Lightweight security for mobile commerce transactions

Kwok-Yan Lam\textsuperscript{a,*}, Siu-Leung Chung\textsuperscript{b}, Ming Gu\textsuperscript{a}, Jia-Guang Sun\textsuperscript{a}

\textsuperscript{a}School of Software, Tsinghua University, Beijing 100084, People’s Republic of China
\textsuperscript{b}School of Business and Administration, The Open University of Hong Kong, Hong Kong, People’s Republic of China

Received 15 November 2002; revised 3 June 2003; accepted 3 June 2003

Abstract

This paper describes a lightweight security mechanism for protecting electronic transactions conducted over the mobile platform. In a typical mobile computing environment, one or more of the transacting parties are based on some wireless handheld devices. Electronic transactions conducted over the mobile platform are gaining popularity and it is widely accepted that mobile computing is a natural extension of the wired Internet computing world. However, security over the mobile platform is more critical due to the open nature of wireless networks. Furthermore, security is more difficult to implement on the mobile platform because of the resource limitation of mobile handheld devices. Therefore, security mechanisms for protecting traditional computer communications need to be revisited so as to ensure that electronic transactions involving mobile devices can be secured and implemented in an effective manner. This research is part of our effort in designing security infrastructure for electronic commerce systems, which extend from the wired to the wireless Internet. A lightweight mechanism was designed to meet the security needs in face of the resource constraints. The proposed mechanism is proven to be practical in real deployment environment.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Electronic commerce; System security; Cryptography; Mobile computing

1. Introduction

Electronic transactions conducted over the mobile platform are gaining popularity due to the convenience and portability of low-cost handheld devices. In order to promote the adoption of electronic commerce, system designers have been actively exploring approaches that may attain higher penetration than the wired Internet. With the explosive growth of mobile phone population and the fast adoption of wireless network technology, support for electronic commerce transactions over the mobile platform has become a realistic and attractive option. Besides, the cost and performance of handheld devices with wireless capability have also improved tremendously in recent years. Indeed, it is widely accepted that mobile computing is a natural extension of the wired Internet computing world [1].

In a typical mobile computing environment, one or more of the transacting parties are based on some wireless computing devices. At the physical layer, wireless links have been commonly used for many years. Such wireless paths, however, are mainly adopted as part of the network backbone and are invisible to the end users; thus do not support the deployment of wireless end user devices. With contemporary mobile computing, wireless links covering wide geographical areas are made available to end users, and providing end users with network access points in a truly mobile manner [2].

However, security over the mobile platform is more critical due to the open nature of wireless networks. A wireless link which forms part of the network backbone can be protected quite easily by adopting link level encryption because of the point-to-point nature of the traffic pattern. Whereas wireless links accessible to end user computing devices are more vulnerable to security attacks because of the openness of the wireless access points. It is much easier for malicious users to gain access to the wireless network and perform fraudulent activities such as eavesdropping and impersonation.

Furthermore, security is more difficult to implement on a mobile platform because of the resource limitation of mobile handheld devices. To realize the benefit of mobile
commerce, end users expect to have network access devices that are easy and convenient to carry. Thus handheld devices such as palm top or pocket PC are the most commonly embraced platforms as end user devices for mobile commerce. Nevertheless, computing operations on handheld devices are subject to resource constraints such as small memory, slower CPU, limited power supply and smaller display. These constraints pose serious limitations to security mechanisms suitable for mobile commerce.

Therefore, it is important to have some lightweight security mechanisms suitable for mobile handheld devices and yet secure enough to protect transactions carried over the wireless network. In face of the resource constraints of handheld devices, security mechanisms for protecting traditional computer communications need to be revisited so as to ensure that electronic transactions involving mobile devices can be secured and implemented in an effective manner. For example, computation-intensive operations such as public key cryptographic systems implemented for traditional computer platforms, however secure they are, may not be the most ideal means for securing mobile commerce transactions.

This paper describes a lightweight security mechanism for protecting electronic transactions conducted over the mobile platform. This research is part of our effort in designing security infrastructure for electronic commerce systems, which extend from the wired to the wireless Internet. A lightweight security mechanism was designed to meet the security needs in face of the resource constraints. The proposed mechanism is proven to be practical through testing in real deployment environment.

The organization of this paper is as follows. In Section 2, the system model for mobile commerce is described. This section aims to define the scope of our studies by discussing the key characteristics of the mobile platform that are of interest to mobile commerce. Section 3 presents existing security mechanisms for protecting mobile transactions and points out the shortfalls of such schemes. A lightweight security mechanism suitable for mobile transactions is also presented in Section 3. Section 4 describes a system architecture within which the proposed security mechanism can be adopted. Section 5 concludes the discussion of this paper.

2. The mobile commerce model

This section describes a system model based on which mobile commerce systems can be designed and developed. The mobile commerce model presented in this discussion was motivated by practical considerations and with the objective of encouraging adoption of electronic commerce through enhancement in convenience and security from the perspective of the end users. In this model, the major criteria for being mobile include the use of wireless network access points by end users and the adoption of handheld computing devices as a client platform to execute electronic transactions.

In this model, mobile commerce refers to electronic commerce systems which allow end users to initiate and participate in electronic transactions over wireless channels. As mentioned, the support for wireless network access point is believed to be a major factor to enhance the convenience, and hence pervasiveness, of electronic commerce. At present, there are two common ways for implementing such wireless channels—the mobile phone network and wireless LAN [3]. Nevertheless, considering the typical profile of target mobile commerce users, mobile phone channel is believed to have a much higher penetration compared to wireless LAN [4]. In this discussion, we mainly consider mobile commerce transactions initiated by client devices over the mobile phone network. A simplified overview of the network architecture is as shown in Fig. 1.

In this system architecture, there are three entities namely the end user, the mobile network and the mobile commerce provider (or m-commerce provider in short). For illustration, we consider a mobile stock trading service such that the m-commerce provider is a securities company (i.e. a stock trading company) and the end user is the investor (i.e. a customer of the stock trading company). The end user connects to the m-commerce provider through a mobile phone network. Mobile network operators supporting m-commerce provides Internet connectivity through an IP gateway. The m-commerce provider connects to the mobile operator, and in turn the end user, through the Internet. Thus the m-commerce provider does not need to handle the wireless connectivity. Instead, its main focus is on the application architecture that caters for the characteristics of handheld devices.
Besides the underlying physical network links, another major factor that defines an electronic commerce systems as being mobile is the support of handheld devices as a transaction platform for end users. In order to attain high penetration of mobile commerce, convenience to end users is of prime concern. In this connection, handheld computing devices such as pocket PC and palm top (typically referred to as personal data assistant or PDA) are attractive options because of the following reasons [5]:

- low-cost,
- low battery consumption,
- highly portable,
- affordable wireless capability,
- reasonable computing and display capability for simple transactions,
- instantaneous power up (no lengthy boot up latency).

Regardless of the attractive features of handheld devices, there are non-trivial challenges to be addressed in order for handheld computing devices to be adopted as a practical platform for electronic commerce transactions. While handheld devices are typically implemented to support web browsing functionality, the limited display capability of handheld devices compared to ordinary PCs requires that web pages developed for traditional e-commerce applications be modified in order to cater for the display constraints of handheld devices. Furthermore, the limited computing power of handheld devices also requires that computation-intensive operations be avoided in mobile commerce applications.

In particular, security mechanisms involving extensive use of public key cryptographic functions are CPU-demanding, thus need to be studied carefully when deployed in mobile applications. For example, the Secure Electronic Transaction (SET) protocol developed by the payment card industry is too heavy weight for mobile platforms [6].

Thus the concept of wireless protocol gateway, which serves as a fixed-line agent for the handheld devices, was introduced. With the wireless protocol gateway, handheld devices are connected to the application server indirectly, i.e. through the gateway server. Typically, the end user is authenticated to the gateway server through some simple authentication scheme such as password login, and the gateway server in turn executes complex transaction protocols on behalf of the handheld devices.

The connection between the handheld device and the gateway server is typically protected by ready security mechanisms such as the Transport Layer Security (TLS) [7] or Wireless Transport Layer Security (WTLS) [8]. For example, TLS (which is commonly known as the Secure Socket Layer or SSL) is readily available as a built-in function of most browsers including those available on pocket PC, and the built-in SSL typically provides an encrypted channel that ends at the web server of the e-commerce provider. However, experiences acquired from actual hacking incidents highlighted the fact that security approaches based on SSL at a web server are hardly sufficient [9]. Note that WTLS is a wireless variant of TLS.

Consequently, the concept of end-to-end security is increasingly important as a protection mechanism for e-commerce transactions [10, 11]. In this context, end-to-end security means that transactions protected by encryption are decrypted only at the execution point, thus are not exposed even at the web server. Therefore, in our mobile commerce model, mobile transactions are implemented by a combination of the wireless protocol gateway and end-to-end security mechanism.

In this paper, we adopt this mobile application model as the basic framework for implementing security of mobile commerce transactions. To summarize, the following system architecture is adopted (Fig. 2).

The objective of our work is to design and implement a lightweight security mechanism for achieving end-to-end security between the handheld device and the gateway server. Cryptographic keys and functions at the gateway server may be protected by tamper-resistant hardware devices if necessary. The design of tamper-resistant devices is outside of the scope of this paper though. For readers who are interested in this topic, they can refer to Ref. [12].
The lightweight security mechanism presented in this paper has been implemented and in use successfully by some e-commerce and e-government applications.

3. Lightweight security mechanism

Due to resource constraints of mobile computing platforms, lightweight security mechanisms are needed for protecting mobile commerce transactions. For example, the WTLS protocol was developed by the WAP Forum [8] as a variant of the TLS protocol to cater for the resource limitations of wireless devices [13]. In this setting, a protocol gateway called WAP Gateway was developed to provide connectivity between a WAP-based handheld device and the application web server.

In its simplest form, wireless handheld devices such as mobile phones may adopt the WAP technology and secure electronic commerce transactions using WTLS. This approach, however, does not meet the end-to-end security requirements that are shown in Fig. 2. With the use of WTLS, the security architecture is as shown in Fig. 3.

In Fig. 3, we intentionally position the WAP gateway as a component outside of the m-commerce provider. Technically, the WAP gateway may be hosted by the mobile operator or the m-commerce provider. In reality, due to cost consideration of m-commerce providers, the WAP gateway is typically hosted by the mobile operator.

The architecture shown in Fig. 3 leads to two major security concerns. First, the WAP Gateway, which translates WTLS-protected traffic to TLS-protected traffic, creates a security gap at the WAP gateway. As a protocol gateway, the WAP gateway receives WTLS traffic from mobile handheld devices, decrypts the WTLS traffic and re-encrypts them by sending the data to the TLS traffic. Therefore, mobile transaction data is exposed at the WAP gateway. The situation is worsened by the fact that the WAP gateway is hosted outside of the perimeter of the e-commerce provider. Future version of the WAP protocol suite is believed to provide TLS at handheld devices though. Secondly, the protected traffic ends at the web server and thereafter transaction data exist in clear form, thus does not satisfy the end-to-end security requirement even with provision of TLS-protection of data starting from the handheld device in future WAP version. This scenario is especially undesirable if user passwords are used for identification. The existence of the security gap will undermine the security of the e-commerce system.

Therefore, it is advisable to provide additional security mechanisms on top of the TLS and WTLS channels. In the context of WAP platform, additional security processing may be implemented in WMLScript [14] which is interpreted by the WAP browser supported by WAP-based handheld devices. However, the WMLScript approach is limited by the expressive power and computation speed of WMLScript interpretation. To implement cryptographic protocols suitable for open e-commerce transactions, computation-intensive cryptographic functions are usually needed. Though WMLScript has been demonstrated to be capable of supporting implementation of some cryptographic algorithms, the choices of algorithms are very limited and the constraints are especially obvious for algorithms that involve plenty of arithmetic operations.

Considering the security issues encountered by mobile commerce applications with or without WAP, we designed and developed a lightweight security mechanism for securing mobile commerce transactions in an efficient and effective manner. The lightweight mechanism is placed on top of the TLS to provide end-to-end security between the mobile handheld devices and the wireless protocol gateway. The mechanism assumes that the mobile handheld device supports plug-in or applet implementations in an Internet browser environment. Nowadays, handheld devices supporting these features are gaining popularity and there is a rapid growth in mobile commerce applications running on such platforms such as electronic banking, electronic stock trading and electronic government services [15].

The lightweight security mechanism suitable for deployment in the mobile commerce model specified in Section 2 need to satisfy the following requirements:

Fig. 3. WAP-based system architecture of m-commerce applications.
The proposed mechanism meets the above requirements through an authentication protocol which, upon successful completion of the protocol run, establishes a symmetric secret key shared only by the mobile handheld device and the wireless protocol gateway. The handheld device thereafter communicates securely with the wireless protocol gateway, which in turn executes complex transaction protocols with the application server efficiently. The lightweight mechanism makes use of a combination of public key cryptographic mechanism and simple password authentication. The mechanism may be complemented by some tamper-resistant hardware device to achieve a restricted form of non-repudiation typical in financial transactions. The discussion on tamper-resistant hardware is outside of the scope of this paper though. The protocol was implemented as an ActiveX module running in an Internet browser on a Windows CE system of a Pocket PC.

Security mechanisms developed for the open system environment are almost invariably based on public key cryptography because of the complex key management problem in such an environment. Nevertheless, security mechanisms involving public key cryptographic functions are CPU-demanding, thus need to be studied carefully when deployed in mobile applications. Furthermore, the storage of private keys also poses another security problem that impact on the practical use of handheld devices. In general, it is considered not cost-economical to deploy public key cryptography in mobile devices. Albeit the availability of public key enabled smart cards may help alleviate the problem, the adoption of extra hardware tokens translates to additional cost and administrative overhead.

Nevertheless, public key cryptography is still believed to be the most suitable mechanism for securing e-commerce transactions due to the complex key management problem in the open network environment [16]. From a practical point of view, mobile commerce transactions tend to avoid using public key cryptography whenever possible. In reality, due to practical consideration, e.g. key management, public key cryptography should be adopted as a basis of e-commerce security whenever possible.

To resolve these conflicting requirements, we separate the consideration of encryption operation and decryption operation. In the context of public key cryptography, the encryption operation is controlled by a publicly known cryptographic key. Whereas, the decryption operation is controlled by another key, the private key, known only to the intended recipient of the cryptogram. For illustration, we consider the RSA system which is the most popular public key cryptosystem to date. In practical deployment of the RSA cryptosystem, it is a standard practice that the length of the encryption key is fixed at a size that is much smaller than that of the decryption key [17]. As a consequence, the encryption operation is less computation-intensive compared to the decryption operation. Furthermore, the public key used in the encryption operation does not require stringent security protection.

The lightweight security mechanism presented in this paper aims to address the computation constraints by exploiting the performance difference between encryption and decryption of public key cryptosystems. In the description of the security protocol, the following denotations are used:

- $S$ denotes the server side of the security protocol;
- $C$ denotes the client side of the security protocol;
- $E_{K_S}$ denotes the public key of $S$. We assume that the public key of $S$ is made known to $C$ prior to the execution of the protocol;
- $D_{K_S}$ denotes the private key of $S$;
- $PIN$ is the password of user $C$ which is known only to $C$ and is verifiable by $S$. The password typically has a length of eight to sixteen printable characters;
- $R_n$ is a nonce identifier (a random number) generated by $S$;
- $R_{b1}$ and $R_{b2}$ are nonce identifiers (random numbers) generated by $C$;
- $SN$ is a sequence number generated by $S$ for this protocol run;
- $SK$ is the symmetric session key shared by $S$ and $C$ for protecting transaction messages in the session;
- $E_{K_S}[X]$ denotes the encryption of data $X$ under the control of public key of $S$;
- $E_{SK}[X]$ denotes the encryption of data $X$ under the control of session key $SK$.

In this discussion, 1024 bit RSA is used as the public key algorithm used by $S$. $SK$ is a 128 bit binary value for some symmetric algorithms such as AES or 3DES. All nonce identifiers are 128 bit binary values.

A mobile user wishing to conduct an e-commerce transaction with the m-commerce provider initiates the security mechanism from his/her mobile handheld device. The mechanism is implemented as an authentication protocol [18] between the handheld device and the wireless protocol gateway. The mechanism starts with the client $C$ sending an authentication request to the server $S$. In response to the request, the following protocol messages are exchanged between $S$ and $C$. Upon successful completion of the protocol run, the two parties share a secret key which may be used to protect subsequent transaction messages in the same session.
Lightweight authentication protocol $P$:

M1. $S \rightarrow C$: $R_a$

M2. $C \rightarrow S$: $E_{K_S}[R_a, R_{b1}, PIN]$, $E_{K_S}[R_a, R_{b2}, SK]$

M3. $S \rightarrow C$: $E_{SK}[SN, R_{b2}]$

An informal description of the protocol $P$ is as follows. As a lightweight scheme, the protocol mechanism is very simple and the logic of the protocol is straightforward. Thus an informal treatment is more effective in describing the protocol. Analysis of the mechanism using formal logic is outside of the scope of this paper.

$M1$: The server $S$ starts the authentication by sending a random challenge $R_a$ to $C$. The purpose of $R_a$ is to allow the server to assure the freshness of messages involved in this protocol run [19,20].

$M2$: The client $C$ returns a response to $S$ which consists of two message tokens: the password authentication token and the key exchange token. The password authentication token contains the secret password $PIN$ which is known only to $C$. The password allows $S$ to authenticate $C$ by verifying the correctness of $PIN$. The secrecy of $PIN$ is protected by encrypting $PIN$ with $E_{K_S}$. When verifying the correctness of the password, $S$ checks the value of $R_a$ so as to ensure that $M2$ is not a replay of some previous protocol message. The nonce identifier $R_{b1}$ introduces additional randomness to the response and is very important to the security of the system. Without $R_{b1}$, the secret $PIN$ can be easily deduced by brute force attack on $M2$ in that $R_a$ in $M1$ was sent in clear form. In actual implementation in financial institutions, this verification operation is typically conducted within a tamper-resistant hardware environment so as to ensure that $PIN$ is not disclosed to anyone including $S$. The key exchange token in $M2$ is a structure similar to the authentication token. In this token, $SK$ is the session key generated by $C$ and is available only to $S$ because the decryption key $DK_S$ is known only to $S$. The nonce identifier $R_{b2}$ is included in the key exchange token to allow $C$ to confirm that $S$ does know $SK$.

$M3$: The server $S$ proves to $C$ that it does know the shared key $SK$ by encrypting the nonce identifier with $SK$. At this point, $C$ is satisfied that the entity that knows $SK$ is $S$ and it continues to communicate with $S$ in a secure manner using the cryptographic key $SK$.

Therefore, the mechanism $P$ has the following characteristics that allow it to satisfy the abovementioned security requirements of mobile commerce:

- $P$ provides security by achieving authentication and key exchange;
- $C$ authenticates to $S$ based on a simple password mechanism, secured by public key cryptographic operations, thus does not require additional cryptographic hardware tokens such as smart cards;
- $S$ authenticates to $C$ by proving knowledge of private key of $S$;
- at the end of the protocol, $S$ and $C$ share a session key $SK$ such that $SK$ is chosen by the entity that knows the $PIN$ of $C$ and is readable by the entity that knows the private key of $S$. That means, only the genuine $C$ and $S$ knows $SK$;
- the protocol can be implemented efficiently and easily without using additional hardware devices, e.g. smart cards;
- $C$ does not perform public key decryption nor digital signature operation thus incurring low computation overheads;
- $C$ does not use any long term secret keys thus does not require secure storage for keeping secret cryptographic keys.

To summarize, the proposed scheme $P$ is a lightweight security mechanism that meets the security requirements of mobile commerce and is suitable for implementation on resource-scarce devices. This scheme was designed to meet the security needs of security-sensitive mobile commerce applications including electronic financial services and electronic government services initiated over mobile handheld devices.

4. Security architecture for mobile commerce

The value and significance of the lightweight security mechanism $P$ will become obvious when we consider its application in a real mobile commerce systems. We consider the architecture of a typical m-commerce application system as an illustrative case. In this case, we adopt the mobile commerce model presented in Section 2 as the underlying design framework. For example, the m-commerce service provider may be a financial institution that provides stock trading facilities for retail investors. Thus, end users of the service are retail investors connecting to the m-commerce system through a mobile network and from some handheld mobile devices. Thus an overview of the application topology is as shown in Fig. 4.

Note that, apart from constraints such as network bandwidth and end user devices, the underlying network architecture is transparent to the application system. Thus typically m-commerce application systems do not normally address network issues specific to mobile communications. Instead, m-commerce applications usually take into consideration the bandwidth, display and computational constraints of mobile handheld devices when designing transactions with and presentation at the end user device.

In the mobile commerce system, the wireless protocol gateway needs to manage login accounts (and passwords) of the end users, authenticates them and executes transactions on behalf of them once they are successfully
authenticated. As elaborated in Section 2, end user devices do not connect directly to the application server. Instead, they establish a trusted connection with the wireless protocol gateway, which in turn executes complex transactions with the application server on behalf of the end users. Thus, the wireless protocol gateway needs to implement a password management scheme and performs the role of $S$ of protocol $P$.

Therefore, in addition to the security mechanism $P$, the password management mechanism at the wireless protocol gateway also plays an important role to the security of the m-commerce system. In a real deployment scenario especially for sensitive applications such as financial and government applications, two key requirements are identified as a balance between computational overhead and security protection.

- User passwords stored at the wireless protocol gateway must be in encrypted form (or must be transformed by some cryptographic one-way function $owf$).
- Verification of password must be performed in a secure environment.

The first requirement regarding storage of password has been commonly used in most operating systems such as Unix. The main objective of this requirement is to prevent anyone including system administrators from viewing the values of user passwords. In reality, however, this requirement alone is insufficient in ensuring the security of the password in that correct passwords entered by genuine users are still exposed to the system. In this connection, the mechanism $P$ together with the second requirement aim to address this security issue. The mechanism $P$ ensures that end user password are encrypted and is readable only to $S$.

In order to achieve stronger security protection, the decryption key $DK_S$ is strictly controlled and stored in a secure environment. The decryption operation, and hence the password verification operation, is conducted within a secure environment according to the second requirement. This way, the security architecture prevents this sensitive information from being accessed by any party within the e-commerce provider’s system. In practice, tamper-resistant hardware devices are adopted to implement the secure environment required by the m-commerce system. In fact, tamper-resistant hardware devices have been used extensively in financial and government systems as a major means for management of cryptographic keys.

Fig. 5 gives an overview of the implementation of a secure environment at the wireless protocol gateway.
In summary, the mobile commerce security architecture presented in this section provides a secure means for authenticating end users and transacting with them. The security architecture provides a wireless protocol gateway to execute complex transaction protocols with the application server and at the same time allows mobile handheld devices to securely connect to the gateway using lightweight mechanisms. The combination of the mechanism $P$ with the secure password management at the wireless protocol gateway serves as an effective balance between computational overhead and security protection.

5. Conclusion

This paper describes a lightweight security mechanism for protecting electronic transactions conducted over the mobile platform. In a typical mobile computing environment, one or more of the transacting parties are based on some wireless handheld devices. However, security over the mobile platform is more critical due to the open nature of wireless networks. Furthermore, security is more difficult to implement on a mobile platform because of the resource limitation of mobile handheld devices. Therefore, it is important to ensure that security mechanisms for electronic transactions involving mobile devices can be implemented in an effective manner.

Lightweight security mechanisms suitable for mobile handheld devices and yet secure enough to protect transactions carried over the wireless network play a critical role in development of mobile commerce systems. In face of the resource constraints of handheld devices, security mechanisms for protecting traditional computer communications need to be revisited so as to ensure that electronic transactions involving mobile devices can be secured and implemented in a practical manner.

Considering the security needs of e-commerce transactions and the resource constraints of mobile devices, security mechanisms designed for mobile commerce transactions need to meet the following requirements:

- secure,
- easy to implement,
- low computation needs,
- no storage of secret cryptographic keys,
- achieve entity authentication,
- achieve key exchange.

In this paper, a lightweight security mechanism $P$ was presented to meet these requirements of mobile commerce applications. The proposed mechanism has the following characteristics that allow it to satisfy the abovementioned security requirements:

- $P$ provides security by achieving authentication and key exchange;
- $C$ authenticates to $S$ based on a simple password mechanism, secured by public key cryptographic operations, thus does not require additional cryptographic hardware tokens such as smart cards;
- at the end of the protocol, $S$ and $C$ share a session key $SK$ such that only the genuine $C$ and $S$ knows $SK$;
- the protocol can be implemented efficiently and easily without using additional hardware devices, e.g. smart cards;
- $C$ does not perform public key decryption nor digital signature operation thus incurring low computation overheads;
- $C$ does not use any long term secret keys thus does not require secure storage for keeping secret cryptographic keys.

The lightweight mechanism is further studied in the context of a mobile commerce security architecture. In this architecture, the role of the security mechanism and its interactions with other system components are elaborated. The mechanism is deployed together with a secure password management scheme typical in security-sensitive application environments such as financial and government applications. In short, the mobile commerce security architecture presented in this section provides a secure means for authenticating end users and transacting with them. The security architecture provides a wireless protocol gateway to execute complex transaction protocols with the application server and at the same time allows mobile handheld devices to securely connect to the gateway using lightweight mechanisms. The combination of the mechanism $P$ with the secure password management at the gateway serves as an effective balance between computational overhead and security protection.

We conclude that the proposed scheme $P$ is a lightweight security mechanism that meets the security requirements of mobile commerce and is suitable for implementation on resource-scarce devices. This scheme was designed to meet the security needs of security-sensitive mobile commerce applications including electronic financial services and electronic government services initiated over mobile handheld devices. This research is part of our effort in designing security infrastructure for electronic commerce systems, which extend from the wired to the wireless Internet. The proposed mechanism is proven to be practical in real deployment environment.

Acknowledgements

This research was funded by the National 863 Plan (Project Numbers 2001AA414220, 2001BA205A06 and 2002AA412010), People’s Republic of China. Part of this work was supported by the Electronic Banking project of PrivyLink International Limited, Singapore.
References


Kwok-Yan Lam is a Professor at the School of Software, Tsinghua University, People’s Republic of China since 2002. Prior to joining the Tsinghua University, Professor Lam has been a faculty member of the National University of Singapore and the University of London since 1990. His main research interests include distributed systems, information security and tamper-resistant design. Professor Lam was a visiting scientist at the Isaac Newton Institute of the Cambridge University and a visiting professor at the European Institute for Systems Security. He has been a chief security architect for a number of electronic banking and electronic government systems in Singapore and Hong Kong. Professor Lam received his B.Sc. (First Class Honours) from the University of London in 1987 and his PhD from the University of Cambridge in 1990.

Siu-Leung Chung is an Associate Professor at the School of Business and Administration, The Open University of Hong Kong. Prior to joining The Open University, he has been a faculty member of the National University of Singapore and the University of Toledo, Ohio, USA since 1991. His main research interests are information and system security, e-commerce security and software project management modeling. Dr Chung received his B.Sc. (Engineering) from the University of Hong Kong in 1977 and his PhD in Computer Science from the University of Illinois in 1991.

Ming Gu is an Associate Professor at the School of Software, Tsinghua University, People’s Republic of China. Her research interests include software system theory and middleware architecture.

Jia-Guang Sun is a Professor at the School of Software, Tsinghua University, People’s Republic of China. Professor Sun, widely regarded as a leader in CAD research in China, is a Fellow of the Chinese Academic of Engineering. Professor Sun is the Director of the National Engineering Research Center for CAD Supporting Software and is concurrently the Executive Vice Dean of the School of Software of Tsinghua University.