COMPLEMENTARY MECHANISMS FOR SECURE MOBILE ELECTRONIC COMMERCE
D. Ponce, M. Soriano, S. Montardit, F. Barceló
{dponce, soriano, barcelo}@entel.upc.es
Departament d’Enginyeria Telemàtica. Universitat Politècnica de Catalunya (UPC)
C/ Jordi Girona 1,3, Módulo C3. Campus Nord UPC
08034 Barcelona (Spain)

ABSTRACT
The possibility of making the Internet accessible via mobile telephone has generated an important opportunity for electronic commerce. Nevertheless, the performances of mobile devices and intrinsic characteristics of wireless environments impose a set of constraints for mobile commerce development. The authors have developed alternative mechanisms based on standard technologies in order to improve delivery of information, usability, security, management and service personalization for clients on mobile environment.

KEY WORDS
m-Commerce, broker, mobile agent, end-to-end security

1. INTRODUCTION
The successful implementation of mobile telephony worldwide presents an important opportunity for the expansion of electronic commerce over wireless environments. Mobile commerce (m-commerce) involves three basic aspects: a) business and service delivery in the client neighborhood, b) timely and geo-referenced information for the mobile user and, c) immediate completion of a transaction. The user should have the following facilities: negotiation and immediate delivery, fast methods of micro and macro payment, and usability in the mobile environment. There are, however, several factors that hinder the implementation and development of m-commerce. These drawbacks are related to the characteristics of the wireless environment (normally less bandwidth, lower latency, less stability of connections, less predictable availability) and constraints in the mobile devices (less powerful CPU, less memory, limitations on power consumption, form factor of displays,).

End-to-end security between the Internet server and the mobile terminal is also indispensable for e-commerce applications. WTLS specification does not provide this level of security. The use of WTLS and TLS allows privacy in wireless and wired channels, but the security achieved is not sufficient in e-commerce environments. End-to-end security between mobile user and Internet server is not provided, compatibility between both is not supported. Following the end-to-end arguments in system design [9], the authors propose the implementation of a new secure layer inside Wireless Application Environment (WAE), named WAE-Sec, that makes end-to-end security, TLS compatibility, and user transparency possible, and avoid translation and decompression in the WAP Gateway node.

In [1] the authors noted that only 16% of the handsets available had Internet-ready WAP devices and that the penetration rate may have increased to 70% today as carriers and users upgrade. The study further indicates that users were put off by slow connection speeds of 9.6K bit/s (compared with the 56K bit/s for a dial-up connection) exacerbated by a lack of comfort and ease of use. A similar study was conducted by Nielsen Norman Group [2]. The studies agreed that faster speeds and wider use of data handsets should help boost mobile data and m-commerce.

In wireless networks, there is the need to speed up the response to the user, and make it easy to use in crowded and noisy environments with limited devices form factor. In order to overcome problems related to delivery of information, usability and customer relationship management, WAP mechanisms must be optimized for faster delivery of data and should use client information to manage content delivery and reuse of data in a predictive manner.

Both strategies together, usability and security allow a better connection between client and server. Once server and client are connected, if an m-commerce transaction is to be performed or agent code is to be downloaded, a secure channel could be used to complete it. Section 2 briefly presents the WAP environment, WAP security model and the proposed solution. Section 3 presents a proposal, based on complementary mechanisms that partially solve usability problems. Section 4 presents the performance measurement criteria. Finally, conclusions of the work are presented in section 5.

2. SECURITY IN THE WIRELESS ENVIRONMENT
Many protocol standards have emerged from the Internet world and are widely accepted, they also have their corresponding “lightweight” version for the wireless world in WAP. In the wired Internet, the WAP gateway becomes the termination point; from that point the WAP protocol assumes the role of TCP/IP. The WAP protocol should complement error recovery mechanisms, and be able to tolerate high latency and
low bandwidth using small devices. Fig. 1 shows both stacks of protocols involved in this environment.

Fig. 1. Protocol stacks of Internet and WAP.

Data access in WAP specification 1.2 [4] uses push and pull mechanisms. The push mechanism avoids a round trip of client request and server response and allows data delivery in a predictive fashion using mobile client profile information. Fig. 2 shows a simplified scheme of data delivery using the two mechanisms.

Fig. 2. Mobile Data Access Paradigm

2.1 The WAP Security Model

Usually, the mobile device connects to the Internet through a gateway that translates between Internet and WAP protocols. Fig. 3 shows the typical WAP security model.

Fig. 3. Typical WAP Security Model

It can be observed that secure channels between the client and the WAP gateway are supported by WTLS, while the channel between the WAP gateway and Internet server is supported by TLS. Translation between WTLS and TLS is executed at the WAP gateway. In consequence, security in WAP environments presents the following weaknesses:
1. No end-to-end authentication between Internet Server and Mobile User.
2. Contents of messages are in the clear in the WAP gateway [5].

End-to-end security between the mobile user and the Internet Server is necessary; also is essential the compatibility with TLS servers

2.2 End-to-end security adopted solution

Following end to end principle [9], in order to provide a secure end-to-end TLS tunnel between an Internet server and mobile user, WAE was chosen as a suitable layer for the implementation of a TLS compatible security layer named WAE-Sec. The WAE-Sec location is shown in fig. 4. To improve security, the new layer inside WAE must provide mainly the following features:
1. End-to-end security.

Fig. 4. Protocol translation with WAE-Sec in the TCP/IP-WAP gateway.

In [15] the proposed solution is presented in detail.

2.3 Related work

The work "Wireless extensions for TLS." [8], includes important features like URL referencing, the possibility to hold user certificates on different machine than the mobile equipment, deployment of OCSP (On-line Status Certificate Protocol) also extend support for wireless compatibility at the server side of connection. All of these topics are desirable improvements for wireless clients. WAE-Sec presents the following advantages over WTLS and [8]:
1. Compatible with already operating Server based on TLS.
2. No need of special agreement between Server and telephony operators.
3. No translation at WAP gateway.
4. Many Internet TLS ready and operating portal and servers available.
5. There is not need to extend TLS at server side.

Security of the broker needs the same requirements than any secure server in Internet. Privacy of customer information is another important issue only mentioned but not covered in this paper.
3. USABILITY AND BROKER MECHANISM

The Theory of Constraints (TOC) maintains that every system is subject to at least one “bottleneck” called a constraint, which prevents the system from achieving infinitely high levels of performance. Removing a constraint, significantly improves the performance of the entire system. In the wireless environment, a critical constraint is the wireless link. Other important drawbacks are noise, unpredictable disconnection and cost. Fast delivery of information is also a concern. The wireless channel between the mobile user and the WAP gateway constitutes the bandwidth "bottleneck" for data transmission.

The solution presented in this paper in order to speed up the response to the user is based on a broker. This broker should be located near this bottleneck on the Internet side to buffer, and cache the traffic to the wireless terminals with a distributed structure similar to [6].

One strategy to improve usability is to reduce data traffic by providing user interaction off-line. The location of a broker on the Internet side of the connection, always connected to the Internet, eases the access to web information from mobile terminals, reuses information and sends useful information in a predictive mode to mobile users, which can reduce data traffic in the wireless link and delegate search and information recovery tasks, even if the user is off-line.

The broker combines intelligent and mobile agent technologies, with cache and proxy systems and customers relationship management. An additional user certificate repository is integrated to the above-mentioned mechanisms and is protected by means of a security layer. Fig. 5 shows a general structure of the broker that manages the data content interchange between client and server.

![Fig. 5 Broker general scheme.](image)

Multi agent systems [12] allow a cooperative behavior between agents for the search and retrieval of information on the Internet. Information obtained must be organized inside distributed proxy and cache systems that consider special characteristics of mobile users (latency, noise, unpredictable availability, disconnection) and then reused by other mobile users. Tasks such as searching, advising, contacting, comparing, filtering, and facilitating access to databases are ideal for agent technology use. Agent technology requires the environment to provide the following: accessibility, determinism, predictable static or dynamic environments, and rules of access. Unfortunately the Internet is not friendly enough for intelligent agents.

Mobile agents [10][11] enable client functions to run not only on mobile hosts, but on stationary hosts as well. Furthermore, mobile agents allow clients to download server code to the mobile host for execution, can contain threads and, therefore, be active. They can maintain state information and make intelligent decisions using it. They differ from downloadable applets in that they can roam independently among different machines and are not limited to being downloaded once from server to client. Mobile Agents are formed partly by code and partly by data. Data could be inserted in a decision tree structure that, once downloaded to the mobile device, could allow off-line user interaction, which is very desirable on a weak link, noisy with unpredictable disconnection.

Knowing the user profile allows us to manage user needs of information, and to delivery information in a predictive fashion and dynamic tracking of clusters of information. A user profile could be managed and updated dynamically, processing information from different sources. CRM (customer relationship management) technology for wireless environments improves usability and extends the capacity of mobile terminals. Ease of use and additional information such as geographic proximity to a service center and delivery of information organized by zone should be provided. The information can be processed before being presented to the mobile terminal. Delivery of contents could be done using SMS, e-mail or WML/Script.

WAP Push and Pull delivery mechanisms allow two different strategies for information content delivery. The push mechanism allows automated delivery without explicit request. The user profile makes an anticipated delivery of contents feasible. Content delivery could be done in a predictive fashion profiting from the facilities of the Push mechanism of WAP. The push mode also involves lower occupation of the wireless channel than pull requests (round trip). The pull mechanism based on user request of information could profit from intermediate storage and agent-assisted search of contents to ease and speed up content delivery.

A public key infrastructure is needed to provide authentication and privacy. The broker is suitable for storing digital certificates in a repository. Combined with OCSP (On line Certificate Status Protocol), fast information about the status of certificates could be obtained.

Security is an important concern and its strategy should involve the network, the operating system, databases protection, code interchange and data transmission. Administrator personnel access measures must be similar to that of a Web Secure Server. Once the broker has connected the mobile terminal to the Internet server, a secure channel must be necessary for negotiations and
payment. The broker ends its intervention and an end-to-end security TLS tunnel is recommended.

3.1 Implementation

Multi agent search mechanism located in the broker utilizes Java and Aglet technologies. Fig. 6 illustrates a functional diagram of multi agent system cooperative operation, agents can be replicated (serialized) allowing concurrent search.

(0) The broker, by means of the agent Listener, receives an external request, through socket port 1500.
(1) The agent listener creates a socket connection with the source of request and delegates the work to the agent master.
(2) The master agent analyzes the request and extracts the type. With this information creates the agent database. Once the database agent is created, it asks information of the name of aglet class to be created and the server where to send to do its task.
(3) & (4) The agent database connects with the database and extract information related with aglet and server.
(5) The agent database delivers the requested information to agent master.
(6), (7) & (8) The Agent Master creates the specific aglet (i.e.: AgletTime, AgletSearch, AgletFligths, AgletHotel, AgletTraffic,...), initializes, serializes* and send to server address indicated by agent database.
(9) & (10) The specific aglet arrives to the destination server, where is deserialized and defined again, keeping data initialized by Agent Master. Get and filters the obtained information.
(11) Once information is obtained, the aglet is serialized and returned home.
(12) & (13) At home, it is de-serialized and delivers information to agent master. Then aglet is deactivated.
(14) The Agent Master delivers information and closes the previously opened socket.

Creation, Serialization and De-serialization, … of aglets are done by Tahiti Aglet Server. The messages between agents are done through AgletProxy interchanging messages with ASDK of IBM.

Figure 6. Multiagent search system

4. PERFORMANCE IN WIRELESS ENVIRONMENTS

Performance of complementary mechanisms in wireless environments could be evaluated by the following parameters:

a) Round-trip time of a request.
b) Data traffic over the wireless link.
c) Facilities for the mobile user.
d) Waiting time.

In order to evaluate the improvements introduced by the broker, some simulations of the overall system have been done. Figure 7 shows the simulated scenario. Wireless channels support up to 15 mobile terminals
(simulated by Poisson traffic) for up and downlink information from Internet through a broker located at Internet side of connection, as well as voice calls. Between mobile users and wireless channels there is an overflow queue for data requested packets with no free channel at the arrival time. The broker has its own storage system, which is a database, with URLs and web pages updated with every request. Each new request is compared with the information contained in this database. If the page is stored, it will be immediately sent to the mobile user, otherwise the request is sent to an Information Server in Internet. When the requested page is found, the server sends it back to the broker. The broker adds it to its storage system.

Considerations about the simulated system:

1. Two types of data traffic have been considered: 
   a) Information request packet, with a fixed length of 1,000 bytes. [16] 
   b) Information response packet, that is a page, that follows a Pareto distribution with a mean equals to 31,341 bytes. [17] 
2. Data traffic and voice calls share the wireless channels. The voice calls have preemptive priority over the data traffic: the data packet will be pulled out of the wireless channel if an incoming call has no free channel. The call will pull out first the uplink packets, and finally the downlink ones. The system works this way because is critical to interrupt a downlink packet instead of an uplink one. 
3. Any packet can use one, two or three wireless channels to be transmitted, depending on the number of available free data channels at this moment. 
4. Downlink packets (pages) will have priority over uplink ones. This means that if there is a downlink packet waiting an empty channel for being transmitted, no uplink packet will be transmitted before it. 
5. Packet transmission time follows an exponential distribution. 
6. An extra time is added to the total time expended in the Internet for the packet, due to all the networks and logical nodes that the packet had gone through. This is the ping time, that follows an exponential distribution with mean 1,259 seconds [18]. 
7. There is the possibility of losing packets because of timeout. The timeout value can be fixed. An appropriate choice could be 150 seconds; in this way a large number of packets can be received. 

There were tested 3 different simulated scenarios:

1. Voice Users: Mainly users that use WAP devices for voice calls. 
2. Data Users: Users that mainly transmit data 
3. Voice and Data Users: average users with 60% of voice traffic and 40% data traffic. 

First of all, the simulation has been executed without the broker, in order to compare the simulation results with [2]. Timeout has been fixed at 150 seconds, the arrival data traffic at 2 arrivals per second and 0.3 voice arrivals per second. The mean packet delay obtained has been 67.2 seconds. The bottleneck is the downlink way, with a large number of packets waiting for being transmitted. The results are similar to [2].

The first scenario, with only voice users, will be useful for testing the correct working of the system related with lost call probability. In order to have a lost call probability lower than 0.02 the incoming call rate is 0.32 with a service time of 30 seconds per call.

The second one, with only data traffic, allows us to calculate the number of arrivals per second to avoid a large number of packets in the overflow queue. With 150 seconds of timeout and one arrival per second the system works correctly. Obviously, the mean packet delay for packets served directly by the broker has decreased a lot (10.7 seconds), whereas for the rest of the packets the mean delay were similar to the initial situation (61.2 sec).

The third scenario is the closest one to reality. For simulating, the 60% were incoming calls, and the 40% data traffic. In 1,000 seconds and 0.3 voice arrivals per second, the systems generated 314 voice calls, so 209 data packets were generated too, that means 0.209 arrivals per second. The simulation results with these inputs were 8 seconds of mean delay for packets coming from the broker, and 53 seconds of mean delay for packets coming from Internet. The results are better than the last scenario because of the incoming data traffic is not as heavy as the other one.

Using the same inputs with the system without the broker, the results obtained showed that the system works better with the broker in general. Simulation results in all cases show a very important improvement in the system performance. As is obvious, these improvements depend on the probability that the information requested could be found in the database of the broker. This improvement is not only due to the fact that it is not necessary to do a search on the Internet, but also to the fact of immediately releasing the channel. All this facts implies that the bottleneck (wireless channel) is not so critical with the broker, because a large number of packets can be served. Timeout influence in the packets coming directly from the broker is less important than in the packets coming from the Internet.
5. CONCLUSIONS

With advances in wireless data telecommunications and portable computers, nomadic users will soon enjoy virtually unlimited access to information and services anytime and anywhere. There are, however, obstacles and limitations inherent in the wireless environment.

End-to-end security compatible with TLS without WAP gateway intervention improves security compared with WTLS. The solution presented establishes end to end compatibility with TLS and locates the layer inside WAE where it is able to control the way data is transmitted to avoid translation at WAP gateway. WAE is a suitable layer for flexible inclusion of new cryptographic libraries and algorithms as a part of services and format options.

The need to improve delivery of information mechanisms and availability of different technologies make it feasible to deliver information to WAP users in a better way. Fast delivery of data contents and customer relationship management are key elements for WAP success in wireless telephony environment. These tasks, and others like CRM can be achieved by means of a broker based solution.

6. ACKNOWLEDGEMENT

This project has been developed as part of the project ACIMUT CICYT TIC2000-1120-C03-03.

REFERENCES

[10] Hardee M., Why wireless needs javatm technology, 2000 JavaOne SM Developer Conference,