A SECURITY TUNNEL FOR CONDUCTING MOBILE BUSINESS OVER THE TCP PROTOCOL

C. Leonidou, A. S. Andreou, A. Sofokleous, C. Chrysostomou, S. Mavromoustakos, A. Pitsillides, G. Samaras, C. Schizas

Department of Computer Science, University of Cyprus, 75 Kallipoleos Street,
P.O. Box 20537, 1678 Nicosia, Cyprus,
Phone: +357 22 892700, Fax: +357 22 892701,
e-mail: {leonidou, aandreou, cchrys, , cspgms1, andreas.pitsillides , cssamara, schizas}@ucy.ac.cy

ABSTRACT

This paper discusses security issues in the Mobile Business area with emphasis on the significance of mobile commerce and provides a framework to address and analyse common security problems of m-Business services and applications. We propose a new model to address efficiently some security problems in mobile environments, which rescinds Gateway/Proxy intermediates and establishes a secure end-to-end communication between mobile users and service providers over the TCP protocol. In this context, an application prototype is developed in J2ME, which implements and demonstrates the proposed model.

1. INTRODUCTION

If we attempt to define what is Mobile Business (m-Business) it is likely to conclude that m-Business is any business operation that is conducted via a mobile telecommunications network. This definition can be expanded both in the business-to-customer and business-to-business area of application. Thus, m-business is the framework that allows us to conduct business by using Mobile Telecommunication Networks and Wireless LANS, as well as appropriate Information Technology Infrastructure. What is the reason, though, we need this new framework in our life for doing business?

The fact that the penetration of mobile devices (telephones, PDAs, etc) increases dramatically, shows that mobile devices have become essential to our way of living (Gemplus, 2002). This also reveals the human need for mobile communication and computation. Mobility and nomadic behaviour are human characteristics that were suppressed by the limitations of the traditional "wired dependant" communications and computation models. These models, due to technological limitations, ignored the need to be able to communicate and perform computation activities in a mobile environment without the communication and computation devices restrictions in terms of mobile communication networks, power and size.

The new technological achievements came up with solutions to the human need for mobile communications and computation. Mobile and wireless networks become faster and faster in terms of transfer rates. Moreover, today's mobile devices use less electric power, while batteries become smaller, lighter and their endurance increases. If we also consider the fact that new applications and services are released everyday, then it should not be surprising that the number of mobile users is constantly increasing.

The fact that mobile communication and mobile computation are becoming increasingly popular it creates a trend, which positively influences the volumes of business operations conducted within the framework of mobile business. It is estimated that the global market for m-business will reach 200 billion USD by 2004, with 130 million users conducting about 14 billion m-business operations. In addition, study forecasts show that the number of m-business users will reach 500 million by 2005 (Gemplus, 2002).

However, there are some serious considerations concerning the level of security in mobile business. The current mobile business environment is based on the Internet model. This means that most mobile services and applications run over TCP/IP protocol stack, enjoying its advantages and bearing its disadvantages. Thus, the security level of most mobile business services and applications is equivalent to the Internet security, something which means authentication and authorization of logins and passwords, and most of the times transmission of unencrypted data. In addition, data transmission over radio frequencies is very vulnerable and everyone can easily eavesdrop, something that reveals a highly unsecured environment for conducting m-business.

The security problems mentioned in the previous paragraph should not discourage us from conducting mobile business but rather find new methods for securing the mobile environment. This paper provides ways to secure data transmission in mobile (and wireless) environments, addresses main security issues like authentication, authorization, non-repudiation, etc. Section 2 reviews some security issues concerning cellular networks and wireless LANS and presents their impact on mobile business. Section 3 proposes a solution, which will increase the level of security for mobile services and applications over TCP/IP networks, while section 4 demonstrates a prototype-software application which implements some of the concepts discussed in section 3. Finally, in section 5 we present our concluding remarks and offer suggestions for further research on the subject.

2. REVIEW OF SECURITY ISSUES IN MOBILE BUSINESS

Mobile business services and applications require strong user authentication, integrity and confidentiality. In this environment, we also require identification, non-repudiation and service availability, which mostly concerns service providers. Because of these requirements, carriers (telecommunication operators and access providers), service and application providers, as well as users demand end-to-end security where applicable.

Technologies used to implement m-business services and applications like iMode, Hand-held Device Mark-up Language (HDML) and Wireless Access Protocol (WAP) 1.1 can secure the transport of data (encryption) between clients and servers, but they do not provide applicable security layers, especially user PIN-protected digital signatures, which are essential for secure transactions. Therefore, consumers cannot acknowledge transactions that are automatically generated by their mobile devices.

In order to analyse the security issues for mobile networks and services, one has to segregate the problem into sub-problems and address each one of them separately. The fist sub-problem is the security level of the access network. By the term “access network” we mean the cellular or wireless LAN networks which offer “physical” access to a network. The second sub-problem lies with services and application security issues encountered when conducting mobile business transactions. Each of those sub-problems is addressed as follows:

2.1. Access network security problems

In reference to security issues, cellular networks and wireless LANs were designed and implemented having the same philosophy used in the design and implementation of the traditional wired telecommunications networks and wired Local Area Networks. The main goal was to ensure that the wireless link would be equivalent to the “wired” one. This kind of approach was the cornerstone of the

Wireless LANs of the IEEE 802.11x family of design and implementation. The approach remained the same for the design and implementation of the GSM, Bluetooth and in general, most of the cellular networks and wireless LANs. As a result of this, we are now facing some security vulnerabilities and weaknesses in these types of networks.

Schuba and Wrona (2001) report several security vulnerabilities for GSM. For example, one may take advantage of the "weaknesses" of the authentication protocols of GSM and determine the International Mobile Subscriber Identity (IMSI). Based on this information, the intruder may signal the mobile phone to stop encrypting the information exchanged between the phone and its base station. Another vulnerability of GSM is the fact that some operators use radio links to connect their base stations. Often, these links offer no encryption at all. Finally, a number of attacks have been observed on the stream cipher algorithm of GSM, namely the A5 algorithm.

Although today the GSM network provides a relatively secure connection through the PIN (Personal Identification Number) inserted when turning on the handset, as well as through the authentication protocol between the handset and the network based on SSL encryption of voice and data, it is not enough to convince people that they can perform secure transactions over a mobile network. On the other hand, smartcards seem to become the preferred way of gaining access to a secure system. The smartcard can be in the form of a credit card or in the form of a SIM-like miniature card. It is possible to run a variety of applications on a single small SIM card. Encryption is being used to ensure confidentiality through a secret key and an appropriate algorithm. This algorithm produces a scrambled version of the original message that the recipient can decrypt using the original key to retrieve the content. The key must be kept secret from the outside world and be shared only between the two parties involved.

There are two basic methods, which can be used to encrypt a document: symmetric and asymmetric. With the symmetric method the same key is used for encryption and decryption. The problem is that the key has to be transmitted to the recipient of the message, and a third party could gain access to the key during this transmission. People have come up with various solutions in order to solve the key distribution problem. The most popular of these methods is the Internet Key Exchange (IKE) (Harkins & Carrel, 1998) and the utilization of an asymmetric cryptography mechanism (RSA1977). Within the symmetric encryption process both parties have typically a key of 1024 – 2048 bits. Using the asymmetric algorithm, also known as public key method, a set of two keys is used – a private and a public key. Information encrypted using the public key can only be retrieved using the complementary private key. With this type of encryption the public keys of all users can be published in open directories, facilitating communications between all parties. In addition to encryption, the public and private keys can be used to create and verify digital signatures. Today, symmetric encryption is the most common way to ensure secure communication, but the asymmetric method has more benefits and is therefore becoming more and more popular in the wireless world. Regardless of the specific method used, the encryption is between 40 – 120 bits depending on the country, network and the level of security required.

Schuba and Wrona (2001) describe a certain vulnerability related to the Bluetooth mechanism, which is located in the key exchange protocol used by Bluetooth devices. A bugging device within the coverage area of the Bluetooth devices can "listen" to a conversation thus being able to obtain enough information to determine the encryption key used by the participating devices. Gaining access to the encryption key is not difficult, thus a potential intruder can illegally participate in the ongoing conversation. Schuba and Wrona (2001) also refer to some other weaknesses of the Bluetooth system related to the stream cipher used and the generation of the initialization key based on the PIN code.

The IEEE 802.11x family enhances its security by using the Wired Equivalent Privacy (WEP) protocol, which implements mechanisms for confidentiality and integrity of the data to be exchanged wirelessly. WEP uses the well-known RC4 stream cipher for encryption, operating by expanding a

short-shared secret key (40 to 104 bits in the case of WEP) into a pseudo-random key stream. Packets integrity is ensured by using an Integrity Check field. To avoid encrypting two packets with the same key stream, an Initialization Vector is used together with the shared secret key in order to generate a different RC4 key for each packet.

2.2. Security issues related to services and applications

There are some security aspects that have been more or less standardized, providing a framework to study and categorise the various security problems. In order to enforce a high level of security we have to enable certain aspects, which are very important, especially in the service and application layer where the user interacts directly with the service provider. In general, the mobile user and the e-business provider involved in an m-business operation demand: (i) Confidentiality - messages are kept secret, (ii) Authentication - each party knows who the other party is, (iii) Message integrity - messages are passed unchanged from sender to receiver, (iv) Spoof attack prevention - any unauthorized re-sending of messages is detected and rejected, (v) Non-repudiation - neither party can later deny that the exchange took place. All these issues must be properly addressed in order to establish a secure system. We are going to use the Hypertext Text Transfer Protocol Secured (HTTPS) as a pilot service/application protocol in order to address certain security vulnerabilities. Most of the concepts behind these vulnerabilities are the same as with other services/application protocols, especially in those which use a Gateway or Proxy.

The Wireless Application Protocol (WAP), used mainly by cellular telephony devices, addresses the above problems (Juul & Jørgensen, 2002) mainly by providing a secure protocol for data transport called WTLS (Wireless Transport Layer Security). WTLS contains features for authentication for both parties, as well as for non-repudiation using message digests and digital signatures. The process of authenticating a GSM phone user may utilize the SIM card as mentioned in Section 2.1. The crucial weakness here lies with the fact that all data transferred between the WAP client and the web server is decrypted at the WAP Gateway, i.e. sensitive data, such as credit card numbers, exists as free text in the memory of the gateway. Thus, essentially there is no end-to-end security. The solution to this problem could be a secure tunnel over the TCP protocol which can ensure a secure end-to-end communication between the user and the service/application provider. We will elaborate on the subject and describe how this can be implemented in the next section.

3. SECURE COMMUNICATION OVER THE TCP PROTOCOL

Section 2 briefly discussed various security issues related to the mobile business environment. If one would like to address the security issues regarding the mobile business environment in a more general form then he must try to establish a secure tunnel between the mobile user on a cellular network or a Wireless LAN and a service or application provider on any network (wired, cellular or wireless), or another peer in the same network as the first peer.

In the case of Wireless LANs, where devices with strong computing capabilities and standard operating systems (i.e. laptops) are available and which run universal protocols (i.e. TCP/IP, IPSec, HTTPS), security problems are substantially reduced. Using these protocols which run in the service providers premises, one can easily establish a secure end-to-end tunnel since there are no intermediates like Gateways or Proxies, so security is an end-to-end matter and security degradation is avoided.

However, in cases in which the devices have proprietary (non-standard) operating systems, or simply they cannot run the same service protocols as their corresponding service providers, we have to use intermediates. As mentioned before, most of the security vulnerabilities are caused by the fact that there is not an end-to-end secure tunnel. Therefore, our proposition is to try to build one. This can be done by having enforced what we call “Application Level Security”. We choose “Application Level Security” versus “Network Level Security” for various reasons. The most important of these reasons is

the fact that Network Level Security typically requires client software installation but it may not be possible to support all required client operating systems. The installation of client software is not very convenient for the mobile commerce model of computing where a number of independent service providers is available and each one of them requires the installation of a client software for enhancing network security. Our aim is to provide the mobile user with a transparent security mechanism.

In order to enforce "Application Level Security" one must develop a way to run the service protocols on mobile devices. The fact that most of these services protocols run over TCP/IP supports our suggestion. Thus, what is required is to develop customized applications that run on the mobile devices over the traditional TCP/IP protocol stack and establish a secure tunnel with the service provider on the application layer. These customized applications need to implement the application layer protocols of the TCP/IP protocol family since they will communicate with the services and applications through the Application Layer Protocols.

Fortunately, some mobile devices vendors (e.g. Nokia, Samsung, Sony-Erickson, Siemens, etc.) have equipped their devices with versions of the Java Virtual Machine (JVMs). Using these JVMs we can easily develop applications which can guarantee an end-to-end secure communication tunnel between the user and the service/application provider. The following section will present an example of this implementation. If we succeed to make the role of the intermediate entity (gateway or proxy) "standing" between the user and the service/application provider obsolete, apart from the increase of the security level we will also be able to simplify the model of communication offering a better environment for conducting mobile business. The service providers will be able to unify and standardise their services offered on the Internet via wired networks and those offered to their users via cellular networks. The protocol stack block diagram of the Gateway/Proxy Approach is depicted in Figure 1 and the corresponding diagram of the proposed Direct-to-Service Approach is shown in Figure 2.

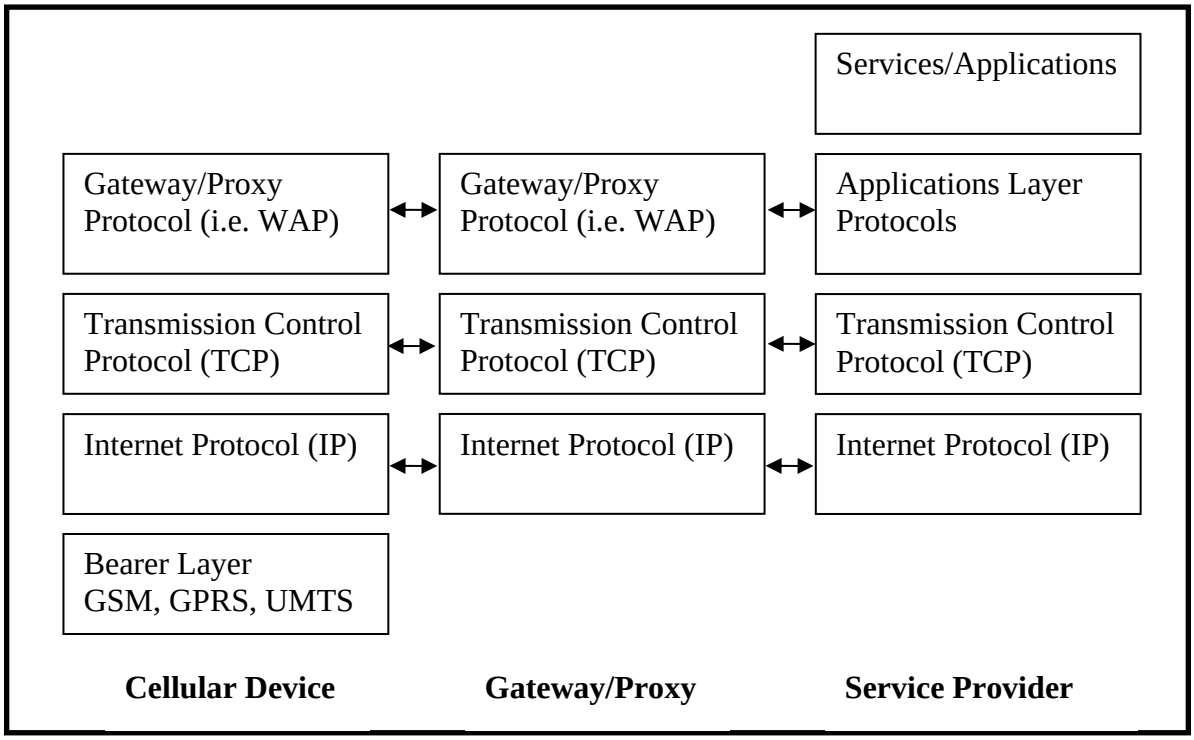

Figure 1: Gateway/Proxy Approach Protocol Stack Diagram

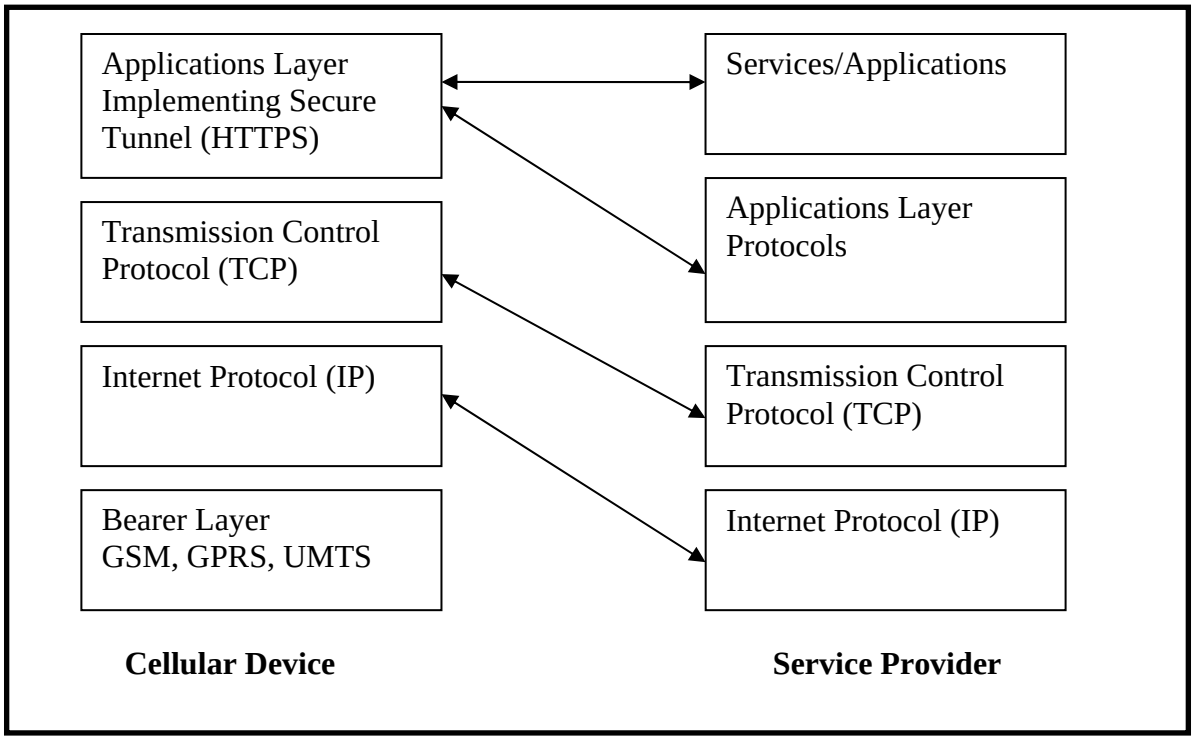

Figure 2: Direct-to-Service Approach Protocol Stack Diagram

4. A PROTOTYPE APPLICATION IMPLEMENTING THE DIRECT TO SERVICE APPROACH

This section demonstrates the proposed Direct-to-Service approach through a prototype application that was developed and tested on Pocket PC 2002 mobile device running Java 2 Micro Edition. The prototype implements secure connections with service providers running HTTPS. We provide the code necessary for the connection establishment and an example of the operation. Our aim was not to develop an application that will help us browse the web. We rather aimed at demonstrating how the proposed Direct-to-Service approach can prove a better model than the Gateway/Proxy approach and that the proposed model can easily be implemented. In the following lines we provide some definitions about the Mobile Information Device Profile (MIDP) that was used in our implementation.

The MIDP combined with the Connected Limited Device Configuration (CLDC), is the Java™ runtime environment for today's mobile information devices (MIDs) such as phones and entry level PDAs. Version 1.0 of the MIDP provides a standard API for application development. The only network protocol the MIDP 1.0 specification requires is HTTP and the support of any other protocol is optional. One possible implication is that the support of socket or datagram connections is not required. The easiest way to exploit HTTPS support in the J2ME Wireless Toolkit is to create an HTTPS URL object as follows:

```
String url = "https://myhost.com/somefile";
HttpConnection c = (HttpConnection) Connector.open(url);
```

Version 2.0 of the MIDP also requires HTTPS, which is basically HTTP over the Secure Sockets Layer (SSL). SSL is a socket protocol that encrypts data sent over the network and provides authentication for the socket endpoints. MIDP 2.0 provides a stable, consistent foundation for wireless

applications that deal with financial transactions or the exchange of sensitive information. MIDP 2.0 has added new packages, including javax.microedition.pki which enables the use of certificates to authenticate information for secure connections. A normal HTTP connection can be obtained as follows:

```
String url = "https://www.cert.org/";
HttpsConnection httpsConnection=null;
httpConnection = (HttpsConnection)Connector.open(url);
SecurityInfo securityInfo = httpConnection.getSecurityInfo();
Certificate certificate = securityInfo.getServerCertificate();
String name = certificate.getIssuer();
```

HttpsConnection, an interface from package javax.microedition.io., defines the necessary methods and constants to establish a secure network connection. Object “securityInfo”, an instance of Class javax.microedition.io.SecurityInfo, contains information about a secure connection. It provides the certificate, protocol, version, and cipher suite, etc. in use. Object “certificate”, an instance of the Class javax.microedition.pki.Certificate, represents a cryptographic certificate.

Following, we provide an example of accessing an HTTPS site and retrieving information through the prototype application. The URL we will use is https://central.sun.net (Figure 3).

Figure 3: URL Selection Application

After the URL selection, a new form appears (Figure 4) and informs the user about an HTTPS transaction.

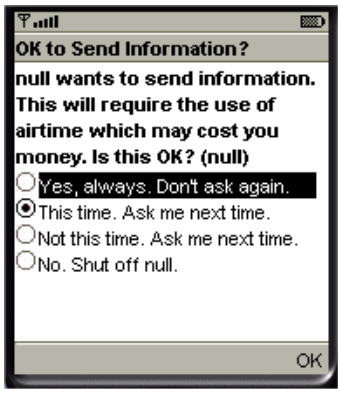

Figure 4: Information About Transaction

If the user confirms the transaction then the relative information about the HTTPS connection is displayed as depicted in Figure 5.

Figure 5: HTTPS Information

The previous example demonstrates retrieving security information from an HTTPS Connection. With this capability, the Java-enabled mobile gadget can act as a secure mobile application device.

During our test, we managed successfully to connect to a Secure Server that uses SSL 124bit security and retrieve all the necessary information we have requested. Due to the fact that our primary focus was only to demonstrate the proposed Direct-to-Service approach we did not develop further the software application.

5. CONCLUSIONS – FURTHER WORK

Mobile business services and applications require strong user authentication, integrity and confidentiality, as well as user identification and non-repudiation. Therefore, it is only natural that carriers (telecomm operators and access providers), services, application providers and users demand end-to-end security where applicable.

In this paper, we proposed a new approach that obsoletes the Gateway/Proxy, enhances security in m-Business transactions and assists the service providers to implement new services for mobile users. Our solution is applicable only with mobile devices able to run J2ME over the TCP/IP stack. The fact that more and more mobile devices manufacturers equip their products with the ability to run Java Micro Edition applications is obviously in favour of our proposition. Using the proposed approach service providers will be able to develop services over well known and widely used protocols resulting to better and more secure mobile business services.

Our further work will focus on refining and improving the Direct-to-Service model, as well as on developing a variety of prototype services and applications in order to test its wide-range applicability and effectiveness.

6. REFERENCES

Gemplus (2002). *Mobile Commerce Security: Essential and Available* [available online from: www.wmrc.com/businessbriefing/pdf/mcommerce2001/tech/Gemplus.pdf, last accessed 23 March 2002].

Schuba, M. & Wrona, K. (2001). Security for Mobile Commerce Applications. *IEEE/WSES International Conference on Multimedia, Internet, and Video Technologies (MIV '01),* Malta, September 2001.

MobileInformationDeviceProfile (2002) *What's New in MIDP 2.0 by Jonathan Knudsen November 2002* [available online from: wireless.java.sun.com/midp/articles/midp20/, last accessed 10 March 2003].

Juul, N. C. & Jørgensen, N. (2002). Security Issues in Mobile Commerce using WAP. *15th Bled Electronic Commerce Conference.* e-Reality: *Constructing the e-Economy.* Bled, Slovenia, June, 2002.

WirelessMarkupLanguage (2002). *Java 2 Micro Edition API* [available online from: www.wmlscript.it/j2me/api/, last accessed 10 March 2003].

Harkins, D. & Carrel, D. (Nov. 1998). RFC2409. The Internet Key Exchange (IKE).

RSA Security (1977). *RSA Security FAQ* [available online from: www.rsasecurity.com/rsalabs/faq/files/rsalabs_faq41.pdf, last accessed 20 May 2003].