End to End Cryptography in Cellular Phone Applications

Geraldo Lino de Campos[1] and Silvia Paladino de Alcântara[2]

[1]Instituto de Pesquisas Tecnológicas, São Paulo, Brazil, geraldo@decampos.net,
[2]Nokia Corporation, São Paulo, Brazil, silvia.paladino@nokia.com

Abstract. This paper presents some considerations about the usage of cellular phones in commercial transactions of lesser value. Security is a major concern, and it is shown that even low end cellular phones have enough capacity to process the usual cryptography algorithms, enabling the use of end to end cryptography in commercial transactions. Two case studies of real applications are presented: a small payments system and a general payment system.

Keywords: Cellular phones, cryptography.

1 Introduction

Cellular phones are the most ubiquitous digital appliances, with about 3.2 billion terminals. As such, it is the most powerful tool for digital inclusion. Increasing its usefulness and the number of available applications will add value and extend the penetration of the cellular phone in increasing strata of the world's population.

The transaction of business via WEB is becoming an imperative for most organizations. Most business transactions are conducted via personal computers. However, there are situations when the potential customer cannot afford it, or simply is away from home or office – but the 3.2 billion cellular phones are available whenever the customer is ready for an operation. There are many opportunities to use cellular phones in business operations.

This paper examines the application of cellular phones to commercial transactions of lesser value. Many people that can not afford a credit card have cellular phones, and many payments are too small to justify the credit card's processing fees. The possibility of using the cellular phone for these payments is attractive both to the consumer and to the merchant.

The purpose of the present research is to investigate the feasibility of simple, inexpensive and secure applications using normal cellular phones for those small payments. The operation must be simple, fast and accessible for anyone.

The idea is not to load special programs on the user's cell phone, which can be too complex for laymen, but only on the merchant's phone. The payment process may, and generally will, involve the user's phone in its plain native state.

Since the realization of any commercial operation raises concerns about security, part 2 examines the security of cellular communications for commercial operations, concluding that they are generally insecure.

Part 3 studies the utilization of end-to end cryptography in cellular phone applications, with emphasis on the execution time of the main cryptographic algorithms.

XXXIV Conferencia Latinoamericana de Informática (CLEI 2008), pp. 853-859, 2008.

Part 4 presents two case studies of real applications using the conclusions of the previous parts.

2 Security of cellular communications

Nowadays there are essentially two competing standards for cellular phones: CDMA and GSM. Usage of CDMA systems is declining; today's networks are predominantly GSM (86% of world cellular base in sep 2007 [1]); nevertheless its security deserves examination.

CDMA air interface is not inherently secure – all information is transmitted in open form. In practice, it is considered secure due to use of spread spectrum technology and the use of Walsh codes. This makes the signal of any individual call difficult to distinguish and decode. The determined intruder, however, can obtain specific equipment, of the same nature used by the telecom operators, and have access to individual calls. Since this kind of equipment is fairly expensive, it is unlikely that someone will bear this expense to fraud operations of small value.

Contrary to general belief, GSM air interface is now insecure. The information is transmitted under cryptography, using some version of the A5 algorithm, but the algorithm was gradually broken during the 1997-2006 period [2, 3, 4, 5]. A detailed discussion of this topic is beyond the scope of this paper. It suffices to quote two paragraphs [5]:

> "We first describe a ciphertext-only attack on A5/2 that requires a few dozen milliseconds of encrypted off-the-air cellular conversation and finds the correct key in less than a second on a personal computer."

and

> "Although GPRS uses a different set of encryption algorithms, the key for GPRS is generated using the same A3A8 algorithm using the same Ki but with a different RAND called GPRS-RAND. Therefore, an attacker can use a fake base station to initiate a (non-GPRS) conversation with the mobile using A5/2, and send the GPRS-RAND instead of RAND. Thus, the resulting key is identical to the key that is used in GPRS, and the attacker can recover it using the attack on A5/2."

The first conclusion is that the air segment is intrinsically insecure in both systems.

Besides that, the user has no guarantee that the information is under cryptography inside the operator systems. This is an option not used by all operators. Data can be obtained by the operator's employees and used in a fraudulent way.

The final conclusion is that the cellular phone systems are insecure, and operations involving money require additional security.

The first idea is to use standard solutions, like IPSEC. Some operators offer a service of VPN (W-VPN - Wireless Virtual Private Network) from their premises to customer premises, but this does not solve the security issues of the air segment or the inside the operator's system, so it won't be considered further.

End-to-end IPSEC is a solution for security. But it has several drawbacks, especially for prototypes or proofs of concept of new applications:

- It requires changes and additions in the corporate infrastructure, generating cost and delays – usually, there is a large backlog of applications and changes with higher priority;
- New applications using the corporate backbones, firewalls and other resources common to mainline applications require lengthy and complex homologation procedures, precluding the realization of proofs of concept;
- The software for IPSEC on cellular phones is expensive, and is available only on high end phones, adding to cost.

The same end-to-end security can be achieved if the cryptographic procedures reside in the application (both on the cellular phone and on the server); demonstrations and proofs of concept don't affect the corporate infrastructure. The possibility of using the installed base of cellular phones is a great plus.

The feasibility of this solution depends on the verification that usual cellular phones have the necessary resources for running the cryptographic algorithms.

3 Cryptographic algorithms in cellular phones

Several phones, ranging from the basic to the most sophisticated models, were selected to verify their suitability for processing cryptographic algorithms. All the models were selected from the same brand due to availability, but there is no reason to suspect that the results would be substantially different for other brands, since the vast majority of cellular phones are built around commodity microprocessors.

The test was conducted with the main cryptographic algorithms: TDES and AES for encryption, and PBKDF2 for protecting the key. Discussion and analysis of the algorithms are beyond the scope of this paper, and can be found on regular textbooks on computer security [6] or in reference documents [7, 8, 9].

The test for TDES and AES consisted in encrypting a 300 characters block. This size was selected considering that very few, if any, applications will require more than that. Anyway, entering that much information into the small keyboard of a cellular phone will not be a practical task.

The implementation used the open source library Bouncy Castle [10]. Since the library is open source, there are no initial costs or royalties for deploying the programs even in a large base of cellular phones. The test included the three implementations of the AES algorithm: standard, speed optimized and size optimized.

Table 1 presents the main characteristics of the phones, and the time to perform de encryption tasks.

The important conclusion is that even the low end models have sufficient processing power to execute the main algorithms. The authentication algorithm PBKDF2 may be quite slow on these models, but since it is executed only once each time the phone is powered on, it is not an impeditive factor. On mid and high end models the execution time is very small for all algorithms.

Table 1. Cell phone models and execution time for the relevant cryptographic algorithms.

Model	Type	Operating system	Memory size		TDES (ms)	AES standard (ms)	AES speed optimized (ms)	AES size optimized (ms)	PBKDF2 (seg)
			Heap	JAR					
3100	Low end	Nokia OS	205 KB	63 KB	278	110	62	146	51
6070	Low end	Nokia OS	512 KB	128 KB	262	98	59	134	48
6101	Mid range	Nokia OS	500 KB	166 KB	261	98	61	137	49
6822	Mid range	Nokia OS	500 KB	147 KB	283	109	63	143	51
5200	Mid range	Nokia OS	1 MB	2 MB	3,4	2,5	1,4	3,4	0,7
6600	Mid range	Symbian 7.0	3 MB	Unlimited	4,4	3,7	2,5	4,3	1,9
6670	Mid range	Symbian 7.0	Unlimited	Unlimited	4,1	2,9	2,2	4	1,5
E6*	High end	Symbian 9.1	Unlimited	Unlimited	2,5	5,4	4,5	6,4	3,2
5700	High end	Symbian 9.2	Unlimited	Unlimited	1,5	3,2	2,6	3,7	1,9
9300	High end	Symbian 7.0	Unlimited	Unlimited	3,2	2,6	2.0	2.8	1,0

*Models E61, E62 and E61i have aproximately the same performance

4 Case studies

This section presents two case studies. The first is an application already in successful operation, and the second awaiting some commercial details for full deployment.

4.1 Case study 1: virtual food stamps

Not really food stamps. Corporations must have a restaurant for the workers, or provide a voucher with the value of a basic meal for each workday. The voucher can be redeemed at accredited restaurants. Vouchers are emitted by specialized corporations, and sold to employers.

The value of each voucher is relatively small, so processing costs are a relevant factor of concern. Most vouchers are physical, but there are several virtual ones, usually based on magnetic cards. Their processing requires specialized equipment on the restaurants.

Many restaurants are small operations and the cost of ownership of additional equipment can be significant.

The system in this application was developed for a voucher operator in a medium-sized city (population: approximately 320.000). The cellular phones implement an intelligent Point-Of-Sale terminal. For using the system, each employee receives a monthly amount to be expended in meals.

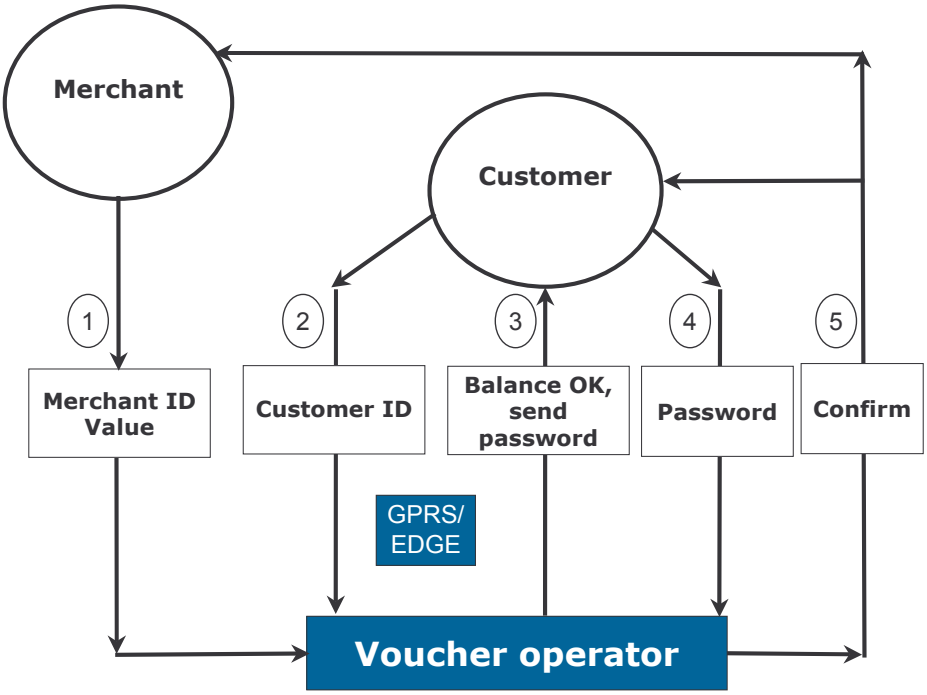

Fig. 1. Simple POS application.

The application is very simple, implemented in J2ME, and run only on the merchant's cellular phone. The operation, depicted on fig 1, is:

1) The merchant calls the voucher operator, sending the bill value (Merchant ID is automatically sent by the application).

2) in the same phone, the customer enters his ID;

3) the voucher operator checks the balance, and if OK asks for the customer password;

4) the customer enters the password on the merchant's cellular phone;

5) the voucher operator moves the amount to the merchant's account and confirms the operation.

Current status: fully operational.

4.2 Case study 2: general solution for small payments

This application was developed for a large bank that intends to deploy it nationally.

This application requires a program running on the merchant's cellular phone, using GPRS (merchant phone) and SMS (customer phone) to communicate with the bank. Cryptography is used in all GPRS communications. The customer must have a working cellular phone, without any software or special requirements.

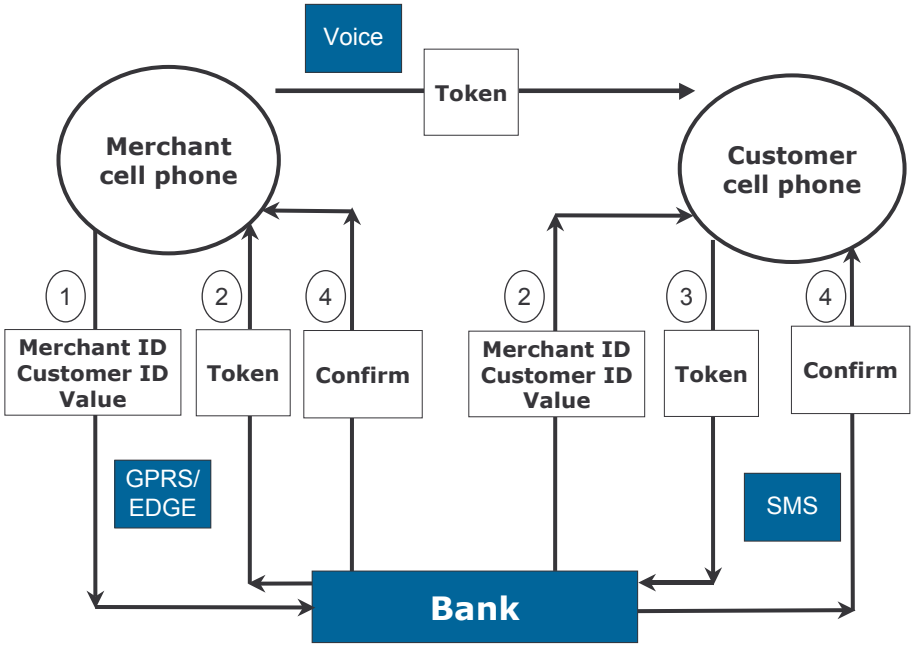

Fig. 2. Generic payment system.

1) The merchant enters the customer's ID (most likely the phone number) and the transaction value. Its ID is part of the program, and is sent automatically.

2) The bank sends a token (a small number) to the merchant's phone, and an SMS message to the customer, containing the merchant's ID and the transaction's value.

3) The merchant reads the token to the customer, who uses it as the answer to the SMS message. Although not essential, this step was added as an additional security measure.

4) The bank sends confirmations to the merchant and to the customer.

Current status: Approved by the bank's business and security units. A business question between the bank and the phone operators about whom and how the SMSs will be paid is delaying deployment.

5 Conclusion

The research was conducted to verify the feasibility of using cellular phones for simple, inexpensive and secure applications on the small payments segment of business. The conclusions are that it is possible, security end-to-end can be obtained even in the low end cellular phones, and no costs or royalties are incurred by the use of available open source cryptographic code.

Two case studies show that practical applications can be done with cost competitive with other solutions.

References

1. World Wireless Market – Subscriptions by technology (Sept 07), http://www.3gamericas.org/English/Statistics/q32007_1.cfm
2. Jovan Dj. Golic, Cryptanalysis of Alleged A5 Stream Cipher, EuroCrypt 1997, pp239–255 (1997)
3. Elad Barkan, Eli Biham and Nathan Keller, Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication, Crypto 2003, pp600–616 (2003)
4. Elad Barkan, Eli Biham, Conditional Estimators: An Effective Attack on A5/1, Selected Areas in Cryptography 2005, pp1–19. (2005)
5. Elad Barkan, Eli Biham, and Nathan Keller, Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication, Technion Technical Report CS-2006-07, http://www.cs.technion.ac.il/users/wwwb/cgi-bin/tr-get.cgi/2006/CS/CS-2006-07.pdf
6. Stallings, W – Cryptography and Network Security, Pearson, (2006)
7. Internet engineering task force, PKCS #5: Password-Based Cryptography Specification, Version 2.0, http://tools.ietf.org/html/rfc2898.
8. Federal Information Processing Standards Publication 197, November 26, 2001, ADVANCED ENCRYPTION STANDARD (AES), http://www.csrc.nist.gov/publications/fips/fips197/fips-197.pdf
9. Federal Information Processing Standards Publication 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, http://csrc.nist.gov/publications/nistpubs/800-67/SP800-67.pdf
10. Available on http://www.bouncycastle.org