system to pay tickets for parking violations

  - m-Gift Cards – inquire, select, purchase, transfer and use mobile gift cards

Customers can use downloaded coupons and gift cards in their payments via m-Payment functionality.

- Mobile security functions: registration of a user, configuration of the SAFE system, detection of locations, management of Wallet’s local and SAFE system PINs, selection of cryptographic options, and managing X.509 digital certificates.

![Security Settings](image1)

**Figure 32: M-Security Functions**

### 4.1.4.5 SAFE Merchant

SAFE Merchant application supports three groups of mobile functions:

![SAFE Merchant](image2)

**Figure 33: SAFE Merchant Main Menu**

- Mobile payments with subscribers using over-the-air (OTA) SMS and wireless protocols and over-the-counter (OTC) Bluetooth protocols. Using merchant application, OTC
payment takes place based on Bluetooth and NFC proximity protocols and OTA using wireless protocol either provided by the Internet providers or telecom carriers.

- Various types of mobile marketing functions: creating and uploading promotions, mobile coupons, mobile gift-cards, and mobile tickets into the SAFE system marketing server. SAFE Merchant supports the following m-Marketing functions:

  ![m-Marketing Functions](image)

  **Figure 34: SAFE m-Marketing Functions**

  - m-Advertising – upload advertising messages to the Mobile Marketing Server and view/remove uploaded advertisements
  - m-Promotions – upload promotions to the Mobile Marketing Server and view/remove uploaded promotions
  - m-Coupons – upload mobile coupons to the Mobile Marketing Server and view/remove uploaded coupons
  - m-Telecom – upload various airtime plans
  - m-Tickets – upload tickets to the Mobile Marketing Server and view/remove uploaded tickets
  - m-Gift Cards – upload mobile gift cards to the Mobile Marketing Server and view/remove uploaded gift cards

- Various types of mobile business functions: accepting discounts based on promotions, mobile coupons, mobile gift-cards, and verification of mobile tickets. Also, providing mobile transactions in locations using PoS systems, then mobile services for the owners/drivers of vehicles, and various mobile commerce transactions and accepting and clearing mobile vouchers from customers, uploaded by merchants using m-Marketing services. SAFE Merchant supports the following m-Business functions:
Figure 35: SAFE m-Business Functions

- M-POS System – mobile functions with PoS systems: create inventory, review inventory, create quote, create order, and create check. Mobile PoS services can be used by merchants that have PoS system, like restaurants, supermarkets, bookstores, small shops, etc. This subsystem provides the following functions:

  - POS Server – configure PoS Server
  - Sale Items – create and manage inventory of items on the PoS Server (goods for sale in shops, etc.)
  - Create Order: to be presented to the customer containing selection of sale items
  - Create Invoice: based on the confirmed order, create invoice to be paid by the customer
  - List Transactions: list all orders and invoices for some period of time
  - M-Vehicles: mobile functions for owners of motor vehicles: pay registration fees, pay toll, pay parking, pay for gasoline
  - M-Commerce: verification of mobile vouchers created and uploaded by m-Marketing functions

- Mobile security functions: registration of a merchant and administration of and access to SAFE mobile pre-paid accounts, configuration of the SAFE system, detection of locations, management of Merchant’s local and SAFE system PINs, selection of cryptographic options, and managing X.509 digital certificates.
4.1.4.6 Secure Mobile Transactions

Describing all functions of SAFE applications is out of scope of this report and you can refer to SAFE application users’ guide. So, some major payment transactions are described below and discussed in terms of security.

Basically in SAFE mobile applications, there are two types of mobile payment protocols between merchant and customer: Over-the-counter (OTC) and Over-the-air (OTA). Currently, SAFE applications support OTC based on Bluetooth protocol while using other proximity protocol is considered as future work. OTA protocol is naturally based on the Internet connectivity (GPRS /3G / 4G /Wi-Fi).

The sequence of steps for both protocols differs in client’s side security modules, but the rest of steps are the same, since both protocols communicate with SAFE servers using the same language.

![Mobile Client Application Structure](image)

**Figure 37: Mobile Client Application Structure**

As described earlier, SAFE mobile applications are implemented based on the internal structure shown above. OTC payment uses Bluetooth communication method in addition to the Internet connection, while OTA payment supports Internet connection. So, communication module recognizes all the communication hardware on mobile device and then based on supported application functions handles communication request either between mobile devices or mobile device and remote gateways.

Business module includes the logic about the transaction protocols and management of communication modules and security module.

Software security module basically provides security services for SAFE applications and manages the access to available hardware security modules (secure elements, etc.). The dedicated security module consists of all libraries, certificate, encryption keys and API compatible with system security modules and components. Security module provides mainly the following functions:
• Storage of sensitive security data required for security operations

• Cryptographic operations consisting of key generation and management, certificate management, signature and encryption: Cryptographic operations are mainly implemented and supported by Java Cryptography Extension (JCE), but certificate management (certificate request and response, etc.) comprises a method which could be applicable on mobile devices with low processing power. This method is described in details below.

• Identification and authentication of available security and communication devices of its surrounding environment: security module uses and implements stable security function interfaces and supported libraries provided approved by trusted companies and vendors to interact with mobile hardware security modules and communication hardware in mobile device. Security module uses “Secure Element Evaluation Kit” for the Android platform which provides API to access the SE (secure SD card). This API uses the Android permission, so the relevant security module function must request in order to obtain access. However, there are some security issues with this Kit which are described in the last chapter. The benefits of using this Kit for SAFE applications are the following [43]:

  • Access control supports certificates handling to provide specific access for each Secure Element application
  • APDU filtering based on client certificates which assigns specific list of allowed APDU commands
  • Based on open standards the access control scheme that can be deployed with any Java Card based Secure Element on the market
  • Easy to implement with small memory footprint on the Secure Element

Below, the sequence of OTC and OTA from SAFE Merchant and SAFE Wallet to SAFE gateways are shown and described with relevant implementation codes for each function and step.
OTC payment uses Bluetooth proximity protocol to initiate the transaction. SAFE applications establish BT connection based on the security module, so that it applies some security restrictions to the default BT of Android devices, so that no other device can interfere between customer and merchant application.

Basically, Bluetooth connection between Wallet and Merchant devices should be a client-server connection, so that you must implement both client-side and server-side scenarios. Server device (Merchant) opens a Bluetooth server socket while client device (Wallet) initiates the connection using server’s MAC address. This connection needs to be established on the same RFCOMM channel between client and server, so that each device can perform data transfer obtaining input and output streams [44].

The restriction with BT connection is UUID. Server device gets a BluetoothServerSocket by calling the listenUsingRfcommWithServiceRecord(String, UUID). The client devices require the UUID as the basis for their connection agreement. When the clients attempt to connect to each other, their UUIDs must match so that, they uniquely identify the service with which they wants to connect. Hence, both SAFE applications must have the same UUID, so that they can establish BT connection.

As you can see below, SAFE applications generate UUIDs based on hash code of signature of the authority which releases the SAFE applications, so that only SAFE applications can have such BT connections.

```java
PackageInfo pkgInfo = 
getPackageManager().getPackageInfo(getPackageName(),
PackageManager.GET_SIGNATURES);

android.content.pm.Signature[] sigs = pkgInfo.signatures;
for (Signature sig : sigs)
    uuid = UUIDGenerator.getUUID(sig.hashCode());
return UUID.fromString(uuid);
```

### 4.1.4.6.1 Over-The-Counter Payment Transaction

The sequence with OTC payment in SAFE applications is basically initiated by the merchant using SAFE merchant application. As the Merchant activates m-Pay button, application triggers the available proximity protocols (Bluetooth) upon user’s permission. After approving the permission, SAFE merchant starts as a Server for handling incoming SAFE Wallet payment requests. Otherwise, it goes on with OTA payment protocol. Considering approving the Bluetooth permission, OTC gets started. This also happens on customer’s Wallet application.
Now, Merchant is in waiting mode pending for incoming payment requests. As it receives payment request from Wallet, starts to establish the Bluetooth connection and compose the payment response for Wallet payment request, otherwise starts the OTA transaction.
Upon successful BT connection, Wallet waits for Merchant payment data.

After filling in the payment form, Merchant sends it as a payment data including the amount and Merchant’s SAFE account to customer’s SAFE Wallet and then it waits.

When Wallet receives payment amount from Merchant, it can select the payment type: SAFE account, bank account or bankcard. Then it can add some more to received amount as a tip and select either of existing accounts for payment and finally approve the payment by pressing Pay button.
By pressing Pay button, SAFE Wallet sends all the payment information in a SAFE message format to SAFE system. Then SAFE system performs the transaction and finally informs the customer about successful completion of the transaction. Then, customer should approve the result received from SAFE system, so that it can inform pending Merchant about transaction result using the already established BT connection.

Next, Merchant receives payment confirmation message from the connected Wallet, which displays transaction result and then, merchant asks the customer if he/she needs the receipt and waits for Wallet reply.
Customer can accept the receipt request or not. If it rejects the request, the transaction is completed on the Wallet side but, if it approves the request, Wallet asks the Merchant for the payment receipt and stands waiting.

Finally, Merchant can approve customer’s request for the payment receipt by pressing “Send Receipt” and sends the receipt to the customer via BT connection then Wallet receives it and payment procedure is over.
4.1.4.6.2 Over-The-Air Payment Transaction

The sequence with OTA payment in SAFE applications is basically initiated by customer using SAFE Wallet application. In this protocol merchant tells the customer about the payment amount and then merchant is informed by checking its SAFE account status. As Wallet starts the payment procedure through m-Pay button, application triggers the available proximity protocols (Bluetooth) upon user’s permission. After rejecting the permission, it performs with OTA payment protocol.

Customer needs to select a payment type and then enter the amount to pay and probably a tip and then merchant tells the customer to which SAFE account the payment should be made specifying merchant’s mobile number.

Customer needs to select a payment type and then enter the amount to pay and probably tip and then merchant tells the customer to which SAFE account the payment should be made by specifying merchant’s mobile number. Next by pressing “Pay” button sends payment message to SAFE system.
SAFE system performs the transaction and transfer of money between customer and merchant SAFE accounts and then notifies the customer about the transaction result. When the customer receives the transaction result, he/she presses the “OK” button and completes the transaction procedure.

![SAFE Payment System](image)

*Figure 52. Transaction Result*
5. Analysis and Discussions of the Solution

This chapter discusses the solution presented in this thesis. The analysis is based on results achieved in view of the goals and purposes mentioned for the work in the first chapter considering some of the delimitations and strength of the solution.

5.1 Meeting the Goals

Here, the results of the thesis project are compared to the goals promised earlier in the first chapter.

As mentioned and emphasized earlier, the ultimate purpose of this thesis was proposing a system for secure mobile payment transactions using a new method to provide efficient security for payment information messages using existing mobile payment models. Hence considering general characteristics of mobile payment models, a typical m-commerce system involved with a specific payment model was adopted based on underlying protocols and methods of the system. To reach the purposed e-commerce system, the following objectives are achieved.

According to current and standard characteristic of mobile commerce systems, there are methods and protocols introduced and implemented, so that the applied payment model could be understood comprehensively in terms of efficiency, usability and security.

Moreover, a typical architecture for mobile commerce and financial transactions has been adopted and implemented. This architecture basically comprises components, protocols and interfaces providing several services to SAFE mobile applications: registration, security of users at different levels, and protection of its own components. The architecture was designed so that it is modular, integrated, extendible and scalable. In addition to all objectives above, the method of secure payment proposed and implemented to enhance all payment and commerce services in terms of security.

The integrated e-commerce system provides a balanced trade-off between main issues of mobile payment standards for mobile users to perform convenient e-commerce transactions in a simple, fast and secure way anytime and anywhere based on adopted location based services.
5.1.1 Functional

According to Figure 9 in Chapter 3 it is been tried to fulfill the function criteria for a standard mobile terminal for e-commerce applications:

- **User interface** provides access control for user’s using remote (Location-based authentication and authorization) and local access control (encryptions by security module) mechanisms. Moreover, users can indicate and compose their requesting using enhance forms provided by mobile applications and specify the security level indication using security settings of SAFE applications.

- **E-commerce protocols** play the key functions in the whole system, so it has been tried to include as many functions required based on users’ needs. Shopping protocols are designed so that they imply availability and convenience for users. Payment protocols are also designed so that they can provide the required measures as both experts and typical users may expect. In fact payment protocols could be initiated even standalone and independent of other triggers, but also shopping protocols are implemented so that they need to initiate payment protocol to fulfill their functional requirements.

- **SAFE Applications** identifies available proximity hardware on mobile device and provides interfaces for users

- **SAFE Applications** can operate without proximity services on mobile device.

- **SAFE Applications** can operate without the Internet connectivity and can function relying on available MNO and SMS service.

- **SAFE Applications** can adapt themselves to containing mobile device, so that they enable functions based on availability of services (Wi-Fi, 3G, Bluetooth or NFC, etc.).

- **The system** identifies and supports available air communications for transaction services.

5.1.2 Technical

In the following the technical features required to fulfill the functional specifications are described:

- **SAFE system** performs the operations initiated from the client side applications, so that all transaction commit and notifies the users about the results.

- **SAFE Applications** are developed using Android Development Environment applicable for Android phones. The applications were run on different Android mobile phones without faults. The applications are backward compatible and developed, so that they can run on different Android phones which may use different Android versions.
5.1.3 Security

As explained earlier, architecture is mainly designed to implement a secure two-party payment model for mobile payments, so that the involving parties are typically a customer and a payment service provider. Since, some current security solutions at the communication layer in mobile environments are not adequate, it is concluded that security at the application layer must be added in order to achieve an end-to-end protection for SAFE transactions. In fact, the application layer’s security is considered independent of the lower layers’ security protocols. So, the architecture is designed in such way that it is possible to isolate the SAFE applications from in-device access by protecting it with application layer security, proximity protocol security, and secure element configuration. Hence, SAFE applications could handle all the security-related functions free of any modification to the SAFE system’s communication infrastructure and protocols.

Moreover, the security architecture allowing access to SAFE system resources through the specifically provided interfaces by SAFE applications. The system architecture is designed so that all accesses to SAFE data and services pass through all of the layers from SAFE applications up to SAFE system architecture layers. In fact, the security architecture only allows services in the application tiers to access the lower level SAFE service set.

In the following the features required and accomplished to fulfill the security specifications mentioned above are described:

✓ SAFE Applications uses a security module to provide security functions for all over the application functions

✓ SAFE Applications use digital certificates to provide security services for all SAFE commercials functions

✓ SAFE Applications communicates with secure element in mobile devices and stores sensitive security keys and information

✓ SAFE Applications applies security protocol for proximity protocols available in mobile devices.

✓ SAFE Applications use a location-based authentication system as a complement for authentication purposes, when required.

5.2 Future Work and Development

In this thesis work it is been tried to provide desired and proposed features and functionalities for a Mobile Payment System to provide the essential level of secure payment service. However, how much this payment system will be successful considering technologies, methods and protocols, remains to be decided by the markets and consumers. Thus, there are
still some important features required to be designed and developed to accomplish a more secure mobile payment system with high performance and functionality. In below, the areas considered for further development have been discussed.

**Digital Certificates:** Digital certificates and public-key cryptography play a key role in providing enhanced level of authentication and privacy in the security architecture. Basically, digital certificates being issued and used in this architecture are based RSA signature algorithms. As a future development, there should be a change in security and performance requirements preparing a shift to pure mobile computing for smaller and faster signatures. Hence, by now, it is been recommended to use digital certificates based on Elliptic Curve Cryptography (ECC)-based signatures.

ECC-based certificates use ECC-based signatures which are smaller and faster to create. Accordingly the public keys are smaller and more agile. Also, using these kinds of certificates makes the verification faster in higher key strength [45, p. 21]. So, the strength and small key size can bring a considerable performance and security benefits for mobile computing applications [46, p. 161]. This feature can significantly affect the certificate handling in SAFE application against SAFE back-end system in terms of high performance and lower processing time and less complexity. In table below the performance of RSA and ECDSA is demonstrated in terms of signing and verifying [47].

<table>
<thead>
<tr>
<th>ECC key size</th>
<th>RSA key size</th>
<th>Symmetric Key</th>
<th>ECDSA Sign (digits/min)</th>
<th>RSA Sign (digits/min)</th>
<th>ECC Benefit (digits)</th>
<th>ECDSA verify (digits/min)</th>
<th>RSA verify (digits/min)</th>
<th>ECC Benefit (verify)</th>
</tr>
</thead>
<tbody>
<tr>
<td>224 bit</td>
<td>2048 bit</td>
<td>3-DES</td>
<td>106840</td>
<td>2940</td>
<td>3600%</td>
<td>47520</td>
<td>26880</td>
<td>177%</td>
</tr>
<tr>
<td>256 bit</td>
<td>3072 bit</td>
<td>AES-128</td>
<td>54000</td>
<td>480</td>
<td>11250%</td>
<td>22800</td>
<td>11280</td>
<td>202%</td>
</tr>
<tr>
<td>384 bit</td>
<td>7680 bit</td>
<td>AES-192</td>
<td>30960</td>
<td>60</td>
<td>51600%</td>
<td>11040</td>
<td>2160</td>
<td>511%</td>
</tr>
<tr>
<td>521 bit</td>
<td>15360 bit</td>
<td>AES-256</td>
<td>14400</td>
<td>60</td>
<td>24000%</td>
<td>5280</td>
<td>480</td>
<td>1100%</td>
</tr>
</tbody>
</table>

Table1: Signing/Verifying Performance of ECDSA vs. RSA

Overall, the technical aspects of ECC-based digital certificates can be a better choice for mobile applications, now and even in the future. In addition to general advantages listed above, they also have a wide range of possible uses in future application in future markets.
References

[28] T. Lan, "Secure mechanism based on concurrent signature for mobile payment services," in Communication


M. Bishop, Introduction to computer security, Addison-Wesley, 2005.


