Bachelor Thesis

Smart Card Based Solutions for Secure Internet Banking with a primitive reader or mobile phone

Feng Zhu

January 31, 2009

Abstract

Internet Banking is performed in an insecure environment, that is, both the PC and the web browser may have viruses and spyware. Therefore the sensitive data such as client passwords and financial transactions can be eavesdropped and modified. This paper presents two smart card based solutions for Internet Banking. Unlike most of earlier approaches which handles all the sensitive data using the web browser, both proposed solutions provides an trusted system which processing the sensitive data using a smart card with a micro-processor and on-card operating system. The difference between two solutions is that the first one uses a primitive smart card reader (connected to the PC) with the trusted display and keypad, whereas the other makes use of a mobile phone (and the smart card) to provide a trusted computing base for Internet Banking. Since the modern mobile phone has bigger screen and enough computing power, the mobile phone approach does not necessarily require a PC to be involved. For both solutions, we have shown that the integrity of client's transactions can be ensured. Furthermore, if we assume that the hardware and the software of the mobile phone can be trusted, then some other security aspects in Internet Banking can be achieved with the mobile phone solution. These aspects includes the authentication of the client and the bank, the confidentiality and the integrity of the client's financial information (not only the transactions).

1 Introduction

Banking via Internet has gained its popularity lately. However, research [5, 3] has shown that this convenient service has some security problems. To start with, let's take a close look at how Internet Banking works in current solutions.

In a typical scenario of Internet Banking shown in figure 1, two parties, the bank and the client, communicate with each other using the *HTTPS* [4] protocol. There are mainly two steps in such communication, the authentication step and the information exchange afterwards. In the authentication step, both parties make sure that the one she is talking to is actually the one she wants to talk with. The bank proves its identify by providing its certificate issued by

a certificate authority, the client proves her identity by providing some secrets shared with the bank e.g. a password. After the authentication succeeded, the communication step takes place and all information exchanged between two parties is encrypted with the session key created in the authentication step, and message with the sensitive information also contains the digitally signed cryptographic hash of this information to ensure its integrity.

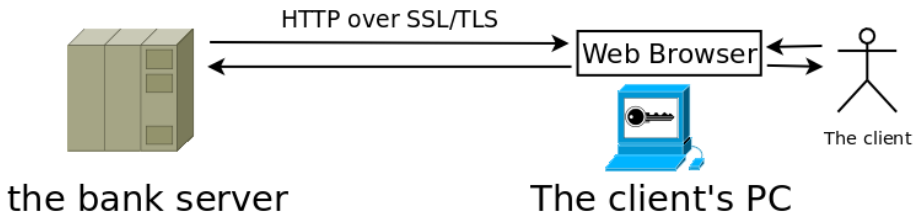

Figure 1: The Internet Banking

The Problem

In the earlier approaches, both mentioned steps are carried out by the web browser in the client side. The bank's certificate is verified by the web browser and the client's password is sent by web browser. In addition, all the information from the bank server is decrypted and displayed by the web browser. So the web browser such as *Internet Explorer* or *Firefox* should be trusted entity because it has full access and control to the financial information of client. However, it is a common knowledge that the PC and web browser may carry malicious code, such as a virus or spyware. For example, one can write a malicious Firefox plugin and trick the victim to install it, then this malicious person is able to eavesdrop or modify victim's sensitive information from the victim's Firefox. Experiments have shown that the so-called man-in-browser attack can be performed easily [5] since Firefox does not has a very strict policy for installation of its plugins.

Current Solutions of PoskBank and ABN-AMRO

Currently (as the paper is written), there are several solutions which address this problem by changing the information needed for authentication constantly to avoid the malicious person logining the bank server with the stolen password. For example, the PostBank uses SMS to distribute a list of passwords and requires the client to use different password every time when she authenticates herself. ABN-AMRO [1] provides an off-line smart card reader with the trusted keypad and display, and the client account card as a smart card carrying a symmetric cryptographic key protected by the PIN code [3]. As shown in figure 2, after the client claims her identity with her account number, the bank server sends a n-digit challenge and asks for a m-digit response. Then the client manually copies the challenge into the reader, the smart card computes the response

[1] While this paper is written, ABN-AMRO has issued their second version of Internet Banking solution with an on-line smart card reader connected to the PC via an USB connection. The configuration is similar to the proposed system but the detail of their system remains secret

by encrypting the challenge and the incremented on-card login counter using the on-card key, and returns it to the reader to display. The client then copies the response manually to the web browser and sends it to the bank server. In this way, no confidential data such as passwords or keys is revealed to the web browser.

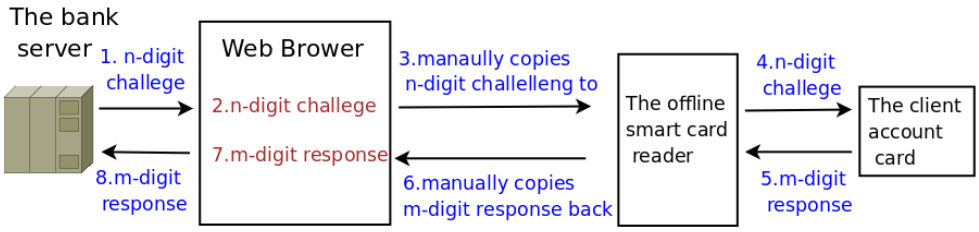

Figure 2: The offline challenge and response scheme

Nevertheless, both solutions mentioned above cannot prevent man-in-browser attacks. After successfully authenticated the client, the financial information can still be accessed and modified by malicious person once she gains the control of the browser since the information is kept in plain text by the browser. So the malicious attacker can steal money from the client by modifying the content of a transaction. The fundamental problem here is the sensitive information is handled by the untrusted system. Consequentially, we need an trusted system which can directly communicate with bank server and the client, and is capable of performing necessary cryptographic tasks. Therefore it must fulfill the following criteria:

1. have enough computing power to run cryptographic algorithms
2. equipped with some cryptographic keys
3. has its own trusted keypad and display to communicate with the client
4. has data channel to communicated with the bank server

Smart Card Based Solutions

Obviously, the smart card together with its reader is very suitable for building the mentioned trusted system since almost every customer possesses a smart card as her account card. Hiltgen et al. have proposed a smart card based solution [3]. In such solution, the trusted system consists of a smart card and a FINREAD (embedded financial transactional IC card reader) specified card reader [2]. The reader has its own cryptographic keys and enough compute power to perform cryptographic algorithms. In this paper, we will propose a different smart card based solution with a primitive smart card reader—the mini-reader. In contrast to the FINREAD specified reader, the mini-reader has no cryptographic keys. What it has are the trusted keypad and display. We will discuss the difference between two approaches in section 2.2.5 after we have introduced the mini-reader system.

Smart Card Based Solution with a Mobile Phone

In both mentioned smart card based solutions, because of the small display of the reader, the financial information are still displayed in the web browser. To address this problem, we will also briefly discuss the possibility of using a mobile phone to do Internet Banking. Because modern mobile phones have a bigger display and enough computing power, we can display the client financial information on the screen of mobile phone. If the mobile phone can be considered as a trusted computing base, then the confidentiality and integrity of these sensitive information can be ensured. The question if the mobile phone can be trusted is the most important problem in this approach and will be further discussed in section 2.4.

The Structure of the Paper

The rest of the paper is organized as follows. Section 2 gives the general architectures of the mini-reader solution and discusses the mobile phone approach. Section 3 analyzes potential threats and their counter measures. Section 4 develops the mini-reader solution in detail. Section 5 discusses some issues over our approach.

2 System Analysis and Design

As we have discussed in previous section, the fundamental problem of Internet Banking is that the financial tasks are performed on an untrusted system such as the PC and web browser. Clearly, the solution to this problem is to build a trusted system for such tasks where the security is very important. However, such job is rather difficult to be done in practice. There is a approach called *Trusted Computing*, in which a small piece of trusted computing device is built into the PC to monitor the whole system. But there are some issues about this approach such as the privacy of the custom because every PC is unique in this solution as argued by Ross Anderson in [1]. He further argues that such approach does not solve the problem because a clean OS or application itself may contain security vulnerabilities and this is the root cause of security problems. For Internet Banking, what we need is a totally trusted system which can process the financial information and communicate with the client. Of course, such a trusted system must also be able to perform some cryptographic algorithms to allow safe communication with the bank server.

With millions of bank clients, the cost of building such a system for each client is not affordable to the bank. The next option would be to protect the most valuable things in a not perfect system. Therefore, in this section we will first discuss what are the important security aspects we want to ensure and which of them we can ensure in reality. Then we will show the architectures of both solutions.

2.1 The Security Aspects of Internet Banking

There are mainly four security aspects which are important in Internet Banking. Firstly, both the bank and the client want to make sure that incoming messages originate from the legitimate one (authenticity). Secondly, nobody other than

the bank and the client should be able to access or modify the financial information transferred via Internet (confidentiality and integrity). Finally, if one party has performed an action i.e. a transaction, then there should be a way to prove this action is actually carried out by this party (non-repudiation).

Among these four aspects, the integrity of the transactions is the most important one. This is because the transactions are the only way to change the status of the account of the client, if the amount of money of a transaction or the target account number is modified by malicious person, the client will lose her money. If we can ensure that only the owner of the account can create the transactions and nobody else can modify the content of these transactions, then the malicious attacker cannot change the status of the client. Hence the client will not loose her money though the malicious person can still observe her financial information if the confidentiality is not ensured.

The next important aspect is the authenticity of the bank and the client because both the client and the bank needs to ensure they are talking to the person who claims she is. However, most of the existing Internet Banking solution uses SSL/TLS for server- side authenticating, faking a bank server requires the private key of the real bank server which is very hard to get. Whereas the shared secret for authenticating the client is much easier to steal (the password or the key protected by the PIN code), so it is much easier to impersonate the client than impersonate the bank server. Based on this observation, we believe the authenticity of the client is more important than of the bank.

The importance of the confidentiality is somewhat arguable, if the client is a company or an organization, then its information is valuable for its competitors. If it is a personal account, then the financial information is private sensitive. However, it is difficult to make a profit from the financial information of a personal account unless the owner of this account is a very important person or a millionaire. Here we assume the account is for average people because this assumption covers large portion of the clients and special client probably will use some other financial services. Therefore we conclude that the confidentiality is less important than t

In general, non-repudiation is very hard to achieve. Although each transaction is digitally signed by the client account card, there is a chance that the card and its PIN code are stolen. Based on these observations, we will focus on the integrity of the transactions and the authenticity of the client in the rest of this section.

2.2 The Mini-reader Approach

In this section, we will show the architecture of the mini-reader solution and briefly discuss the design of the protocols and its possible improvements. Note that we will present two variants of the mini-reader system, the basic version and the improved version. In the basic version, the authenticity of the client and the integrity of the transactions are ensured. In the improved version, the authenticity of the bank server and the client, the integrity of the transactions and all other financial information are ensured.

2.2.1 The architecture of the mini-reader approach

As depicted in the figure 3, The client uses a smart card as her account card, and all the necessary cryptographic work is shifting to this card. The mini-reader acts as an extension of the card, it connects the card to the PC via an USB connection. The trusted display and keypad are responsible for displaying the transaction messages and collecting the PIN code and the signing confirmation message from the client.

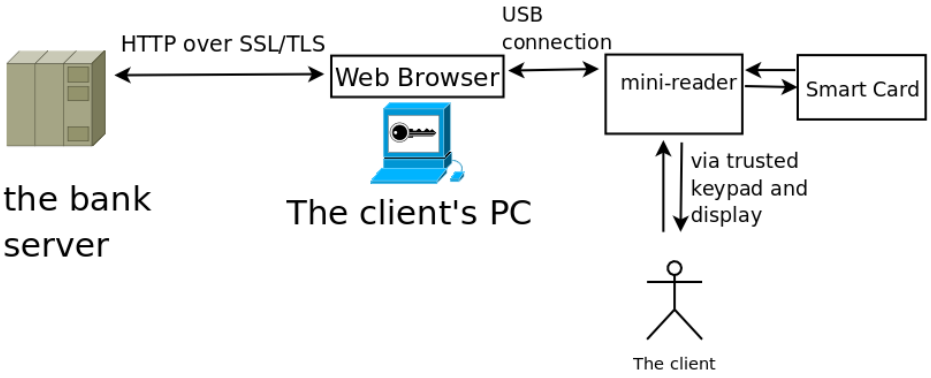

Figure 3: Architecture of mini-reader system

2.2.2 Key Management

Both bank server and the client smart card have their own public key-pair. The bank server has the SSL certificate of its public key signed by a well-known authority because this certificate will be checked by the web browser. The smart card has the certificate of its public key signed by the bank server's private key. In addition, the smart card also has the public key of the bank server for to check the signature of the messages coming from the bank.

2.2.3 The Basic Version of the Mini-reader System

In this basic version of the mini-reader solution, we focus on the authenticity of the client and the integrity of the transaction. So we still use the web browser to authenticate the bank server and handle the financial information except the transactions. The smart card and mini-reader are solely used for the authentication of the client and ensuring the integrity of the transaction. The Internet Banking process works as follows:

1. The authentication
   (a) The web browser authenticates the bank server using the standard SSL/TTL server-side authentication protocol
   (b) The web browser and bank server generates the session key.
   (c) The bank server authenticates the client by sending a challenge to the client's card

(d) The card generates the response using the PIN code protected private key on the card

2. Making a transaction

   (a) The client creates an new transaction in the web browser

   (b) The created transaction is displayed in the trusted display of the mini-reader

   (c) The client confirms the correctness of the transaction

   (d) The smart card computes the hash of the transaction and sign the hash with its private key

   (e) The bank confirms the smart card after successfully received the transaction

The solution makes use of challenge-response mechanism. After the web browser successfully authenticated the bank server using the standard SSL/TLS protocol, the bank server sends its challenge to the account card via the USB connection of the reader, the response is calculated by the PIN code protected cryptographic key on card and sent back to the bank server. Thus no passwords will be revealed to the web browser and the client does not need to manually copy the challenge and the response between the browser and the reader as the solutions discussed in section 1. After successful authentication, if the client makes a new transaction, it will be first viewed by the client via the trusted display of the mini-reader. If the client confirms the correctness of the transaction, its hash value will be digitally signed by the card's cryptographic key to ensure the integrity of the transaction.

Because the information of the client authentication is not kept in plain text in the web browser and the transactions are handled by the smart card, the secret for authenticating the client is safe and the integrity of the transaction is ensured in this solution. However, the authentication of the bank is done by the web browser, so we cannot ensure the authenticity of the bank server. Also the financial information such as the total amount of money and old transactions in the account are still kept in plain text by the web browser, thus the confidentiality can still be compromised. To achieved confidentiality of this much information, much more complex trusted hardware and software are needed, that is, to decrypted all financial information and displayed in a acceptable way, the reader needs much bigger display, very complex structure and more computing power. This increases the cost dramatically. In addition, complex system may introduce potential security risks than a simple one. Nonetheless, we can improve the design of protocol to ensure the authenticity of the bank server, and reduce the possibility that the integrity of the financial information being compromised without introduce new devices.

2.2.4 The Improved Version of the Mini-reader System

In this improved version, in addition to the authenticity of the client and the integrity of the transactions, we will also focus on the authenticity of the bank server and the integrity of the financial information (not only the transactions). The idea of improvement is to build a virtual channel between the smart card and the bank server. That is, in client side all messages are handled and stored

by the smart card in stead of the web browser. The web browser is only used to display the information on demand of the smart card and the integrity of the financial information displayed in the web browser is checked by randomly comparing some of them to the original ones stored in the smart card.

Both the card and the bank server have a public key, so the mutual public key authentication protocol can be used [4] to ensure the authenticity and a session key is created after the authentication. Then the bank server sends the all records of the financial information to the smart card encrypted using the session key. The smart card decrypts these records and stores them in the card. Next the smart card sends these records to the web browser in plain text to display them.

To ensure the integrity of the financial information displayed in the web browser. The smart card will randomly display some records of the financial information using the display of the mini-reader. The client can then check them on the web browser. In this way, if a malicious person modifies the information displayed in web browser, it can be detected.

The improved protocol can be briefly described as follows:

1. The authentication
   (a) The bank server and the client smart card perform the mutual public key authentication
   (b) The session key is created using the information in authentication
2. Random checking
   (a) The smart card decrypts the financial information sent by the bank server
   (b) The smart card stores and indexes the information
   (c) The smart card sends the information in plain text to the web browser to display it
   (d) The smart card randomly displays some records in the display of the mini-reader and ask the client to confirm them in the web browser.
   (e) The client can also randomly chose a record of the financial information shown in the web browser and check it with the original one stored in the smart card using the display of mini-reader.
3. Making a transaction is the same as the previous protocol

The indexing part of random checking protocol works like this. The whole financial information of the client consists of many records, each record represents one transaction that the client has made. When the smart card received these records, each record is assigned with an unique index number, for example, the oldest transaction with index *1* and the second oldest transaction with index *2* and so on. These index number will also be displayed in the web browser. So the index numbers of the records can be used as a reference to synchronize the record displayed in the display of the mini-reader and the one in the web browser. The drawbacks of this approach is the smart card has to process and store large amount of information, hence the performance might be poor due to the relatively lower processing speed of the card (compared to the PC).

2.2.5 Difference between the Mini-reader and FINREAD Approaches

Both systems have similar architecture as shown in the figure 3. Furthermore, the basic version of the mini-reader system and the FINREAD system focus on the same security aspects and have exactly the same functionality. They both ensure the authentication of the client by authenticating the client using the smart card and the integrity of transactions by digitally sign the hash value of the transactions using the private key of the smart card.

The difference is that the FINREAD specified reader has its own cryptographic key and it is capable of perform some cryptographic algorithms. So it is unable to be cloned and by specification [2] tamper-resistant. Whereas the mini-reader has no cryptographic key and cannot perform any cryptographic algorithms. Therefore the mini-reader can be replaced or tampered with. In practice, however, replacing or tampering with one's reader with a malicious one is difficult because it requires physical access.

Another advantage of having each client using a genuine reader is that it may prevent the misuse of the lost card. This is because in FINREAD system, each message from the client to the bank will contain the signature of the FINREAD reader and the bank server will refuse the messages without such a signature. So without a genuine reader the lost card cannot be used, and the misuse of the lost card with a FINREAD reader can be tracked down. But in this case the PIN code protects the card being misused and the smart card itself is tamper-resistant (under normal condition, tamper-evidence under all conditions) and unable to be cloned.

Based on above observations, employing a relatively expensive reader for the security attack which rarely happens would not be necessary in Internet Banking, because it increases the cost of the implementation of the solution. Considering millions of bank clients, if a solution costs much then the bank cannot afford to pay for the equipments needed for this solution for each of its customers. If the client have to pay for these expensive equipments, the client will likely not to purchase them and stay with the risks. This is the reason why the FINREAD system has never been implemented by any bank so far.

2.3 The mobile-phone approach

In this section, we will discuss the possibility of using a mobile phone (and the smart card) as a trusted computing base to perform secure Internet Banking. The reason of using a mobile phone is because its bigger display and computing power. Furthermore, nowadays almost every person has a mobile phone, if it can be used as the advanced smart card reader the cost of the implementation will be much less.

2.3.1 Mobile Phone as a trusted computing base

The central problem of this approach is if the mobile phone can be considered as a trusted computing base. This question is not easy to be answered. Currently the operating systems of the mobile phone (eg. symbian, android, windows mobile) are less complex than the ones for the PC. But with more and more features adding to them, eventually they will suffer from the similar security problems as Windows or Linux (or other modern OSes) has.

A mobile phone must have a SIM card to function. The SIM card is also a smart card, so essentially every mobile phone is a smart card reader.The SIM card can be used as a trusted computing module to certify the state of the operating system of the mobile phone. The idea is that the SIM card scans the current state of the registers, the BIOS and the memory and compute the hash value of this state. Then it checks the computed hash values against the one computed with a clean operating system. Therefore if the system is compromised, it can be detected. The detail of how the trusted computing module works fall out of the scope of this paper.

As we have already discussed in section 2, having a clean system installed does not mean the system is secure because the system itself may contains security problems. This is mostly the case when the complexity of the system has reached a certain level. Note that for Internet Banking we only need the trusted display, the keypad and the network modules (to communicate with the bank server) of the mobile phone. Therefore the mobile phone can be designed to have a secure mode with only these three necessary modules to perform tasks where the security is important, such as Internet Banking. Because to secure this small piece of software (in scale of tens of thousands lines of code) is much easier than secure the whole system. In Europe project ROBIN (Open Robust Infrastructure, robin.tudos.org), a trusted system with approximate 70K line of code is developed to perform security tasks for a mobile phone.

If a mobile has the security mode described above, and the integrity of the system loaded in such security mode can be ensured (fix the system in the hardware or use SIM card to certify it). Then we can consider such a mobile phone as a trusted computing base.

2.4 The architecture of the mobile phone approach

The architecture of the mobile phone approach is shown in Figure 4. In this approach, the mobile phone can connect to the bank server using its wireless connection or a wired connection (via PC). We do not specify the communication channel in the architecture, and since the financial information can be displayed in the mobile phone, the PC and web browser is not necessarily required. Therefore we ensure the confidentiality and the integrity of the financial information (includes the transactions) of the client. The key management is the same as the previous approach, and the mutual public key authentication protocol can be performed to ensure the authenticity of both side. Ideally, the client smart card can be integrated with the SIM card, but this is very difficult in practice since the banks and companies tends not to cooperate unless it is really necessary. So the mobile phone can be build as a wireless smart card reader and the client uses a wireless smart card, or build the mobile with an extra smart card slot.

3 Threats and Counter Measures

In this section, we will discuss the possible security threats of both systems and their counter measures. There are several possible attacks to our solutions, they are listed below:

- A malicious attacker may guessing the PIN code of the smart card

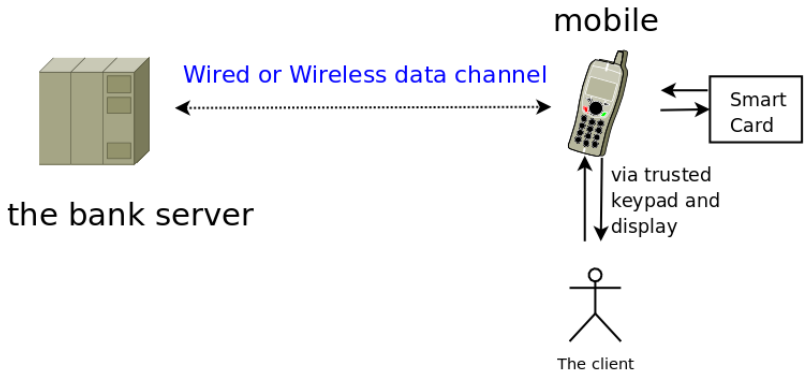

Figure 4: The architecture of mobile phone approach

- The mini-reader or the mobile phone (not the SIM card) may be replaced
- The mini-reader or the mobile phone (not the SIM card) may be tampered with

3.1 Guessing the PIN Code

In general, the smart card is designed to be tamper-evident, and the card will be blocked after several wrong PIN code attempts in a row. So it seems that the guessing of the PIN code will do no harm to our system because there are only limited number of chances. However, this functionality can be used to form a Denial Of Services attack, which is, to block the usage of the card by sending some on-line PIN guessing messages to the card. This is not a problem to our systems because the reader will interpret all the messages targeting the card as non-execution data, so the only PIN code verifying command will be generated by the reader itself by interaction with the user through the trusted keypad of the reader.

3.2 Replacing or Tampering with the Reader or the Mobile Phone

The mini-reader and the mobile phone are not tamper-resistant, thus they are not immune to tampering attacks. Here with the mobile phone we mean the phone without the SIM card. Moreover, they have no cryptographic key employed, it is possible to replace them with a fake one. However, replacing and tampering attacks cannot be perform on-line because it requires physical access. The person who does such attacks faces the high risk to be discovered because the small number of suspects. In addition, such attacks will always have to come together with a malicious web browser or compromised operating system. For example, if a fake mini-reader is displaying a wrong transaction, the message being displayed has to agree with what is showed in the web browser. In another scenario, if the fake mini-reader tricks the client to input her PIN, this PIN need to be collected by sending it over the Internet, again the malicious web browser or the compromised operating system plays its role here. It is

not so easy to write the specific code for a web browser to display the exact number as the fake mini-reader does, most of beginner crackers just download the malicious code from Internet and they are not able to modify the code by themselves. For the mobile phone, the comprised operating system should be detected by its trusted computing module. Furthermore, assume a malicious person has successfully launched a replacing or tampering attack and managed to get the PIN code of the client, she still need to steal the smart card to steal money from the client account.

Adding a key to the mini-reader would protect the system from the replacing attack. But then the mini-reader needs to have cryptographic abilities such as encryption and decryption. Also the bank has to maintain a database for the key of mini-readers. This is against our purpose of mini-reader's design, which aimed to build a low cost solution to Internet Banking.

As argued in previous paragraph, replacing or tampering attacks exist but may rarely happen. So there is a trade-off between the cost and the security, this is also the initial reason of developing the mini-reader system. For the special clients, such as millionaires, the bank may consider the FINREAD specified reader. After all, if a mixed use of both kind of readers would help, then why not deploy them.

4 Protocol Design

In this section, we will discuss three protocols of the mini-reader system and its improved version, they are:

- The client authentication protocol in the basic version of the mini-reader system
- The transaction protocol in both the basic and improved versions of the mini-reader system

Because the mutual public key authentication protocol is a standard protocol, the random checking protocol is explained in section 2.2.4, and the mobile-phone system uses the the same protocols as the ones used in the improved mini-reader system, we will not repeat them here. Note that in each protocol, each time when using the private key of the smart card the PIN code is required. In implementation the PIN code can be asked when the smart card is inserted into the mini-reader and the private key of the smart card can be used for a short time internal before the PIN code is required again. However, we will not specify any implementation detail here.

We will define some terms in the following protocols for the convenience reason:

- ID_{card} is the id number of the account card
- E_x is the x's public key
- D_x is the x's private key
- $\{message\}E_x$ means the message is encrypted by x's public key
- $\{message\}D_x$ means the message is digitally signed by x's private key

- h(message) is the cryptographic hash of message

Here x can be either the bank or the client account card.

The certificate of the client account card's public key is defined as follows:

$$\begin{aligned} Certificate_{card} = & ID_{card}, E_{card}, \\ & \{h(ID_{card}, E_{card})\}_{D_{bank}} \end{aligned}$$

4.1 The Client Authentication Protocol

This protocol refers to the client side authentication (the first version of the mini-reader system) discussed in section 2.2.3.

$$\begin{aligned} bank\_server &\rightarrow card : client\_authentication \\ card &\rightarrow bank\_server : Certificate_{card} \\ bank\_server &\rightarrow card : \{nonce\}E_{card} \\ card &\rightarrow client : input\_PIN\_code \\ client &\rightarrow card : PIN\_code \\ card &\rightarrow bank\_server : \{nonce\}E_{bank} \end{aligned}$$

In this protocol, the bank server first requires the certificate of the client account card's public key. Then it sends a random challenge number **nonce** encrypted by the public key of the card. When the card received this message, it decrypts the nonce with the pre-loaded public key of the bank and ask the PIN from the mini-reader's keypad, If the PIN code is valid, the smart card will encrypt the nonce with the public key of the bank and sends it back to the bank server. Note that every message between the bank server and the card is forwarded by the browser and the reader.

4.2 The Transaction Protocol

$$\begin{aligned} browser &\rightarrow card : transaction \\ card &\rightarrow client : input\_PIN\_code \\ client &\rightarrow card : PIN\_code \\ card &\rightarrow reader : transaction \\ reader &\rightarrow card : transaction\_displayed \\ card &\rightarrow client : shall\_I\_sign? \\ client &\rightarrow card : confirm\_signing \\ card \rightarrow bank\_server &: transaction, \\ & \quad \{h(transaction), nonce\}E_{bank} \\ bank\_server &\rightarrow card : \{nonce + 1\}E_{bank} \end{aligned}$$

When the new transaction is going to be signed, it has to be first displayed in the display unit of the mini-reader. If the displayed transaction message agrees with the transaction message displayed in the web browser, the client can be sure that she signs the right transaction. Thus, she presses the sign button in the keypad of mini-reader. After the card has received this confirmation message,

it signs the hash of the transaction and the nonce with its private key, note that here the nonce is the same nonce generated by the bank for authentication of the session. After the bank server successfully received the transaction, it confirms the card with the incremented nonce as shown in the last step of the protocol.

5 Discussion

The basic version of the mini-reader solution ensures the same security aspects as the FINREAD solution does, that is, the authenticity of the client and the integrity of transactions. In addition, the improved version of the mini-reader also ensures the authenticity of the bank server and the integrity of all other financial information. The cost of the mini-reader solution is lower than the FINREAD one because the mini-reader is much simpler than FINREAD specified one.

Nevertheless, the mini-reader system has its own drawbacks. Firstly, without any identification the mini-reader is vulnerable to replacing attacks, which means the original reader is replaced by a malicious one without being noticed. Secondly, the simple structure and functionality make the mini-reader much easier to be manipulated. Thirdly, with the improved version of the mini-reader system the performance can be poor because of the relatively slow processing speed of the smart card. Finally, it cannot ensure the confidentiality of the financial information of the client (includes the transactions) because this information is still kept in plain text in the web browser for display.

The currently used mobile phones cannot be directly used for Internet banking since they may not qualified as a trusted computing base. But together with the smart card, they provide an almost perfect solution for Internet Banking. However, since the SIM card is not developed for certifying the state of the OS, much effort may be needed to implement this feature. The security mode feature also requires quite some effort to implement, but this does not require the change of the infrastructure of the mobile phone design. Therefore it remains one practical solution for Internet Banking with low cost. Nonetheless, the cooperation of the banks and the mobile phone manufactures is the key factor of the success of such a solution. Furthermore, in case of the confidentiality of the financial information is not important, for better readability one can still display this information in the web browser with some random checking to ensure the integrity.

Acknowledgment

I would like to thank Erik Poll for supervising this paper and helping improving my academic writing skills.

References

[1] Ross Anderson. Trusted computing frequently asked questions. 2003. http://www.cl.can.ac.uk/ rja14/tcpa-faq.html.

[2] European committee of Standardizations. Embedded financial transactional ic card reader. *CEN Workshop 14174*, 2004.

http://www.cen.eu/CENORM/sectors/sectors/isss/
cen+workshop+agreements/finread.asp.

[3] Alain Hiltgen, Thorsten Kromp, and Thomas Weigold. Secure internet banking authentication. *IEEE security & privacy*, 2006.

[4] Andrew S. Tanenbaum. *Computer Networks, fourth edition*. Prentice Hall, 2003.

[5] Julian Verdurmen. Firefox extension security. *bachelor thesis, Radboud University Nijmegen*, 2007. http://www.cs.ru.nl/bachelorscripties/2007/.