Analysis of Payment Transaction Security in Mobile Commerce

Seema Nambiar, Chang-Tien Lu
Department of Computer Science
Virginia Polytechnic Institute and State University
7054 Haycock Road, Falls Church, VA 22043
{snambiar, ctlu}@vt.edu

Lily R. Liang
Dept. of Computer Science
University of the District of Columbia
Washington, DC 20008
lliang@udc.edu

Abstract

Mobile payment is the process of two parties exchanging financial value using a mobile device in return for goods or services. This paper is an analysis of the security issues in mobile payment for m-commerce. We introduce m-commerce and mobile payment, discuss the public key infrastructure as a basis for secure mobile technologies, and study the features for different security technologies employed in current m-commerce market, including WAP, SIM application toolkit and J2M. In addition, we compare the effectiveness of these security technologies in supporting a secure mobile payment, and discuss research issues to enhance the security of mobile payment for large scale deployment of m-commerce.

1. Introduction

A mobile device is a wireless communication tool, including mobile phones, PDAs, wireless tablets, and mobile computers. Mobile commerce (M-commerce) can be defined as any electronic transaction or information interaction conducted using a mobile device and mobile networks, which leads to transfer of real or perceived value in exchange for information, services, or goods. M-commerce offers consumers convenience and flexibility of mobile services anytime and at any place, and is playing an increasingly important role in payments and banking. The global m-commerce market is expected to be worth a staggering US$200 billion by 2004[1].

M-commerce applications differ from E-commerce applications in the following factors: limitations of the mobile devices, portability of mobile devices, allowance of pervasive computing, capability of location awareness, and portability of merchant machines[2]. The security challenges in mobile commerce are related to but not limited to the mobile devices, the radio interface, the network operator infrastructure and the type of mobile commerce application[3].

Mobile payment, a major component of M-commerce, is defined as the process of two parties exchanging financial value using a mobile device in return for goods or services. Security is an essential consideration for mobile payment which can be challenged during sensitive payment information handling or transmission. Comparing to E-commerce, mobile payment has particular security and privacy challenges due to the differences between their underlying technologies. The major difference is that the transport of payment involves wireless service provider. The ability to address the issue is a major factor affecting the customer confidence, market penetration, and long-term success of M-commerce applications.

Four properties have always been essential for secure transaction, including authentication, confidentiality, integrity, and non-repudiation[4]. ***Authentication*** is concerned about verifying the identities of parties in a communication and confirming that they are who they claim to be. ***Confidentiality*** is about ensuring that only the sender and intended recipient of a message can read its content. ***Integrity*** is concerned about ensuring the content of the messages and transactions not being altered, whether accidentally or maliciously. ***Non-repudiation*** is about providing mechanisms to guarantee that a party involved in a transaction cannot falsely claim later that she did not participate in that transaction

In this paper, we discuss the background on public key infrastructure as a basis for security in different mobile technologies, and study the security measures in WAP, SIM application toolkit, and J2ME, which are the mobile security technologies employed in the current M-commerce market. Effectiveness of WAP, SIM application toolkit and J2ME in supporting a secure M-commerce payment application are analyzed and compared. Several research issues in M-commerce payment security are also discussed.

2. Public key infrastructure

Public key infrastructure (PKI) is a system of digital certificates, certification authorities, and other registration authorities that provides solutions to enable a secure mobile commerce. The theory of PKI is presented as follows.

Public Key Cryptography: Public key infrastructures are based on public key cryptography, which uses two keys: a private key that is kept a secret, and a public key that can be divulged publicly. An interesting property of this pair of keys is that to decrypt messages encrypted with one, the other is needed. The keys are said to be asymmetric. The most popular algorithm for public key cryptography is RSA. Elliptic curve cryptography algorithms are starting to gain acceptance into mobile devices. They rely on different mathematical properties that allow for shorter keys, which enable faster computations, lower power consumption, less memory and bandwidth requirements, and hence are quite appealing for mobile devices.

Digital Signatures: Digital signatures can ensure the authenticity of transaction parties, integrity, and non-repudiation of transmissions. A digital signature is created when the document to be transmitted is enciphered using a private key. The process of enciphering the document using the private key authenticates the document, since the document could only have been enciphered using the private key of the owner. A digitally signed document or message is unalterable after the signature. The recipients can verify the signature by deciphering using the public key. In real world, documents are not completely encrypted to save time. In such cases one-way hash functions are used. A hash uses a one-way mathematical function to transform data into fixed length digest called a hash, which is subsequently enciphered. The verification of the signature involves reproducing the hash generated from the received message and comparing it with the deciphered original hash[5].

Digital Certificates: Digital signatures are not sufficient means for automatic verification since even if a signature can be verified; there is no guarantee of the fact that the person who made the signature is who he claims to be. Public key certificates are a powerful means of establishing trust in public key cryptography. A certificate is someone's public key, signed and packaged for use in a public key infrastructure [5]. In general, a certificate contains the following three pieces of information: i) the name of the subject for whom the certificate has been issued, ii) the public key associated with the subject, and iii) a digital signature signed by the issuer of the certificate. The digital signature will verify the information of the certificate, and if the verification succeeds it is assured that the public key in the certificate does in fact belong to the entity the certificate claim[6]. A certificate may also contain information related to the secret key and the signed public key. The trustworthy party naturally signs this extra information along with the key.

The above technologies together help in setting up secure environments for mobile payment, which we introduce in the next section.

3. Mobile security technologies

The following protocols and technologies facilitate the handling and transmitting of sensitive payment information to and from the mobile devices in an M-payment transaction.

3.1 Wireless application protocol (WAP)

The WAP forum has specified a series of protocols, which cover all the protocol layers from the transport level to the presentation layer. The functional areas related to security in WAP considered include Wireless Transport Layer Security (WTLS), Wireless Identity Module, WAP Public Key Infrastructure, WML Script signText, and End-to-End Transport Layer Security.

The WTLS (Wireless Transport Layer Security) protocol is a PKI-enabled security protocol, designed for securing communications and transactions over wireless networks. It is used with the WAP transport protocols to provide security on the transport layer between the WAP client in the mobile device and the WAP server in the WAP gateway. The security services that are provided by the WTLS protocol are authentication, confidentiality and integrity. WTLS provides functionality similar to the Internet transport layer security systems TLS (Transport Layer Security) and SSL (Secure Sockets Layer), and has been largely based on TLS, but has been optimized for narrow-band communications and incorporates datagram support. WTLS is implemented in most major micro-browsers and WAP servers. WAP 1.x series use the WTLS protocol to protect messages in the wireless network part and someway into the wired network, that is, between the wireless device and WAP Gateway. The WAP gateway transforms the WAP 1.x stack to/from the wired TCP/IP stack, relays the data between the wireless and wired network, and communicates with the Web Server that the mobile device is accessing.

Wireless Identity Module [7] (WIM) is used in performing functions related to WTLS and application level security by storing and processing information like secret keys and certificates needed for authentication and non-repudiation. To enable tamper resistance, WIM is implemented as software on a microprocessor-based smart card. WMLScript [8] signText includes support for digital signatures of WML (display format of data in wireless world analogous to HTML) coded content. SignText function allows a wireless user to digitally sign a transaction in a way that can be verified by a content server. This provides end-to-end authentication of the client, together with integrity and non-repudiation of the transaction.

WPKI [9] is an optimized extension of a traditional PKI for the wireless environment. WPKI requires the same components as a traditional PKI: an End-Entity Application (EE), a Registration Authority (RA), a Certification Authority (CA) and a PKI Repository. In WPKI, the end entities (EE) and the registration authority (RA) are implemented differently, and a new entity, referred to as the PKI Portal, is introduced. The EE in WPKI runs on the WAP device. It is responsible for the same functions as the EE in a traditional PKI. The PKI Portal can be a dual-networked system, like a WAP gateway. It functions as the RA and is responsible for translating requests of WAP clients to the RA and interacts with CA over wired network. The RA validates the EE's credentials to approve or reject the request to receive a digital certificate.

The WAP PKI defines three levels of transport layer session security, WTLS classes 1, 2 and 3, and a signText --WMLScript functionality for digital signatures. WTLS Class 1 provides encryption; WTLS Class 2 provides encryption and gateway authentication; WTLS Class 3 provides encryption and two-way authentication. The WMLScript signText is a functionality that the user interface can utilize for creating digital signatures. The signText uses the underlying security element WIM (Wireless Identity Module) that actually performs the cryptographic procedures and stores the secret keys securely. Basically, WPKI is concerned with the requirements on a PKI imposed by WTLS and the sign Text function. The WPKI architecture for WAP 1.x series is shown in Figure 1.

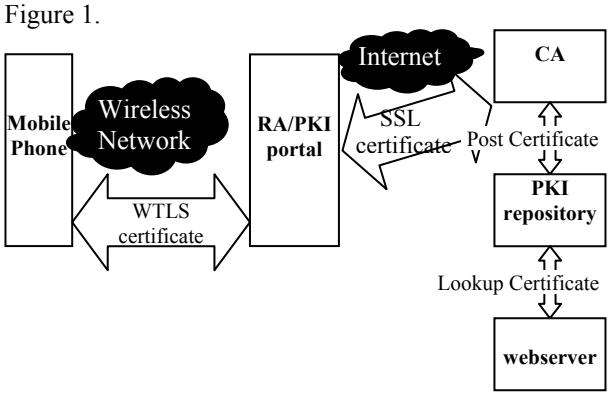

Figure 1: WPKI architecture for WAP 1.x series

The merchant server authenticates itself by sending its digital certificate (SSL certificate) to the WAP gateway, which will have the root certificate of the CA that issued the merchant servers digital certificate. Similarly the WAP gateway will authenticate itself to the mobile client by passing its digital certificate to the mobile client. The mobile client in turn will have the root certificate of the CA that issues the gateway's certificate. The root certificate in the mobile client is stored in the wireless identity module (WIM) in the form of a compact certificate called WTLS certificate. For security the mobile client should also be able to check whether the WAP gateway certificate has been revoked. Even there are a number of solutions for checking certificate revocation in the wired world, the same cannot be applied to the wireless world due to its many constraints, and hence the solution is to issue short-lived WTLS certificates to the gateway. For mobile client authentication, two methods can be applied: i) using WTLS class 3 between the client and the gateway, ii) using WMLScript digital signatures between the mobile client and the merchant's server. These methods require a private key and a digital certificate to be stored in a WIM. For client authentication, the client should have an URL pointer to the location of the complete SSL certificate, which is too large to store in a mobile phone. All the members involved in a mobile payment system can access the full version of the SSL certificates.

3.2 SIM Application Toolkit (SAT)

The GSM (Global System for Mobile Communications) Subscriber Identity Module (SIM), which stores personal subscriber data, can be implemented in the form of a smart card called SIM card. SIM toolkit is a specification of SIM and terminal functionalities that allow the SIM to take control of the mobile terminal for certain functions. SIM application toolkit (SAT) is used to create Short Message Service (SMS) based mobile payment applications. In SIM Application toolkit based systems, the communication between the mobile client and the payment server occurs using SMS. The SMS is used to initiate and authorize payments. The user is identified and authenticated by GSM authentication service and hence the GSM mobile network operator acts as an intermediary between the mobile client, the payment server, and the merchant [10].

SAT provides confidentiality, authentication, integration, and message replay protection, but does not provide denial of service or non-repudiation. This lack of support for non-repudiation is a major hindering factor for the adoption of SAT mobile commerce applications. SAT has built in support for data encryption standards including triple DES. The service provider places the encryption key before the SIM is issued to the customer. This ensures that secret key never goes over the air interface. Authentication is provided by strong authentication algorithms, which can be chosen by the payment provider. Data integrity is realized using message digests like SHA and MDS 5. Other than not providing support for prevention of non-repudiation, the SAT also has another flaw caused by its usage of the mobile clients PIN code. PIN codes are usually 4 digit numbers, which can be guessed and entered into stolen

or lost mobile phones, and undo the security provided by encryption algorithms or large keys. Security requirements of SIM Toolkit [11] cover the transport layer security issues, such as peer authentication, message integrity, replay detection and sequence integrity, proof of receipt, and message confidentiality. Each payment application message is divided into packets that are individually secured by protecting the payload and adding security headers. *Proof of execution* is required as well, to assure the sending application (e.g., a bank application) that the receiving application (e.g., the home banking application on a SIM card) has performed an action initiated by the sending application. This proof should be provided at the application layer, so no mechanism for it is defined in the GSM specifications.

3.3. Java 2 Platform, Micro Edition (J2ME)

There are two relevant JSRs (Java Specification Requests) in relation to cryptography and secure m-payments according to the Java community process program: *Mobile Information Device Profile*, MIDP 2.0 [12] and *Security and Trust Services API for J2ME* [13]. MIDP 2.0 is a specification of a security framework for Java applications designed to run within the MIDP Java environment. MIDP uses a Java Virtual Machine of reduced complexity designed specifically for mobile devices. MIDP 2.0 specifies how a signed Java application can be verified to belong to a domain defined by a root certificate and an associated policy file. The policy file specifies the capabilities of Java applications within that domain.

M-payment solutions based on MIDP/SIM APIs, SSL and the Java Card platform provide greater transaction security and network efficiency. J2ME combined with MIDP is used to create a number of java based mobile applications for various types of mobile devices. J2ME can be used together with other protocols like WTLS to be used in WAP enabled phones. J2ME's MIDP platform is said to have the following advantages as regards to security, which makes it attractive: *Transaction protection*, in which the complete transaction is encrypted and with the support of WAP/WTLS, the entry session can be protected as being performed across SSL3.0.; *Cryptography*, which provides a Security and Trust Services API for J2ME [14] .

Security and Trust service API aims to develop application-programming interfaces for cryptographic operations, which support the payment method, and also provides data integrity and confidentiality. This API will be a part of MIDP 2.0. Given the importance of HTTPS in relation to M-commerce, Sun Microsystems had added an unofficial support for HTTPS (kssl) as a part of the MIDP 1.0.3 reference implementation and the J2ME Wireless Toolkit version 1.0.3. Encryption relies on a corresponding key (symmetric or private key) that is accessible to the MIDlet (J2ME payment application) during runtime. J2ME provides facilities to use and store encryption keys. Keys can be stored and updated in the record store or in a unique resource file generated at deployment.

4. Analysis and discussion

In contrast to many areas, research and development in the area of M-commerce are mainly initiated and done by industry. The reason is that mobile devices are already widespread in use and hence vendors are developing new value-added services. In the course of this process, the old concepts such as the Web are basically being accommodated. This allows faster development and immediate customer acceptance. However, many privacy concerns and advanced security concepts still need to be addressed. This section provides an analysis of the current mobile payment security and discusses security issues for mobile payment.

4.1. Analysis of current techniques

In WAP, security is provided through Wireless Transport Layer Security (WTLS) protocol (in WAP 1.0) and IETF standard Transport Layer Security (TLS) protocol (in WAP 2.0). They provide data integrity, privacy, and authentication. The feature of data integrity ensures that the content of messages is not altered during transmission. Privacy makes sure that only the intended recipients can read the original content. Authentication verifies the identities of communication participants. One security problem, known as the "WAP gap," is caused by the existence of a WAP gateway in a security session, in which encrypted messages sent by end systems might temporarily become clear text on a WAP gateway when messages are processed.

In SIM toolkit, security requirements cover the usual transport layer security issues such as peer authentication, message integrity, replay detection and sequence integrity, proof of receipt, and message confidentiality. Each application message is divided into packets that are individually secured by protecting the payload and adding security headers. Proof of execution is required to assure the sending application that the receiving application has performed an action initiated by the sending application. This proof should be provided at the application layer and hence it is not standardized. SIM toolkit is based on SMS. The sender and receiver of an SMS are identified, and an attacker cannot forge without breaking the network security mechanisms such as cloning a SIM card. Hence, SMS messages can be used for authentication. Furthermore, SMS data is transmitted in the network-signaling plane, which ensures the confidentiality of messages. However,

the protection ends in the network, there is no end-to-end security, and the network operator and its infrastructure (SMSC, Short Message Service Centre) must be trusted when no other security mechanisms are applied to the SMS message.

J2ME provides several levels of security, such as class loader, byte code verifier, and security manager. These security levels protect client systems from unreliable programs. The security advantages of J2ME over WAP are end-to-end security, less use of network and content-based encryption.

1) **End-to-end security.** J2ME supports end-to-end encryption, authentication, and verification. In WAP, a request from a wireless device is encrypted in WTLS and this request needs to be decrypted as Transport Layer Security (TLS) data. While this conversion takes place, the data is unencrypted making it highly vulnerable. J2ME does not need a gateway between the device and the server. This allows J2ME to provide end-to-end security. There is no conversion of data from WTLS to TLS, thereby eliminating the chance of the data being unencrypted at any point of time. End-to-end security is through SSL (kssl).
2) **Less use of network.** J2ME allows data to be processed locally, unlike WAP that needs to connect to the network for any kind of data processing. This feature in J2ME in turn reduces the possibility of data loss or theft.
3) **Content-based encryption.** J2ME applications process data before sending it across a network. A J2ME application can set the security policy based on the content.

HTTPS is required in the MIDP 2.0 API, released in 2002. The best possible implementation of HTTPS should be coordinated between manufacturers to ensure homogeneity across devices and compatibility of all secure J2ME applications. The implementation should also avoid the WTLS specification to ensure end-to-end and independency of WAP-gateways. Compared to WAP and SAT, the MIDP 1.0 API provides enhanced GUI and UI possibilities. Since graphics are used and generated locally on the device, network bandwidth usage will be reduced and the performance be enhanced. Compared to WAP-based payment, all business logic is fetched from the web server and usually no new software or hardware is required on the device. New hardware may be required for SAT/WAP-payment, if the application logic depends on a wallet or keys stored in a SIM/WIM. The end-user could then be compelled to upgrade the SIM.

4.2. Discussion

We address several research issues, including access to network without prior relationship, security standard, and PKI complexity management, which will enhance the security of mobile payment for large-scale deployment and further development of M-commerce.

4.2.1. Access to a network without prior relationship. PKI, tailored for wireless environments, is currently used by a number of security protocols to enable (end-to-end) security for services and applications such as WAP. In these systems, PKI is not used for securing network access, because there always exists a relation between the service provider and the mobile subscriber, allowing the use of symmetric key based methods, which are efficient to be used in a mobile environment. Future concepts for mobile devices accessing networks will be by gaining access to a network without a prior relationship to the network provider. This may be achieved by using account-based payment protocols, a joint-signature scheme, or by following a policy based mobile payment architecture.

Account-based payment protocol: Account based payment protocol uses symmetric key techniques, such as Message Authentication code (MAC) and hash functions [15]. This protocol not only reduces the amount of computation to be performed, but also satisfies all the transaction security properties, including non-repudiation property, which is only available to PKI protocols.

A joint-signature scheme: A joint-signature scheme acts as an alternative to traditional digital signatures [16]. This scheme is based on collaborative use of one-way hash functions and traditional digital signatures with the network operator. This scheme not only reduces the mobile computation costs, but also provides lower communication cost as opposed to other digital signature security schemes. This joint-signature scheme is based on the hypothesis that if a third party, like the network provider which has with ample computation and communication resources, signs a digital signature containing a secret that is only shared between the customer and the merchant, then the merchant can treat the digital signature as a joint signature originated from the customer and signed by the third party/network provider.

Policy-based mobile payment architecture: The information model and architecture for policy based mobile payment server is based on a number of policy-related constraints and rules for customers, merchants, and payment providers. The information model is based on PCIM (Policy Core Information Model) developed by the IETF (Internet Engineering Task Force) Policy group. This information model describes the concepts of

policy groups like rules, conditions, actions, repositories, and relationships. The policy-based payment server consists of policy rule repository, which stores the policy rules, conditions, actions, and other related policy data. Policy-based payment engine evaluates the policy conditions and triggers appropriate actions. Future research on this architecture will focus on the mobile payment specific extension of this model.

4.2.2. Security standards: Currently, there is no common agreement as to how PKI-related tasks should be divided between the mobile devices and network agents. As mobile devices are constantly gaining in processing power, the network agents will probably cover less PKI functionality. This issue creates instability for wireless PKI clients. Hence, there is an obvious need for standardizing wireless PKI clients. In addition, an open security standards are required to ensure that the wireless infrastructure can be created for secure transactions between parties that have had no prior relationship [17].

There is only limited serious standardization work in this area, and the implementation of wireless PKI leads to solutions that are not open to other PKI software providers. The Security Group of the Wireless Application Forum (WAP Forum) is a good example of attempts in standardization work. But forums of this type concentrate on specific environments like WAP, with other potential mobile devices and WPKI environments remain undefined. More and more organizations are now aware that by ensuring interoperability across solutions, services, and platforms, we can create a more significant impact on the security of payments. There is increasing appeal for adopting international standards and specifications produced by open industry consortia.

4.2.3 PKI complexity management: PKI is a solid concept for providing security. However, a number of challenges have to be overcome for widely adoption in future mobile systems, which are highly heterogeneous in nature. These challenges include complexity management of PKI in limited devices like mobile phones, complexity control of PKI for limited bandwidth, and interoperability and organization issues for deploying PKI in large-scale heterogeneous mobile systems.

5. Summary

In this paper, we introduce mobile commerce and mobile payment, discuss the backgrounds on public key infrastructure as a basis for security in different mobile technologies, and study the security measures in mobile security technologies which employed in the current M-commerce market. In addition, we compare the effectiveness of WAP, SIM application toolkit, and J2ME in supporting a secure M-commerce payment application, and address several research issues in M-commerce payment security. Our future research work will be concentrating on the formal analysis of the security strength and effectiveness of these supporting technologies.

Reference

[1] D. Lonergan, "Mobile Commerce Market opportunity," pp. 48, 1999.

[2] S. Chari, P. Kermani, S. Smith, and L. Tassiulas, "Security Issues in M-Commerce: A Usage-Based Taxonomy," in *E-Commerce Agents, Marketplace Solutions, Security Issues, and Supply and Demand*. London: Springer-Verlag, 2001, pp. 264 - 282.

[3] *Security for Mobility*. London: The Institution of Electrical Engineers, 2004.

[4] N.-J. Park and Y.-J. Song, "M-Commerce security platform based on WTLS and J2ME," presented at Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium on, 2001.

[5] C. Peikari and S. Fogie, *Maximum Wireless security*, 1st Edition ed: Sams Publishing, 2002.

[6] S. Oaks, *Java Security*, 1st ed: O'Reilly & Associates, Inc, 1998.

[7] WAP Forum, "Wireless Identity Module,Candidate Version 1.2," vol. 2004: Open Mobile Alliance Ltd, 2004.

[8] WAP Forum, "WMLScript Crypto Library," in *Wireless Application Protocol*, vol. 2004, Version 05-Nov-1999 ed, 1999.

[9] K. Raina and A. Harsh, *mCommerce Security: A Beginner's guide*, 1st ed: McGraw-Hill Companies, 2002.

[10] S. F. Mjølsnes and C. Rong, "On-line e-wallet system with decentralized credential keepers," *Mob. Netw. Appl.*, vol. 8, pp. 87--99, 2003.

[11] ETSI, "Digital cellular telecommunications system (Phase 2+): Security mechanisms for the SIM Application Toolkit;Stage 1GSM 02.48 version 8.0.0 Release 1999," vol. 2004: ETSI TS 101 180 V8.0.0, 1999.

[12] Sun Microsystems Inc., "JSR 118: Mobile Information Device Profile 2.0," vol. 2004, 2002.

[13] Sun Microsystems Inc, "Security and Trust Services API for J2ME™," vol. 2004, 2003.

[14] R. Vichr, "Tips & tricks: m-payments with J2ME," vol. 2004, 2002.

[15] S.Kungpisdan, B.Srinivasan, and P. D. Le, "A secure account-based mobile payment protocol," presented at Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on Volume: 1 , April 5-7, 2004.

[16] L.-S. He and N. Zhang, "A new signature scheme: joint-signature," presented at Proceedings of the 2004 ACM symposium on Applied computing, Nicosia, Cyprus, 2004.

[17] S. Pugh, "MasterCard Leads the Way in Securing Mobile Payments," in *Wireless Business & Technology*, vol. 3, 2003.