Achieving Identity-Based Cryptography in a Personal Digital Assistant Device

L. Martínez-Ramos*[1], L. López-García[2], F. Rodríguez-Henríquez[3]

[2] Centro Universitario Zumpango
Universidad Autónoma del Estado de México
Camino viejo a Jilotzingo
continuación calle Rayón
Valle Hermoso
Zumpango, México, C.P. 55600
*mlopez@computacion.cs.cinvestav.mx
[1,2,3] Computer Department, CINVESTAV-IPN
San Pedro Zacatenco, México, D.F

ABSTRACT

Continuous technological advances have allowed that mobile devices, such as Personal Digital Assistants (PDAs), can execute sophisticated applications that more often than not must be equipped with a layer of security that should include the confidentiality and the authentication services within its repertory. Nevertheless, when compared against front-end computing devices, most PDAs are still seen as constrained devices with limited processing and storage capabilities.

In order to achieve Identity-Based Cryptography (IBC), which was an open problem proposed by Adi Shamir in 1984, Boneh and Franklin presented in Crypto 2001, a solution that uses bilinear pairings as its main building block. Since then, IBC has become an active area of investigation where many efficient IBC security protocols are proposed year after year. In this paper, we present a cryptographic application that allows the secure exchange of documents from a Personal Digital Assistant (PDA) that is wirelessly connected to other nodes. The architecture of our application is inspired by the traditional PGP (Pretty Good Privacy) email security protocol. Our application achieves identity-based authentication and confidentiality functionalities at the 80-bit security level through the usage of a cryptographic library that was coded in C++. Our library can perform basic primitives such as bilinear pairings defined over the binary field $\mathbb{F}_{2^{233}}$ and the ternary field $\mathbb{F}_{3^{97}}$, as well as other required primitives known as *map-to-point* hash functions. We report the timings achieved by our application and we show that they compare well against other similar works published in the open literature.

Keywords: Identity-based identity, bilinear pairings, PGP, mobile devices.

RESUMEN

Los incesantes avances tecnológicos han permitido que los dispositivos móviles, tales como los Asistentes Digitales Personales (PDAs), puedan ejecutar aplicaciones sofisticadas que usualmente no realizaban, para lo cual deben estar equipadas con una capa de seguridad que debe incluir en su repertorio los servicios de confidencialidad y autenticación. Sin embargo, este tipo dispositivos comparados con servidores, aún exhiben limitaciones en procesamiento y almacenamiento.

Con la finalidad de hacer realidad la criptografía basada en la identidad (*IBC* por sus siglas en inglés), la cual fue considerada un problema abierto propuesto por Adi Shamir en 1984, Boneh y Franklin presentaron en Crypto 2001, una solución que usa emparejamientos bilineales como su bloque principal. Desde entonces, IBC ha sido un área activa de investigación en la cual muchos protocolos eficientes de seguridad son propuestos año tras año. En este articulo, presentamos una aplicación criptográfica que permite el intercambio seguro de documentos de un Asistente Digital Personal (PDA), el cual es conectado de forma inalámbrica con otros dispositivos. La arquitectura de nuestra aplicación está inspirada en el tradicional protocolo de seguridad de correo electrónico PGP (Pretty Good Privacy). Nuestra aplicación alcanza las funcionalidades de autenticación y cifrado, ambos basados en la identidad con un nivel de seguridad de 80 bits, a través del uso de una biblioteca criptográfica que fue codificada en C++. Nuestra biblioteca implementa primitivas básicas tales como emparejamientos bilineales definidos sobre los campos binarios $\mathbb{F}_{2^{233}}$ y el ternario $\mathbb{F}_{3^{97}}$, así como otras primitivas requeridas, conocidas como funciones picadillo *map-to-point*. Presentamos también los reportes de los tiempos obtenidos por nuestra aplicación y mostramos una comparación de los mismos contra otros resultados publicados en la literatura abierta.

1. Introduction

In the last few years we have witnessed the exponential growth of mobile computing. Nowadays, it is possible to receive/send multimedia information in mobile devices connected through wireless LANs. Mobile applications rank from weather predictors to financial transactions, mobile e-commerce and so on. However, typical computational constraints of mobile devices still constitute a challenge to proven security solutions already in place in the wired Internet scenario. Therefore, most of the time, designers are forced to use highly efficient protocols striving to keep the computational complexity of security algorithms as low as possible, looking for light cryptographic schemes in terms of both execution time and footprint size.

The security model of *symmetric* or *secret-key cryptosystems* is based on the notion that only authorized communication parties have access to certain secret information, known as *secret key*. The security model of this paradigm works under the assumption that any eavesdropper that manages to obtain a ciphertext while in transit will not be able to recover the corresponding plaintext due to the fact that he/she will still ignore the exact value of the secret key. Symmetric encryption/decryption can be performed with extremely high efficiency. However, this paradigm has two major drawbacks, namely, its inherent key distribution problem and the fact that key management when working in a large community of users, tends to be cumbersome. These two issues can be solved by means of the *public-key cryptography* paradigm that was first proposed by Diffie and Hellman in 1976. The Diffie-Hellman protocol allows two parties to agree on a shared secret key, even though they are under the restriction of exchanging messages in public. Shortly after, Rivest, Shamir, and Adleman proposed the RSA cryptosystem in 1977. Today, RSA is one of the most widely known public-key systems.

In the public-key model, each party is assigned with a pair of keys, one secret and the other one public, and therefore, the encryption/decryption process is not symmetric anymore. Public-key encryption permits the definition of the digital signature concept as well as a stronger notion of authentication. Nevertheless, it remains as an open problem to establish a binding between the public key, a digital object, and its legitimate owner. In practice, this issue has been solved by means of the so-called Public-Key Infrastructure (PKI), which defines trusted certification authorities that are in charge of issuing digital certificates. A digital certificate allows uniquely determining the ownership of a given public key.

Since the work by Boneh and Franklin in 2001 [1], we know how to implement Identity-Based Cryptography (IBC) efficiently using a mathematical tool called *bilinear pairings*. In the case of IBC, the concepts of certification authorities and digital certificates disappear. Instead, we can define user's public key to be any user's public information such as his/her email address or any social number identification. Hence, in principle we eliminate both the expensive key exchange protocols required in symmetric cryptography as well as the costly verifications needed under the PKI scheme.

PGP (Pretty Good Privacy) is an email security standard that combines secret-key and public-key cryptography to offer the confidentiality service efficiently. PGP uses a *web of trust* in order to provide an authentication service based on good reputation among the system's users. However, this mechanism is not well-suited for e-commerce applications where such trust model simply cannot be applied, among other reasons, because of legal issues.

Our Contribution: In this paper we present a PGP-based application for a PDA device, that is able to offer the confidentiality and authentication services using Identity-Based Cryptography. Our application allows achieving a reasonable level of 80-bit of security without resorting to the costly verifications of digital PKI certificates or the unsecure PGP scheme known as web of trust. In order to do this, we develop a C++ library that allows us to implement the basic operations required by identity-based secure protocols.

The rest of this paper is organized as follows. We discuss in Section 2 basic cryptographic concepts of bilinear pairings and the Identity-based Cryptography. Section 3 presents the identity-based encryption/decryption and signature/verification algorithms used in this work. Then, in Section 4 we describe our mobile application for secure exchange of documents: identity-based PGP. Section 5 reports the timings

achieved by our tool as well as a comparison with other related works previously published in the literature. Finally, in Section 6, some concluding remarks are drawn.

2. Basic concepts

A finite field is a set having finitely many elements in which the usual arithmetic operations (addition, subtraction, multiplication, division by nonzero elements) are well defined. Moreover, all usual algebraic laws, namely, commutative, associative and distributive laws hold. The order of a finite field is defined as the number of elements q that it contains. Such a finite field exists for every prime p and positive integer n, and contains a subfield having p elements. This subfield is called the ground field of the original field. For the rest of this work, we will consider: $q = p^m$, with $p = 2, 3$.

A polynomial in is irreducible if is not a unit element and if then or must be a unit, that is, a constant polynomial. Let be an irreducible polynomial over of degree , and let be a root of , i.e., . Then, we can use to construct a finite field with exactly elements, where itself is one of those elements. Furthermore, the set forms a basis for , and is called the polynomial (canonical) basis of the field.

2.1 Bilinear Pairings

The equation of a supersingular elliptic curve defined over the underlying field $\mathbb{F}_{p^m}$, where p is a prime and m is a positive integer, is given as,

$$y^2 + cy = x^3 + ax + b \quad (1)$$

where the coefficients a, b and c belong to $\mathbb{F}_{p^m}$.

Let $E(\mathbb{F}_{p^m})$ denote the group of all points (x, y) that satisfy Equation 1, with $x, y \in \mathbb{F}_{p^m}$. It is known that the set $E(\mathbb{F}_{p^m})$ together with the point $\mathcal{O}$, form an abelian additive group. The elliptic curve scalar multiplication is the operation that computes the multiple $Q = dP$, defined as the point resulting of adding $P + P + ... + P$, d times.

Let n be a positive integer such that $gcd(p, n) = 1$ and $\mathbb{G}_1 = E(\mathbb{F}_{p^m}[n]$ denote the set of points $P \in E(\mathbb{F}_{p^m})$, with order n, i.e., $nP = \mathcal{O}$.

The embedding degree k, is defined as the least positive integer such that $n|(p^{km} - 1)$, holds. Let P be a point in $\mathbb{G}_1 = E(\mathbb{F}_{p^m})[n]$ and let us define $\mathbb{G}_2$ as the multiplicative group $\mathbb{G}_2 = \mathbb{F}_{p^{km}}$, with identity 1 and order n.

Then, a bilinear pairing on $(\mathbb{G}_1, \mathbb{G}_2)$ is a mapping $\hat{e}(\mathbb{G}_1 \times \mathbb{G}_1) \longrightarrow \mathbb{G}_2$ satisfying the following properties:

1. Bilinearity: $\forall\, R, S, T \in \mathbb{G}_1$
a. $\hat{e}(S + R, T) = \hat{e}(S, T) \cdot \hat{e}(R, T)$
b. $\hat{e}(S, R + T) = \hat{e}(S, R) \cdot \hat{e}(S, T)$
2. Non-degeneracy: $\hat{e}(P, P) \neq 1$
3. Computability: $\hat{e}$ can be efficiently computed

The bilinear property implies the following: Let $a, b \in Z_n^*$

$$\hat{e}(aP, bP) = \hat{e}(bP, aP) = \hat{e}(abP, P) = \hat{e}(P, abP) = \hat{e}(P, P)^{ab}$$

In this work, we use the bilinear pairing called η_T [2]. To implement it, we utilize fields of characteristic two $\mathbb{F}_{2^{233}}$ and characteristic three $\mathbb{F}_{3^{97}}$. Our library computes all the basic arithmetic operations, namely, addition, squaring, multiplication, modular reduction, square/cubic root and inversion in an efficient way using tower fields.

2.2 Identity-Based Cryptography (IBC)

The theoretical concept of Identity-Based Cryptography (IBC) was originally proposed by Shamir in 1984 [3]. The IBC is considered part of public key cryptography, but with a special feature: user's public key can be any user's non-confidential public information that can identify a given user in a unique way. On the other hand, the private key is generated by a special entity called Private Key Generator (PKG), using its master private key and the user's public information. As a result, the IBC reduces the complexity for establishing and managing the public key authentication framework known as Public Key Infrastructure (PKI) previously alluded [4].

As we mentioned above, the PKG is a special entity that generates all user's private keys, which means that the PKG has a dangerous and privileged position. In general, the overall system's security depends on several considerations like the security of the underlying cryptographic functions, the secrecy of the sensible information stored by the PKG, the user's authentication performed by the PKG to create the private key, and the user's precaution to prevent the loss, duplication or usurpation of sensible information [3].

Figure 1 shows the interaction among the aforementioned entities. The user's public key is $k_e = i$, and the private key is derived from i. It is necessary a single interaction between a given user and the PKG. When the public operation is used to encrypt, the scheme is called Identity-Based Encryption (IBE), otherwise, it is named Identity-Based Signature (IBS).

The IBC can be divided into two branches: pairing-based schemes and factoring-based schemes. In this work, we focus on the first of them. In 2001, Boneh & Franklin proposed an identity based encryption using pairings [1], additionally, Florian Hess in 2003 [5] presented an identity-based signature scheme based on bilinear pairings. However, the pairing implementation is complicated and the calculation of the bilinear pairing needs to be completely efficient such that the benefit of using schemes based on pairings can improve any other public cryptographic scheme. In order to obtain the best solution to solve this problem, researchers have developed efficient implementations of bilinear pairings; the interested reader can see [6, 7, 8, 9, 10, 11].

3. Identity-based cryptography using bilinear pairings

3.1 Identity-Based Encryption

In an identity-based cryptosystem, the encryption key is any unique user's identification $k_e = i$ and the decryption key k_d is derived from and the seed K. As a consequence, it is not necessary to have a separated channel in order to obtain the encryption key. This scheme is presented in Figure 2.

We utilize the identity-based encryption algorithm of Boneh & Franklin [1] which is given by four algorithms:

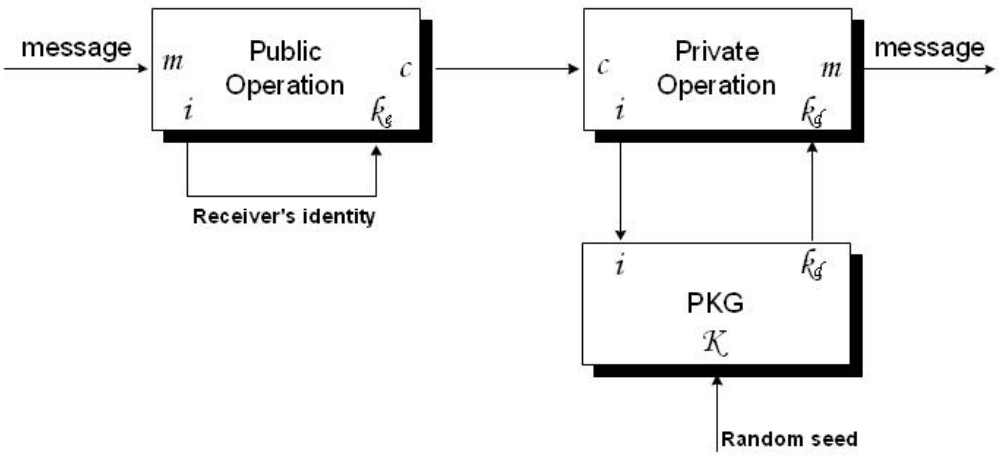


Figure 1. Identity-based cryptosystem (IBC).

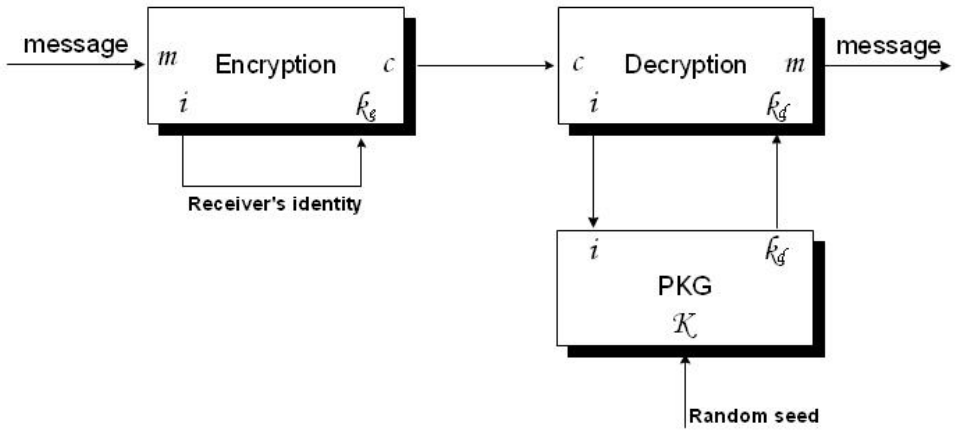


Figure 2. Identity-based encryption.

• ***Setup***: define the private system parameter s which is only known by the PKG, and the public system parameters $(\mathbb{G}_1, \mathbb{G}_2, P, n, H_1, H_2, P_{pub})$, where $\mathbb{G}_1$ is an additive group generated by point P and $\mathbb{G}_2$ is a multiplicative group, both of order n, $H_1 : \{0,1\}^* \longrightarrow \mathbb{G}_1$ is an special function called *map-to-point* hash function, $H_2 : \mathbb{G}_2 \longrightarrow \{0,1\}^\ell$ with ℓ as the string's length and $P_{pub} = sP$ is the global public key.

• ***Extract***: given a public identity $ID \in \{0,1\}^*$, compute the public key $Q_{ID} = H_1(ID) \in \mathbb{G}_1$ and the private key $S_{ID} = sQ_{ID}$, calculated by the PKG.

• ***Encrypt***: choose a random $r \in [1, ..., n]$ and calculate the ciphertext for the message M as follows, $C = \langle rP, M \oplus H_2(\hat{e}(Q_{ID}, P_{pub})^r) \rangle$.

• ***Decrypt***: given $C = \langle U, V \rangle$, compute $M = V \oplus H_2(\hat{e}(S_{ID}, U))$

This operation is used to decrypt a message. Due to the bilinear pairing properties, the last equation is equal to

$$\begin{aligned} M &= V \oplus H_2(\hat{e}(S_{ID}, U)) \\ &= M \oplus H_2(\hat{e}(Q_{ID}, P_{pub})^r) \otimes H_2(\hat{e}(S_{ID}, rP)) \\ &= M \end{aligned}$$

3.2 Identity-Based Signature

An identity-based signature scheme is as a mirror with respect to the identity-based encryption counterpart. To carry out the signing and verifying operations, message m is signed with private key k_d, both m and signature s are transmitted. Thereafter, any entity can verify s, using the sender's public key k_e. The general steps of the identity-based signature are shown in Figure 3.

We utilize the identity-based signature proposed by Hess in 2003 [5] that produces a signature in four phases:

• ***Setup***: choose a scalar s, compute $P_{pub} = sP$. Two hash functions are used: $H_1 : \{0,1\}^* \longrightarrow \mathbb{G}_1$ and $H : \{0,1\}^* \times \mathbb{G}_2 \longrightarrow \mathbb{Z}_q^*$, and P_{pub} are the PKG's private and public keys, respectively. Hash function H receives two inputs: a message m and one element in $\mathbb{G}_2$.

• ***Extract***: given a public identifier $ID \in \{0,1\}^*$, the PKG computes public key $Q_{ID} = H_1(ID) \in \mathbb{G}_1$ and private key $S_{ID} = sQ_{ID}$.

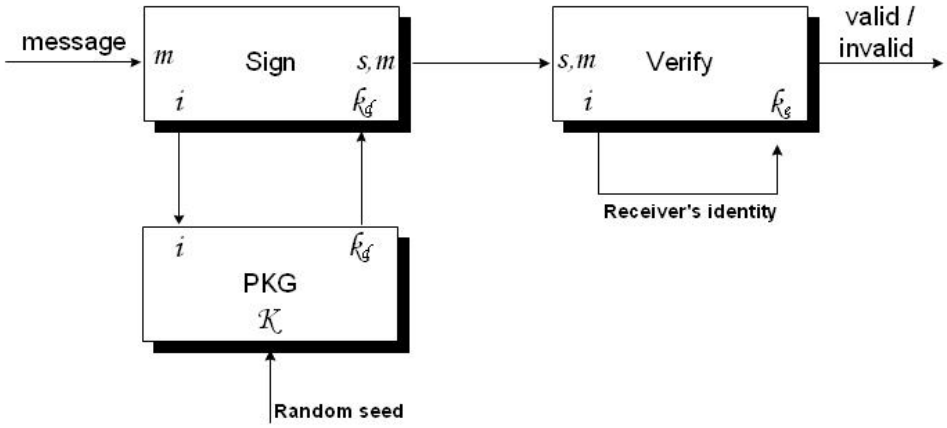


Figure 3. Identity-based signature.

• ***Sign***: given the signer's private key S_{ID}, and message $M \in \{0,1\}^*$; the signer chooses randomly $P_1 \in \mathbb{G}_1$ and a scalar $k \in [1, ..., n-1]$ and calculates:

$$\begin{aligned} r &= \hat{e}(P_1, P)^k \\ v &= H(M, r) \\ u &= vS_{ID} + kP_1 \end{aligned}$$

The pair (u, v) is a signature for message m.

• ***Verify***: given Q_{ID}, message M, and signature (u, v), the verification is as follows:

$$r = \hat{e}(u, P)\hat{e}(Q_{ID}, -P_{pub})^v$$

The signature is valid if and only if $v = H(M, t)$. In order to prove this step, we once again use the bilinear pairing property, so that $u = vS_{ID} + kP_1 = vsQ_{ID} + kP_1$, and $P_{pub} = sP$, we have:

$$\begin{aligned} r &= \hat{e}(u, P)\hat{e}(Q_{ID}, -P_{pub})^v \\ &= \hat{e}(vsQ_{ID} + kP_1, P)\hat{e}(vQ_{ID}, -sP) \\ &= \hat{e}(vsQ_{ID} + kP_1, P)\hat{e}(-svQ_{ID}, P) \\ &= \hat{e}(vsQ_{ID} - svQ_{ID} + kP_1, P) \\ &= \hat{e}(kP_1, P) \\ &= \hat{e}(P_1, P)^k \end{aligned}$$

4. PGP With IBC

Before presenting our identity-based PGP application, we would like to mention some works which have been published in order to offer the confidentiality and authentication services in mobile devices.

Several authentication protocols have been proposed for this scenario. For example, in [12, 13], authors presented protocols based on elliptic curve cryptosystems. With respect to the confidentiality service, Niansheng et al. [14] implemented AES (Advanced Encryption Standard) in mobile devices.

In the context of bilinear pairings in which we mainly focus our attention, Kawahara et al. [15] presented an efficient implementation of Tate pairing on a mobile phone using Java as their programming language. They reported significant fast timings and concluded that the cryptosystems based on pairing could be efficiently implemented in mobile phones.

Unfortunately, we could not find some protocols such that both services, authentication and confidentiality, were offered using identity-based cryptography. For this reason, we presented in this section an efficient implementation which was motivated in PGP protocol (Pretty Good Privacy) [16], this implies that we developed the basic block of security (encryption/decryption; sign/verify) with

the main idea of guaranteeing the confidentiality and authentication services over a specific mobile device: a Personal Digital Assistant (PDA).

4.1 Pretty Good Privacy (PGP)

PGP was created by Philip R. Zimmerman [16]. This protocol provides confidentiality and authentication services when digital information is sent through Internet.

PGP combines secret and public key cryptography. The functionality to encrypt a message is as follows. First, the sender uses PGP to create a session key, which is the secret key that is used to encrypt the plaintext. Then, the session key is encrypted with a receiver's public key and it is transmitted along with the ciphertext to the receiver. To decrypt, the receiver uses his/her private key to obtain the session key that is used to decrypt the ciphertext and get the message.

In order to offer the authentication, data integrity, and non-repudiation services, this scheme uses a hash function on the message to be signed. Let us call the result of such hashing *message digest*. Then, a message digest is signed by the signer using his/her private key. The plaintext and the signature are transmitted to the receiver that uses the sender's public key to verify the signature.

4.2 PGP using IBE

Figure 4 shows the encryption and decryption operations. We now discuss this figure from left to the right. First, the sender generates a random session key (128-bits) which uses a private key in the cipher AES to encrypt the plaintext. The IBE scheme is utilized to encrypt the session key with a sender's public key. Then, the sender transmits the ciphertext and the encrypted session key to the receiver. On the right side, the receiver requests her/his private key S_{ID} to PKG and uses it to decrypt the session key which is a private key of AES. Finally, the receiver decrypts the ciphertext.

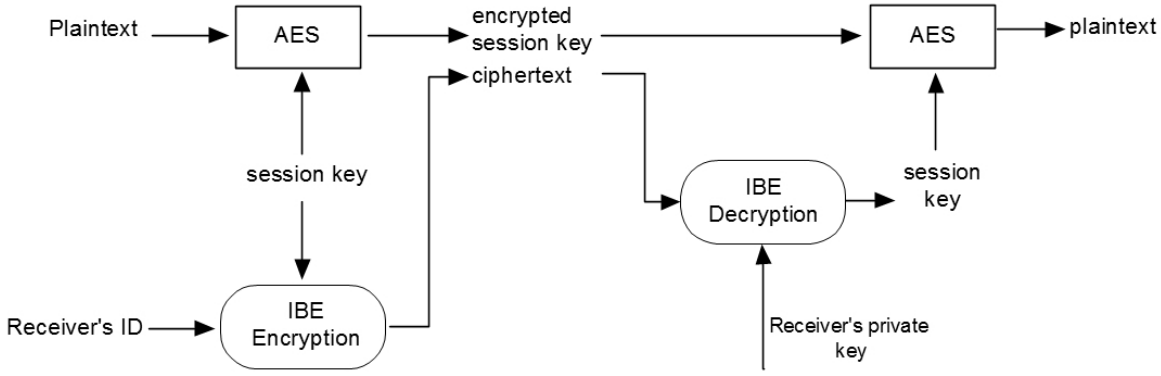


Figure 4. PGP Id-based encryption/decryption.

4.3 PGP using IBS

In this process, our application takes a message and produces the corresponding 256-bit message digest, which is signed using the sender's private key S_{ID} that was previously generated by the PKG. On the other side, any entity verifies the signature, producing the message digest and using the sender's public key. Figure 5 illustrates the signature and verification process.

5. TIMINGS

This section presents the timings obtained for all basic cryptographic primitives using the bilinear pairings. Our system was implemented in a PDA Sharp Zaurus-5600 that has an Intel Processor XScale 400 MHz PXA250, ROM Memory of 32 Mb, RAM Memory of 64 Mb, TFT LCD Screen 3.5" (240x320 pixels) and Operating System Linux2 (Embedix3). Regarding the software, the library was developed in C/C++ Programming Language and GNU/Linux Compiler. To generate test vectors, we used MAPLE Language.

Figure 6 illustrates the architecture of our application divided by layers. From a left-to-right perspective, the first layer represents the arithmetic of the binary and ternary fields required to obtain the bilinear pairing function. The next layer depicts the cryptographic primitives and the last layer represents all possible applications that can be implemented using the bilinear pairing function.

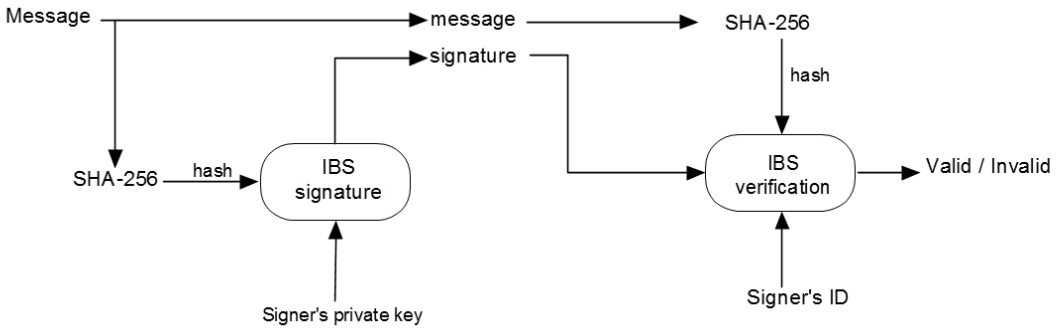


Figure 5. PGP Id-based signature/verification.

Figure 6. PGP Id-based cryptography in a PDA.

Finite field	Bilinear pairing Time *ms*
$\mathbb{F}_{2^{233}}$	29.5
$\mathbb{F}_{3^{97}}$	66.9

Table 1. Times obtained for the bilinear pairing operation on the PDA.

Table I presents timings results on the PDA device for the η_T bilinear pairing function in $\mathbb{F}_{2^{233}}$ and $\mathbb{F}_{3^{97}}$.

Table II illustrates the execution time of basic operations on the PDA architecture. Both encryption/decryption and signing/verifying were tested with a message-length of 128 bits. The difference of times between characteristic two and three is due to the special hash *function map-To-point*. As shown by Barreto & Kim [8], one can implement efficiently the map-to-point function in characteristic three by solving a cubic equation over $\mathbb{F}_{3^m}$.

Operation	Characteristic 2 $\mathbb{F}_{2^{233}}$	Characteristic 3 $\mathbb{F}_{3^{97}}$
Encryption (IBE)	95.085 *ms*	78.8 *ms*
Decryption (IBE)	30.4 *ms*	67.4 *ms*
Signature (IBS)	46.7 *ms*	88.396 *ms*
Verification (IBS)	123.37 *ms*	143.6 *ms*
Map-to-point	58.2 *ms*	7 *ms*
AES Encryption	220 *ms*	
AES expansion key	20.2 *ms*	

Table 2. Time obtained in basic blocks on PDA.

6. Conclusions

We implemented six main building blocks and modules, namely, two η_T bilinear pairing functions on finite field $\mathbb{F}_{2^{233}}$ and $\mathbb{F}_{3^{97}}$; the AES cipher algorithm, two map-To-point functions on $\mathbb{F}_{2^{233}}$ and $\mathbb{F}_{3^{97}}$, and the SHA-256 function hash.

The combination of all these blocks together produced a fully functional application of PGP using Identity-Based Cryptography on a Personal Digital Assistant (PDA).

We obtained good efficiency in our application, thanks to the use of programming language C++ along with the use of efficient pairings algorithms.

References

[1] Boneh D. and Franklin M. K., Identity-Based Encryption from the Weil Pairing, Proceedings of CRYPTO '01, the 21st Annual International Cryptology Conference on Advances in Cryptology, 2001, pp. 213-229, Springer-Verlag, London, UK.

[2] Beuchat J-L., Brisebarre N., Detrey J., Okamoto E., Shirase M., and Takagi T., Algorithms and Arithmetic Operators for Computing the □T Pairing in Characteristic Three, IEEE Trans. Comput., Vol. 57, Issue. 11. 2008, pp. 1454-1468, IEEE Computer Society, Washington, DC, USA.

[3] Shamir A., Identity-based cryptosystems and signature schemes, Proceedings of CRYPTO'84 on Advances in Cryptology, 1985, pp. 47-53, Santa Barbara, California, United States.

[4] Gorantla M.C., Gangishetti R., and Saxena A., A survey on ID-Based Cryptographic Primitives, Cryptology ePrint Archive, Report 2005/094, 2005.

[5] Hess F., Efficient Identity Based Signature Schemes Based on Pairings, SAC '02: Revised Papers from the 9th Annual International Workshop on Selected Areas in Cryptography, 2003, pp. 310-324, Springer-Verlag, London, UK.

[6] Harrison K., Page D. and Smart N., Software Implementation of Finite Fields of Characteristic Three, for use in Pairing-Based Cryptosystems, London Mathematical Society J. Comput. Math., Vol. 5, pp. 181-193, 2002.

[7] Duursma I. And Lee H-S., Tate Pairing Implementation for Hyperelliptic Curves y2= xp - x + d, ASIACRYPT '03: Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security, 2003, Vol. 2894, pp. 111-123, Springer-Verlag.

[8] Barreto P., Kim H. Y., Lynn B., and Scott M., Efficient Algorithms for Pairing-Based Cryptosystems, CRYPTO '02: Proceedings of the 22nd Annual International Cryptology Conference on Advances in Cryptology, 2002, Vol. 2442, pp. 354-368, Springer-Verlag, London, UK.

[9] Galbraith S. G., Harrison K. and Soldera D., Implementing the Tate Pairing, ANTS-V: Proceedings of the 5th International Symposium on Algorithmic Number Theory, 2002, Vol. 2369, pp. 324-337, Springer-Verlag, London, UK.

[10] Beuchat J-L., Shirase M., Takagi T., and Okamoto E., An Algorithm for the □T Pairing Calculation in Characteristic Three and its Hardware Implementation, ARITH '07: Proceedings of the 18th IEEE Symposium on Computer Arithmetic, 2007, pp. 97-104, IEEE Computer Society, Washington, DC, USA.

[11] Beuchat J-L., Brisebarre N., Detrey J., Okamoto E., Shirase M., and Rodríguez-Henríquez F., A Comparison Between Hardware Accelerators for the Modified Tate Pairing over F2m and F3m, Cryptology ePrint Archive, Report 2008/115, 2008.

[12] Yang J-H., and Chang C-C., An ID-Based Remote Mutual Authentication with Key Agreement Scheme for Mobile Devices on Elliptic Curve Cryptosystem, Computers & Security, 2009, Vol. 28, Issues 3-4, pp. 138-143.

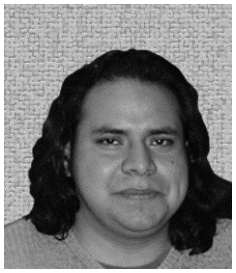
[13] Chang C-C., Lin S-Y., Yang J-H., Efficient user authentication and key establishment protocols with perfect forward secrecy for mobile devices, Proceedings of the 2009 Ninth IEEE International Conference on Computer and Information Technology, 2009, Vol. 2, pp. 131-135, Xiamen, China.

[14] Niansheng L., Donghui G., and Jiaxiang H. AES Algorithm Implemented for PDA Secure Communication with Java, Anti-counterfeiting, Security, Identification, 2007 IEEE, pp. 217-222.

[15] Kawahara Y., Takagi T., and Okamoto E., Efficient Implementation of Tate Pairing on a Mobile Phone Using Java, CIS, 2006, pp. 396-405.

[16] Callas J., Donnerhacke L., Finney H., Shaw D., and Thayer R., RFC-4880 OpenPGP Message Format, 2007.

Authors´ Biographies

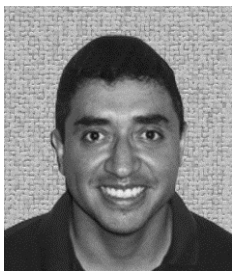


Luis MARTÍNEZ-RAMOS

Luis Martínez-Ramos received his B.S. degree in computer system engineering from Escuela Superior de Cómputo (Higher Education School of Computing), ESCOM, of Instituto Politécnico Nacional (National Polytechnic Institute), IPN, Mexico, in 2006, and his M.S. degree from Centro de Investigación y Estudios Avanzados (Center for Research and Advanced Studies) of Instituto Politécnico Nacional (National Polytechnic Institute), CINVESTAV-IPN, Mexico, in 2008. His research interests include cryptography and efficient implementation of algorithms. Contact him at lumramos@computacion.cs.cinvestav.mx, luis.mpzr@gmail.com.

María de Lourdes LÓPEZ-GARCÍA

She received her M.S. degree in computer science from Benemérita Universidad Autónoma de Puebla (Meritorious Autonomous University of Puebla), BUAP, Mexico, in 2007, and her Ph.D. from CINVESTAV-IPN, Mexico, in 2011. Her research interests include public-key cryptography, e-voting protocols, provable security and information security. Contact her at mlopez@computacion.cs.cinvestav.mx.



Francisco RODRÍGUEZ-HENRÍQUEZ

He is an associate professor in the Computer Science Department at CINVESTAV-IPN, Mexico. His research interests include cryptography, computer arithmetic, and efficient implementation of algorithms on reconfigurable hardware devices. Rodríguez-Henríquez has a PhD. in electrical and computer engineering from Oregon State University. He is a member of IEEE, the Academia Mexicana de Ciencias, and an alumni member and research associate of the Information Security Laboratory at Oregon State University. Contact him at francisco@cs.cinvestav.mx.