An Authentication Protocol for Mobile Devices

Paulo Simões
paulo.simoes@link.pt
IST/Link

Pedro Alves
pedro.alves@link.pt
IST/Link

José Rogado
jose.rogado@link.pt
Link

Paulo Ferreira
paulo.ferreira@inesc.pt
IST/INESC

Abstract

Currently, most of e-commerce applications rely on asymmetric cryptography to guarantee the authentication of the involved parties. On the other hand, a growing demand for mobile devices has geared a shift towards mobile e-commerce applications. This paper argues that the existing authentication protocols, based on asymmetric cryptography, are not appropriate for such devices due to their limitations in computing power, memory capacity and cryptographic support.

We have designed and implemented an efficient protocol for resource-constrained platforms that achieve a level of security similar to the one achieved by the protocols in use today. This protocol is based solely on symmetric cryptography and our implementation proves that the performance achieved is good.

1 Introduction

The extraordinary development of communication technologies and the widespread use of the Internet have contributed to the growth and maturing of e-commerce. On the other hand, we have seen a growing demand for mobile devices. This search for smaller, cheaper and faster platforms has led to the appearance of PDAs, cellular phones and pagers. Therefore, although the PC platform has been the dominant target for client Internet applications, we are able to predict a migration of e-commerce applications from the traditional desktop to these mobile devices. For example, in a near future, one might think of buying/selling stock shares through a mobile phone or browsing pay-per-view news on a PDA, while waiting on the bus stop.

However, being the Internet an open and inherently insecure network, some concern has been raised in transmitting sensitive information. The solution lies in using cryptography and secure authentication protocols that guarantee the confidentiality, authentication and integrity of communications. Such protocols, like SSL [Netscape96] and SET [SET99], already exist and are widely used in current e-commerce applications. Nevertheless, most of them are based in public-key cryptography, which may be infeasible in resource-constrained environments. In fact, most mobile devices have a slow processor, small memory and lack of support for cryptography, making it harder to implement such protocols without incurring in intolerable latencies.

We devised a protocol based solely on symmetric cryptography that performs well in resource-constrained platforms and maintains the high security level that one can achieve with the protocols in use today. We also solved the main problem of symmetric-cryptography

protocols, the key distribution, splitting the protocol in two halves: the acquisition of the shared key and the authentication itself. This protocol was successfully implemented in a HomeBanking application for PalmPilot devices, achieving reasonable performance and usability.

The paper is organized as follows. In section 2, we describe the main aspects of our protocol. In section 3 we provide a description of the implementation of the protocol in an application. Finally, in section 4, we present our conclusions.

2 Architecture

The authentication protocol must be able to create a secure channel between two principals on top of an insecure network, like the Internet. It's not difficult to eavesdrop a line or to compromise a router and be able to listen/alter all messages in transit [CERT95]. In order to prevent this, the protocol must ensure the mutual authentication of both parties and the confidentiality and integrity of all data transmitted through it. Such protocols already exist and have gone through deep analysis, like SSL and TLS [TLS97]. However, they rely heavily on asymmetric cryptography, which causes some concern about their performance on resource-constrained small devices. In fact, we came out with some performance measurements for cryptographic functions on one of these devices, the PalmIII[1] from 3Com:

Algorithm	*Key length (bits)*	*Kb/sec*
DES	56	5.07
IDEA	128	4.07
Blowfish	128	6.85
MD5	--	58.74
SHA-1	--	29.7

Algorithm	*Key length (bits)*	*Time*
RSA key generation	512	~2m45s
RSA key generation(a)	1024	~20 minutes
RSA sig. generation	512	~5 sec
RSA sig. verification	512	640 ms

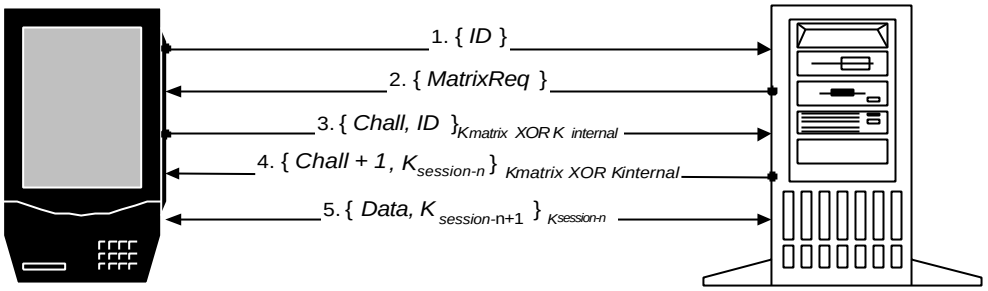
Figure 1 – Performance measurements for some cryptographic algorithms ((a) was extracted form [Daswani98]).

As we can see, generating RSA keys on the PalmPilot is prohibitively expensive. Moreover, an RSA signature generation is also very slow, which makes a protocol like SSL to become unusable. On the other hand, symmetric-key and digest algorithms perform well, which lead us to a simple conclusion: the protocol must be based solely on symmetric cryptography and digest algorithms. Furthermore, we want it to be simple, yet highly secure.

One of the classic problems in authentication based on a shared key is the distribution of that key [Tanenbaum96]: we need a secure channel to distribute the key, which will, in turn, create a secure channel! We solved this problem in a very simple but effective way, by using the client's PC. In fact, the protocol needs an internal key, which should be acquired via a browser on the client's PC and then installed on the device. The authentication can be made using SSL or other similar protocol. Notice that the internal key travels from the server into the device through a secure channel, using the client's PC as a proxy. We assume that the channel between the PC and

[1] The PalmIII device has a 16.67Mhz processor and 2 Mb of memory capacity.

the device is secure, because it is made under the control of the device owner. We'll now present our authentication protocol, assuming that the internal key was already acquired as described above and is safely stored in the PDA:



{ *Message* }$_K$ ::= *Encrypt_with_K*(**Message** + *Digest*(**Message**))

Figure 2 – The authentication protocol over the Internet.

The idea behind this protocol is simple: in step 1, the PDA starts the protocol by sending its ID (e.g. serial number) to the server. In step 2, the server creates a session for this client and generates a random request, which we called MatrixReq. This is basically a line/column pair like "line 1/column B", to which the client should respond with the appropriate key[2]. This key is called matrix key.

In step 3, the PDA generates a challenge and sends it along with its ID to the server, encrypted with a combination of the matrix key and the internal key. The server decrypts the message and verifies if this ID matches the ID sent in message 1. This authenticates the client.

In step 4, the server sends the challenge received in the previous message plus one and a randomly generated session key. The PDA then decrypts this message and verifies the challenge. If it matches the one that was sent in message 3, then the PDA can trust that it's indeed talking to the right server.

From now on, a secure channel has been created and all data is encrypted with a session key. Notice that we setup a new key for each message to prevent replay attacks.

3 Implementation

The proposed protocol was successfully implemented in a HomeBanking application for PalmPilot devices. With the growing diversity of mobile devices to which the application could be targeted for, its portability was a major concern since the beginning. Therefore it was developed using the language Java, whose features perfectly meet this requirement.

In order to be able to execute Java code on the PalmPilot a virtual machine was necessary. We choose the Ghost, an implementation of the Java VM for the Pilot. However, it lacks several Java features which imposed the use of the C Language for some of the application modules. Among these is the cryptographic module that had to be developed using the *PilotSSLeay* package which is an implementation of the well known *SSLeay* package for the PalmPilot.

[2] This key is obtained consulting the matrix card, which is unique to each user.

To achieve the high security level required, the *Blowfish* (with 128-bit keys) and *MD5* algorithms were used for the encryption and digest calculation of the messages exchanged during the application.

However, being security the main concern and due to the limitations of the PalmPilot, some other measures were taken. The most important one was the protection of the internal key inside the device. Since no memory protection is available on the Pilot, an unauthorized access to the key could be a serious flaw in the application. To avoid this, the Pilot password is used to encrypt the internal key when the application is executed for the first time. This way, a physical user authentication against the device can be provided since the password must be entered every time the application is executed.

On the other hand, the identification of the device against the server is made using its internal serial number as the *ID* sent to the server during the protocol.

Despite the limitations of the device, a reasonable performance can be achieved. The exchange of a request-response message pair takes on average 18 seconds, including not only the security related processing, but also the communication latency and some additional operations done in the Pilot and in the server.

This way, using the authentication protocol presented here, the developed HomeBanking application can securely provide to the user several banking operations like balance inquiry, service payment or check book request.

4 Conclusions

Our work shows that it is possible to implement e-commerce applications in resource-constrained mobile devices with reasonable performance. However, protocols based on asymmetric cryptography, which are widely used in e-commerce nowadays, cannot be directly used in such devices. Instead, we must rely on symmetric cryptography.

This paper addressed the design of a protocol based on symmetric cryptography. Furthermore, we've described an implementation for a particular mobile device, the PalmPilot from 3Com. A detailed description of this application can be found in [TFC99].

The lack of a standard authentication protocol targeted to small mobile devices can compromise the emerging market of mobile e-commerce. We hope our work to be a big contribution to the development and widespread acceptance of such technology.

5 References

[CERT95] CERT Advisory CA-95.01. - IP Spoofing Attacks and Hijacked Terminal Connections

[Daswani98] Neil Daswany, Dan Boneh. Experimenting with Electronic Commerce on the PalmPilot. Stanford University, 1998.

[Netscape96] The SSL Protocol Version 3.0. Netscape Communications, 1996.

[SET99] The SET Standard Specification. http://www.setco.org/set_specifications.html. 1999.

[Tanenbaum96] Andrew Tannenbaum. Computer Networks. Prentice Hall, 1996.

[TFC99] Pedro Alves, Paulo Simoes. Course final project – "Aplicação de HomeBanking para PalmPilot". IST 1999. (in portuguese).

[TLS97] Transport Layer Security Working Group. The TLS Protocol (Internet-Draft). 1997.