On Assurance of Information Security using Elliptic Curves Cryptosystems

Ion Tutănescu, Constantin Anton, Laurenţiu Ionescu, Daniel Caragaţă
University of Piteşti, Romani

***Abstract* – We present in this paper an important area of information security emerged in the last decades, namely Elliptic Curves Cryptosystems (ECC). Compared to traditional public-key cryptosystems like RSA or Diffie-Hellman, ECC offers equivalent security with smaller key sizes; these result in faster computations, lower power consumption, as well as memory and bandwidth savings. ECC are more and more considered as an attractive public-key cryptosystem for mobile/wireless environments. ECC are especially useful for mobile devices, which are typically limited in terms of their CPU, power and network connectivity.**

ECC are the next frontier in the use of security mechanisms by providing good security margins with lower computational cost. ECC's domain is an important field emerged in information security. The elliptic curves (EC) are used for conceiving efficient factorization algorithms and for proving the primality. They are used in public key cryptosystems and in pseudorandom bit generators, too. The elliptic curves were also applied in Codes Theory, where they were used to create very good error protected codes.

In this paper, our aim is to examine the security, implementation and performance of ECC applications on various mobile devices. Also, our goal is to compare ECC and conventional PKC performances. Doing these, we want to prove that ECC could become the next-generation of PKC.

***Key words*: elliptic curves, hyper-elliptic curves, cryptographic algorithms, public key, factorization, information security.**

I. INTRODUCTION

Elliptic curves were first proposed for use in public-key cryptography by Koblitz [1] and Miller [2]. Other work on the security and implementation of elliptic curve cryptosystems (ECC) was reported in Menezes [3], Menezes, Okamoto and Vanstone [4] [5].

Some public-key cryptosystems using hyper-elliptic curves were proposed [6]. Hyper-elliptic curve cryptosystems were proposed by Koblitz [7], but little research has since been done regarding their security and practicality.

At the time of their discovery, ECCs were considered unpractical. Since then, they were deeply and intensively researched. ECC can be used for providing the following security services:

- confidentiality,
- authentication,
- data integrity,
- non-repudiation,
- authenticated key exchange.

The progress in factorization and parallel processing leads to the need of larger and larger keys for public-key cryptosystems. But, the growth of keys length will do these cryptosystems slower than before. The use of ECC allows the increasing of security. In the same time, ECC decreases the overloading.

ECC security consists in the difficulty to calculate logarithms in discrete fields (discrete logarithms problem): being given A (an element from a finite field) and A^x, it is practically impossible to calculate x when A is big enough [16] [17].

Actually, there are several cryptosystems which are based on discrete logarithms problem in multiplicative group Z^*_p. But these cryptosystems can be also defined in any other finite group, as the group of points of an elliptic curve. The elliptic curves are suitable in applications where:

- the computing power is limited (intelligent cards, wireless devices, PC boards);
- memory size on integrated circuit is limited;
- a great speed of computing is necessary;
- digital signing and its verification are used intensively;
- signed messages have to be transmitted or memorized;
- digital bandwidth is limited (mobile communications, certain computer networks).

From the advantages of ECC usage, there can be mentioned:

- increased security: cryptographic resistance per bit is much greater than those of any public-key cryptosystem known at present time;
- substantial economies in calculus and memory needs in comparison with other cryptosystems;
- great encryption and signing speed both in software and hardware implementation;
- ECC are ideal for small size hardware implementations (as intelligent cards);

- encryption and signing can be done in separate stages.

The intense research done on public-key cryptosystems, based on elliptic curves, demonstrated that ECC are suitable for the vast majority of existing applications.

An ECC with 160-bit key offers a security level equivalent with that offered by a cryptosystem based on a 1024-bit Z_p field [15] [20].

Because of this, ECC provide a feasible method of implementation for a high level security system on a PC card, on an intelligent card or on a mobile communications device.

II. ALGORITHM DESCRIPTION

Elliptic curves are mathematical constructions. An elliptic curve can be defined over any field (of real, rational or complex numbers), but - generally speaking - the elliptic curves used in cryptography are defined over finite fields.

An elliptic curve E consists of:

a) several elements (named *points*) of type (x, y) which satisfy the equation:

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

where a and $b \in Z_p$ are constant, so that

$$4a^3 + 27b^2 \neq 0 \pmod p \quad (2)$$

and p is a prime number;

b) a singular element, named „*point from infinite*"; intuitively, this point can be seen as the point from the top and from the bottom of any vertical line.

An elliptic curve E has an Abelian group structure on addition. Addition of two points on an elliptic curve is defined using a simple set of rules, as seen in Figure 1, where:

$$P_3 = P_1 + P_2 \quad (3)$$

Being given two points on E, $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$, we have next cases:

a) if $x_2 = x_1$ and $y_2 = -y_1$,
then

$$P_1 + P_2 = 0 \quad (4)$$

b) in all other situations

$$P_1 + P_2 = (x_3, y_3), \quad (5)$$

where

$$x_3 = \lambda^2 - x_1 - x_2,$$
$$y_3 = \lambda(x_1 - x_3) - y_1 \text{ and} \quad (6)$$
$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1}, \text{if } P_2 \neq P_1 \\ \dfrac{3x_1^2 + a}{2y_1}, \text{if } P_2 = P_1 \end{cases}$$

Addition on an elliptic curve is the correspondent operation of multiplication in public key cryptosystems, and multiple addition is the correspondent of exponentiation.

Although the calculation rules in the group of an elliptic curve's points seem complicated enough, their arithmetic can be efficiently implemented; the calculus in this group is realized much faster than in group Z_p.

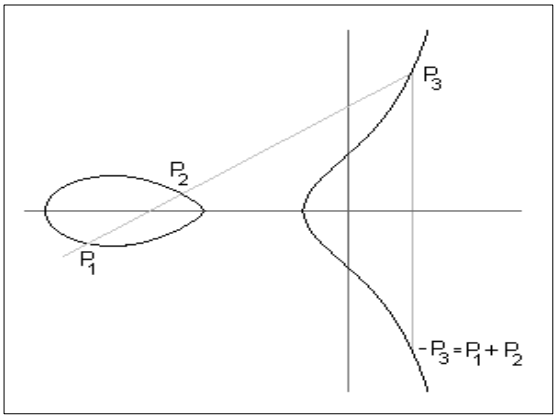

Figure 1. Addition on an elliptic curve

III. NUMERICAL APPLICATION

The curve E is expressed below as:

$$y^2 = x^3 + x + 6 \quad (7)$$

over Z_{11} field.

First, the points of E are calculated. In order to do this, it is calculated z for any $x \in Z_{11}$:

$$z = x^3 + x + 6 \bmod 11 \quad (8)$$

Next, it is tested if z is a square remainder, for a given x, using Euler criterion. Using the calculus formula of the square roots for a *modulo p* square remainder, we get:

$$\pm z^{\frac{11+1}{4}} \bmod 11 = \pm z^3 \bmod 11 \quad (9)$$

The calculus of E elliptic curve's points is illustrated in Table I:

TABLE I. Calculus of the elliptic curve's points

x	$x^3 + x + 6 \bmod 11$	Is it a square remainder mod 11?	y
	6	no	
1	8	no	
2	5	yes	4.7
3	3	yes	5.6
4	8	no	
5	4	yes	2.9
6	8	no	
7	4	yes	2.9
8	9	yes	3.8
9	7	no	
10	4	yes	2.9

The rang of EC points' group is a prime number, therefore the group is a cyclic one.

Let suppose that α = (2, 7) is the group's generator. We can calculate multiples of α, which are powers of α, because the group is additive.

For calculating

$$2\alpha = (2, 7) + (2, 7), \quad (10)$$

first we calculate

$$\lambda = \left(3 \cdot 2^2 + 1\right) \cdot (2 \cdot 7)^{-1} \bmod 11 = 2 \cdot 3^{-1} \bmod 11 = 2 \cdot 4 \bmod 11 = 8$$

In the same way, the following multiples are calculated and the following results are obtained:

$$\begin{array}{lll} \alpha = (2,7) & 5\alpha = (3,6) & 9\alpha = (10,9) \\ 2\alpha = (5,2) & 6\alpha = (7,9) & 10\alpha = (8,8) \\ 3\alpha = (8,3) & 7\alpha = (7,2) & 11\alpha = (5,9) \\ 4\alpha = (10,2) & 8\alpha = (3,5) & 12\alpha = (2,4) \end{array} \quad (11)$$

It can be noticed that chosen α is truly the group generator.

Let study further on an El Gamal encryption example on the same elliptic curve, supposing that α = (2, 7) and taking the secret exponent as $d_A = 7$. We will have:

$$\beta = 7\alpha = (7, 2) \quad (12)$$

The *encryption* of plaintext x with the key k is done as follows:

$$e_K(x, k) = (k\alpha, x + \beta k) = (k(2,7), x + k(7,2)), \quad (13)$$

where

$0 \le k \le 12$ and $x \in E$.

The *decryption* is done so:

$$d_k(y_1, y_2) = y_2 - 7y_1. \quad (14)$$

Let suppose that the user A wants to encipher the message x = (10, 9), which is a point on elliptic curve E, in order to transmit it to the user B. In

order to do this, user A chooses the random value k=3 and calculates

$$y_1 = 3(2,7) = (8,3) \quad (15)$$

and

$$y_2 = (10,9) + 3(7,2) = (10,9) + (3,5) = (10,2) \quad (16)$$

The enciphered text is:

$$y = ((8, 3), (10, 2)). \quad (17)$$

At reception, user B deciphers the message:

$$x = (10,2) - 7(8,3) = (10,2) - (3,5) = (10,2) + (3,6) = (10,9) \quad (18)$$

Therefore, the original plaintext was found.

4. ECC SECURITY AND ADVANTAGES

The security of an ECC depends on the difficulty of solving the discrete logarithm problem over elliptic curves. The correspondence of discrete logarithms problems to elliptic curves is the problem of logarithms on elliptic curves. It is defined as following: being given a point G on an elliptic curve of rank r (r = number of the points on the curve) and another point Y, find an unique point x ($0 \le x \le r - 1$) so that $Y = x \cdot G$, that is Y is a x-multiple of G.

The best attacks against the problem of logarithms on elliptic curves are the methods applicable to any group, so that the attacks are very inefficient on a certain particular case.

Two general methods of solving this problem are known [8]: one is the square root method, which is a general method for the discrete logarithm problem; the other is the Silver-Pohlig-Hellman (SPH) method, which factors the order of a curve into small primes and solves the discrete logarithm problem as a combination of discrete logarithms for small numbers.

The square root method is the most general attacking method for the discrete logarithm problem, and its computation time is proportional to the exponent of half the key length; that is, the computation time varies exponentially with respect to the key length.

A public key cryptosystem is regarded as being very secure against an attack if the attack takes an

exponential amount of time with respect to the key length. From this criterion, we can say that ECCs are very secure against the square root method. The SPH method is effective only when the order of the curve is expressed as a product of small primes. Otherwise, the computation time is equivalent to that of the square root method. Therefore, for an ECC, if we select the order of the elliptic curve to be a prime or nearly a prime whose factors include a large prime, the computation time needed to break the ECC will vary exponentially. Therefore a high level of security can be achieved [11].

Comparing the security of ECCs with that of RSA it can be concluded that the security of RSA resides in the difficulty of factoring large numbers.

The number field sieve is the most effective known method for factoring large numbers, and it takes a sub-exponential amount of computation time with respect to the key length to do that task. Therefore, the best-known attack against RSA takes a sub-exponential amount of time with respect to the key length. An attacking method with a sub-exponential time/key-length relationship takes less time than one with an exponential relationship (and more time than a method with polynomial relationship). This is why the securities of 1024-bit RSA and 160-bit ECC are equivalent.

Due to the lack of specialized attack methods, ECC – using reduced length keys – offer the same security level as that offered by cryptosystems based on the problem of discrete logarithms, using very big keys. The majority of public key systems in use today use RSA and Diffie-Hellman algorithms. The US National Institute for Standards and Technology (NIST) had recommended that 1024 bits are sufficient for use until 2010.

Then, NIST recommends that they be upgraded to something providing more security: one option is to simply increase the public key parameter size to a level appropriate for another decade of use; another option is to move from first generation public key algorithms to elliptic curves.

Elliptic curve systems offer more security per bit increase in key size than either RSA or Diffie-Hellman public key systems. In addition, EC cryptosystems are more computationally efficient than the first generation public key systems, RSA and Diffie-Hellman.

Also, in channel-constrained environments, elliptic curves offer a much better solution than first generation public key systems like Diffie-Hellman. A comparative analysis between ECC and RSA classes, that provide the same level of security [14], is displayed in Table II.

Table II. Comparative analysis between ECC and RSA

Security Level	Public key size[1] (bits)		Ratio ECC/RSA public keys
	ECC	RSA	
80	192	1024	5x smaller
112	224	2048	9x smaller
128	256	3072	12x smaller
192	384	7680	20x smaller
256	521	15360	29x smaller

[1] NIST-recommended key sizes.

There are a variety of different choices for choosing an elliptic curve as the foundation of a public key system. NIST has standardized on a list of 15 elliptic curves of varying sizes: ten of these curves are for what are known as binary fields and 5 are for prime fields. Those curves listed provide cryptography equivalent to symmetric encryption algorithms with keys of length 80, 112, 128, 192, 256 bits and beyond.

V. APPLICATIONS OF ECC

Using ECC for secure transactions makes sense for a number of reasons. Transactions will need to be processed more efficiently – more and smaller devices that require security are being connected to the Internet, from onboard automotive computers to smart cards and process control sensors.

The US National Security Agency (NSA) has decided to move to elliptic curve based public key cryptography for protecting both classified and unclassified information. Where appropriate, NSA plans to use the elliptic curves over finite fields with large prime moduli (256, 384, and 521 bits) published by NIST.

The Cryptographic Modernization Initiative in the US Department of Defense aims at replacing almost 1.3 million existing equipments over the next 10 years.

In addition, the Department's Global Information Grid will require a vast expansion of the number of security devices in use throughout the US Military. This will necessitate change and rollover of equipment with all major US allies.

Most of these needs will be satisfied with a new generation of cryptographic equipment that uses elliptic curve cryptography for key management and digital signatures [12].

The United States, the UK, Canada and other NATO nations have all adopted some form of elliptic curve cryptography for future systems to protect classified information throughout and between their governments.

The application of elliptic curves to the field of Information Security opened up a wealth of possibilities in terms of security, encryption, and real-world applications [18] [19] [21] [22] [23].

A short survey of ECC applications seen on the market today is presented below. Results of this survey can be broadly divided into four categories: the Internet, smart cards, PDAs and PCs [9][10].

a. Internet

SUN Microsystems contributed to the implementation of an ECC cryptographic library and also to a common hardware architecture for accelerating ECC (as well as RSA) to be used in OpenSSL. OpenSSL is a developmental toolkit for the implementation of SSL (Secure Sockets Layer) and TLS (Transport Layer Security) protocols, which are commonly used today in over-the-web transactions and secure document transfers. SUN corporation hopes to promote ECC standardization with SSL, which is the dominant security protocol used on the web today.

ECC uses less bandwidth than alternative cryptographic algorithms for SSL/TLS. The processing power itself is increasing and hackers have even more resources available to them than ever before. Although 1024-bit RSA keys are currently most often used today, use of 2048-bit key is becoming more and more common.

Figure 2 shows the impact that ECC usage has in server response time [13]. According to this chart, someone using RSA instead of ECC would have to purchase and maintain 3.5 times as many web servers in order to handle the same amount of traffic. Use of ECC cipher suites can offer significant performance benefits to SSL clients and servers, especially as security needs increase in time. Already, there is significant momentum behind widespread adoption of the Advanced Encryption Standard (AES) which specifies the use of 128-bit, 192-bit and 256-bit symmetric keys.

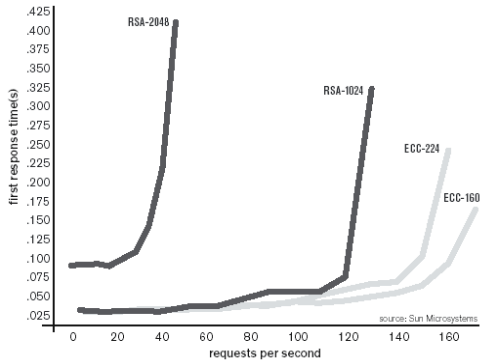

Figure 2. Server Response Times

Key sizes for public key cryptosystems used to establish AES keys will also need to increase from current levels. This trend bodes well for the future of ECC and not just for wireless environments.

b. Smart Cards

The idea of implementing digital signature/ identification schemes in the form of smart cards has quickly gained momentum. Smart cards are one of the most popular devices for the use of ECC.

Elliptic curve cryptosystems can provide security with short key lengths, requiring less data for storage on a smart card and less computation. Many manufacturing companies are producing smart cards that make use of elliptic curve digital signature algorithms. These manufacturing companies include Phillips, Fujitsu, MIPS Technologies and DataKey, while vendors that sell these smart cards include Funge Wireless and Entrust Technologies.

Smart cards are very flexible tools and can be used in many situations. For example, smart cards are being used as bank (credit/debit) cards, electronic tickets and personal identification (or registration) cards. Some years ago, the US Treasury Department's Bureau of Engraving and Printing completed a four-month e-commerce pilot program involving the use of smart cards and ECC with SET (Secure Electronic Transaction) specifications.

SET is a standard that enables secure credit card transactions over the Internet. The pilot program tested the use of smart cards, embedded with ECC technology, in making online purchases. This program involved a total of nine companies, including MasterCard, Certicom (who supplied the ECC algorithms), Digital Signature Trust Co (who supplied the MasterCard smart cards) and GlobeSet (a SET vendor), a.o. The previous version of SET, version 1.0, supports only RSA Data Security encryption algorithms, but MasterCard hopes to add ECC to the upcoming version of SET.

c) ***PDAs***. PDAs are a very popular choice for implementing public key cryptosystems because they have more computing power compared to most of the other mobile devices, like cell phones or pagers. However, they still suffer from limited bandwidth and this makes them an ideal choice for using ECC. For example, 3Com4 Corporation teamed up with Certicom in order to implement ECC in its PalmPilot organizer series and Palm Computing platform. This feature provides protection of confidential information on the hand-held organizers, user authentication in wireless communications and e-commerce transactions, and also ensures data integrity and proof of transactions.

d) ***PCs***. Constrained devices have been considered to be the most suitable platforms for implementing the ECC. Several companies have created software products that can be used on PCs to secure data, encrypt e-mail messages and even instant messages with the use of ECC.

PC Guardian Technologies is one such company that created the Encryption Plus Hard Disk and Encryption Plus Email software products. The former makes use of both RSA and EC Diffie-Hellman while the latter makes use of a strong 233-bit ECC key to encrypt its private AES keys.

VI. CONCLUSION

Examining the security, implementation and performance of ECC applications on various mobile devices, we can conclude that ECC is the most suitable PKC scheme for use in a constrained environment. ECC efficiency and security makes them an attractive alternative to conventional cryptosystems, like RSA and DSA, not just in constrained devices, but also on powerful computers.

Elliptic curve cryptosystems are expected to become the next-generation public key cryptosystems. ECCs require a shorter key length than RSA cryptosystems, which are the de facto standards of public key cryptosystems, but provide equivalent security levels.

Because of the shorter key length, ECCs are fast and can be implemented with less hardware.

Although ECC's security has not been completely evaluated, it is expected to come into widespread use in various fields in the future because of its compactness and high performance when it is hardware-implemented.

The multitude of elliptic curves with familiar cryptographic properties, but conveniently without properties that commonly facilitate cryptanalysis suggests the need to continue these studies with different elliptic curves and different cryptosystems.

Previously neglected elliptic curves might be applied to the cryptosystems studied so far, since the choice of curves can seriously affect the security and efficiency of an elliptic curve cryptosystem. The search for suitable elliptic curves will be ongoing.

In this paper, we examined the security, implementation and performance of some ECC applications on various mobile devices and to compared ECC and conventional PKC performances. Our opinion is that ECC could become the next-generation of PKC.

References

[1] N. Koblitz, "Elliptic curve cryptosystems" in *Mathematics of Computation*, 48 (1987), pp. 203-209.

[2] V. S. Miller, "Use of elliptic curves in cryptography", in *Advances in Cryptology* – CRYPTO'85 (LNCS 218), pp. 417- 426, 1986.

[3] A. Menezes, "Elliptic curve public key cryptosystems" in Kluwer Academic Publishers, Boston, 1993.

[4] A. Menezes, T. Okamoto and S. Vanstone, "Reducing elliptic curves logarithms to logarithms in a finite field" in *IEEE Transactions on Information Theory*, 39 (1993), pp.1639 – 1646.

[5] A. Menezes, S. Vanstone, "Elliptic Curve Cryptosystems and Their Implementation", Journal of Cryptology, pp. 209-224, 1993.

[6] T. Okamoto, K. Sakurai, "Efficient Algorithms for the Construction of Hyperelliptic Cryptosystems", Advances in Cryptology - CRYPTO '91 Proceedings, pp. 267-278, 1992.

[7] N. Koblitz, "Hyperelliptic cryptosystems", *Journal of Cryptology,* 1 (1989), 139 - 150.

[8] N. Torii, K. Yokoyama, "Elliptic Curve Cryptosystem", FUJITSU Sci. Tech. J., 36, 2, December 2000.

[9] W. Chou. "Elliptic Curve Cryptography and Its Applications to Mobile Devices", University of Maryland, College Park, USA.

[10] V. Gupta, S. Gupta, S. Chang, "Performance Analysis of Elliptic Curve Cryptography for SSL", *WiSe'02,* September 28, 2002, Atlanta, Georgia, USA.

[11] V. Srivastava, S. Sharma, "Performance analysis of elliptic curve cryptography in network systems", *Proceedings of 3rd International Workshop in Wireless Security Technologies*, IWWST '05, 4-5 April, 2005, London, UK, pp. 144-151.

[12] NSA-Central Security Service, "The Case for Elliptic Curve Cryptography", http://www.nsa.gov/business/programs/elliptic_curve.shtml.

[13] *Certicom's Bulletin of Security and Cryptography*, "Code and Ciphers", vol. I, no. 4, 2004.

[14] *Certicom's Bulletin of Security and Cryptography*, "Code and Ciphers", vol. II, no.2, 2005.

[15] I. Tutănescu, "Applications of Elliptic Curves Cryptosystems", MCC'2007 Conference Proceedings, Bonn, Germany, 2007.

[16] K. Rabah, "Using Elliptic Curve Cryptography to Secure Online Data&Content", *Information Security Research Journal*, Vol. 1, No. 1, July 2009.

[17] K. Rabah, "Using Elliptic Curve Cryptography for Information Security", Information Security Research Journal Vol. 1, No. 2, July 2009.

[18] K.S. Sung, H. Ko, H.S. Oh, "XML Document Encrypt Implementation Using Elliptic Curve Cryptosystem", ICCIT 2007, Korea.

[19] M. W. Paryasto, S. Sutikno, A. Sasongko, "Issues in Elliptic Curve Cryptography Implementation", Internetworking Indonesia Journal, Vol. 1, No. 1, 2009.

[20] R. Shanmugalakshmi, M.Prabu, " Research Issues on Elliptic Curve Cryptography and its applications", IJCSNS *International Journal of Computer Science and Network Security*, Vol. 9 No. 6, June 2009.

[21] Y. K. Lee, K. Sakiyama, L. Batina, I. Verbauwhede, "Elliptic-Curve-Based Security Processor for RFID", *IEEE Transactions on Computers,* Vol. 57, No. 11, November 2008, pp. 1514-1527.

[22] G. M. Dormale, J.J. Quisquater, "High-speed hardware implementations of Elliptic Curve Cryptography: A survey" , *Journal of Systems Architecture*, Vol. 53, Issues 2-3, February-March 2007, pp. 72-84.

[23] I. Tutănescu, C. Anton, D. Caragaţă, „Use of Elliptic Curves Cryptosystems in Information Security", 5th International Conference on Information Technology (ICIT'2011), May 11-13, 2011, Amman, Jordan.

