Comparative Study on Asymmetric Key Cryptography Algorithms for Heterogeneous Network

Maria Nenova, Georgi Iliev, Kiril Kassev

***Abstract* – A great amount of cryptographic devices and standards apply public key cryptography, which is based on the problem of factorization of prime big digits (RSA) and retrieval of discrete logarithm proposed by El Gamal. Implementation of the approach of eliptic curves gives equivalent protection in comparison with earlier developed protocols, and it has less amount of bits.**

***Keywords* – RSA, Elliptic curves, security of information, asymmetric cryptographic algorithms.**

I. INTRODUCTION

The ability of a cryptosystem to save the needed information is defined as strength of the system. This strength is provided by the implemented key and its number of bits. Public-key cryptography is one of the most widely used technology for secure transmission of information via internet and communication systems. The concept was proposed first by Diffie and Hellman [1]. This type of cryptography is called asymmetric, and the main asymmetric algorithm is RSA – developed by Rivest, Shamir, and Adleman [2]. In asymmetric encryption algorithms are used different keys for encryption and decryption, and the decryption key cannot be derived from the encryption key, and this property is their main advantage. They are also called public key methods, and are important because they can be used for transmitting encryption keys or other data securely. The types of asymmetric encryption algorithms in heterogeneous networks are as follows: RSA, Knapsack, Digital Signature Algorithm, El Gamal, ECDSA, XTR. [3], [4], [5].

The RSA encryption algorithm is based on the multiplying of two large secret prime numbers and this is an easy forward function. In the inverse way the finding factor operation is much more difficult. The problem for an attacker is the computing of factorizing **n**. The operation performed using the cryptosystem is arranged that the operations wished to be tractable require the multiplication. The operations which should be made difficult are the finding of the plaintext from the chifer text using only the public key. This action requires performing the inverse operation — solving the factoring problem.

[1]Maria V. Nenova is with the Faculty of Telecommunications, Technical University of Sofia, 1756 Sofia, Bulgaria, E-mail: mvn@tu-sofia.bg

[2]Georgi L. Iliev is with Faculty of Telecommunications, Technical University of Sofia, 1756 Sofia, Bulgaria, E-mail: gli@tu-sofia.bg

[3]Kiril M. Kassev is with Faculty of Telecommunications, Technical University of Sofia, 1756 Sofia, Bulgaria, E-mail: kmk@tu-sofia.bg

During the last decades in order to achieve higher level of security, the amount of bits increases, which leads to big computational load. The approach implementing elliptic curves *E* ensure equivalent security in difference with earlier developed protocols, with les amounts of bits. Cryptosystems based on elliptic curves *E* are proposed in 1986 by Victor Miller and Neal Koblitz [3].

TABLE I
A comparison of key sizes needed to achieve equivalent level of security with different methods.

ECC Key Size	RSA Key Size	Key-Size Ratio	AES Key Size
163	1,024	1:6	n/a
256	3,072	1:12	128
384	7,680	1:20	192
512	15,360	1:30	256
Key sizes in bits.		*Source: Certicom, NIST*	

Those cryptographic algorithms are based on the problem of finding a discrete logarithm for points of an elliptic curve over a finite field, and exactly they are used in modern cryptograph technology. The problem of factorization of large integers and finding discrete logarithms for elements of finite groups is the main task for attacks against the algorithm.

It was designed for devices with limited compute power and memory, such as smartcards and PDAs. Elliptical curve cryptography can operate more quickly than RSA. For elliptic curve keys is not necessary to be prime, which is an ease for key generation.

Over the past few years elliptic curve cryptography has been gaining popularity and being standardized around the world by agencies as ANSI (X9.62, X9.63), IEEE group P1363 and ISO/IEC, and RSA labs and Certicom. It is proposed only with one byte to be specified the type of the elliptic curve. The U.S. National Security Agency has endorsed ECC technology by including it in its Suite B set of recommended algorithms and allows their use for protecting information classified up to top secret with 384-bit keys.

Shorter key size of ECC is an advantage. It is a reason for application of ECC in:

– Wireless communications

– Smart cards. For example in smart cards the upper limit for number of bits for the implemented key is 1024bits, that is why ECC are good alternative.

– Web applications, for example SSL.

In general where security is needed but lacks the power, for storage and computational power for the cryptosystems.

For the reasons listed above ECC is important for wireless sensor networks[8], [9].

II. MATHEMATICAL EXPLANATION

One important property of a set of solutions of an elliptic curve is that it forms a group which enables us to do cryptography. The public key is a point in the curve and the private key is a random number. The public key is obtained by multiplying the private key with the generator point G in the curve.

Elliptic curves are the set of solutions of an equation of the form $y^2 = x^3 + ax + b$. Where the coefficeints a and b are elements of the field and $4a^3 + 27b^2 \neq 0$. For different values of a, and b can be received different elliptic curves. . If the number of points on elliptic curve is marked with E, then the upper, and the lower limit for it is calculated by Hasse's theorem, and is:

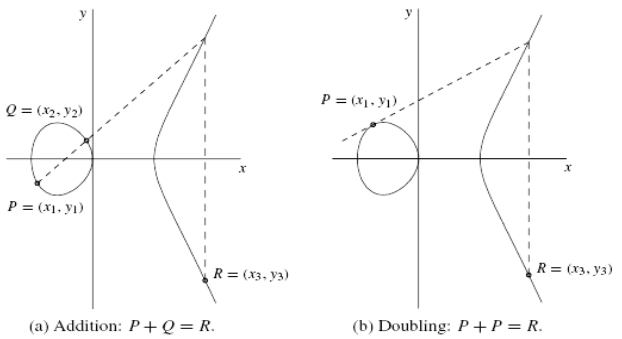
$$p + 1 - 2\sqrt{p} \leq E \leq p + 1 + 2\sqrt{p} \quad (1)$$

The elliptic curve cryptography usually is defined over two types of finite fields.

The NIST recommended [6] a certain set of elliptic curves for government use. This set of curves can be divided into two classes:

- Prime field F_p
- Binary field F_2^m

The elements of finite fields are integer numbers between 0 and p – 1, where p is representing the number of bits. All the operations such as addition, subtraction, division, multiplication are made over the integers. The prime number p is chosen such that there is finite large number of points on the elliptic curve to make the cryptosystem secure. SEC (Standards for Efficient Cryptography) [6], [7] specifies curves with p varying between 112-521 bits. In the experiments and results received in this paper is presented elliptic curve with 256 bits. The curves over F(p) are of the form [rsa lab] $y2 = x3 - 3x + b$ with b random, while the curves over F_2^m are either of the form $y2 + xy = x3 + x2 + b$ with b random or Koblitz curves. A Koblitz curve has the form $y2 + xy = x3 + ax2 + 1$ with $a = 0$ or 1.



(a) Addition: $P + Q = R$. (b) Doubling: $P + P = R$.

Fig.1. Addition and doubling of elliptic curve points

The main operations over elliptic curves are addition and doubling. Point multiplication is achieved by two basic elliptic curve operations.

- Point addition
- Point doubling

For example: if we assume m = 23 then mP can be expressed as R=23.P = 2(2(2(2P) + P) + P) + P.

If knowing two points on the curve can be received a unique third point which is the intersection of the curve with the line through the two points. If the line is tangent to the curve at a point, then that point is counted twice; if the line is parallel to the y-axis, is defined the third point as the point at infinity. Exactly one of these conditions then holds for any pair of points on an elliptic curve.

The main feature of elliptic curve cryptography is that if you have a point on the curve, all multiples of this point are also placed on the curve.

ECC are defined by the elements (p,a,b,G,n,h) defining the elliptic curve, that is called the domain parameters - a and b define the curve (by the equation); p defines the finite field F_p in the prime case - computations end by taking the remainder on division by p; G is the base point and defines the cyclic subgroup; order n represents the number of discrete points on the curve - the smallest non-negative number such that nG=O; h is called the cofactor and preferably is equal to 1.

Can be concluded that elliptic curve model is a hierarchical model of three layers:

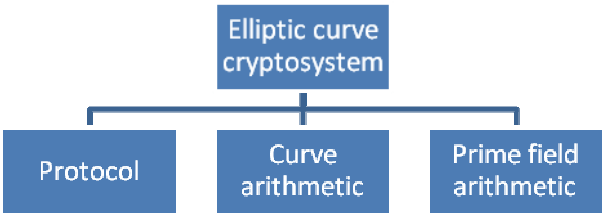


Fig.2 ECC structure

The first two layers were discussed above. On the top of the structure of ECC-system are cryptographic protocols. The protocol is a distributed algorithm for retrieving of a solution to a given cryptography task by the participants. Unfortunately between the trusted parties could be the "enemy". The aim of the protocol is predominantly to defend against unauthorized actions from the other participants. The protocols are realized for concrete aim. It is one of the fastest growing areas of theoretical cryptology, and is a field for further research.

For example a curve of the type shown bellow is used in the Microsoft Windows Media Digital Rights Management Version2.

$$y^2 = x^3 + 3176890812\ \ 5132550347\ \ 6317476413\ \ 8276932727\ \ 46955927\ x$$

$$+7905289660\ \ 7878758718\ \ 1205720257\ \ 1853543210\ \ 0651934$$

The process of the necessary amount of bits for the keys during next years is as follows:

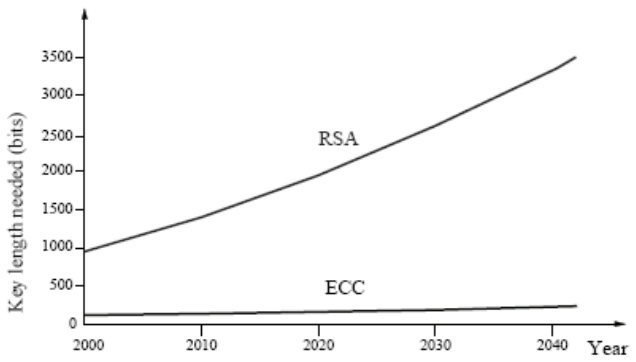


Fig 3. Compare between key length for RSA and Elliptic curves in the next few decades

The evolution in key length is according to the Moor's law. He proposed and improved that one each 18 months the number of bits increases [8], [9].

III. EXPERIMENTAL RESULTS

In this section the simulation results for the time need in key generation and the factorization are presented.

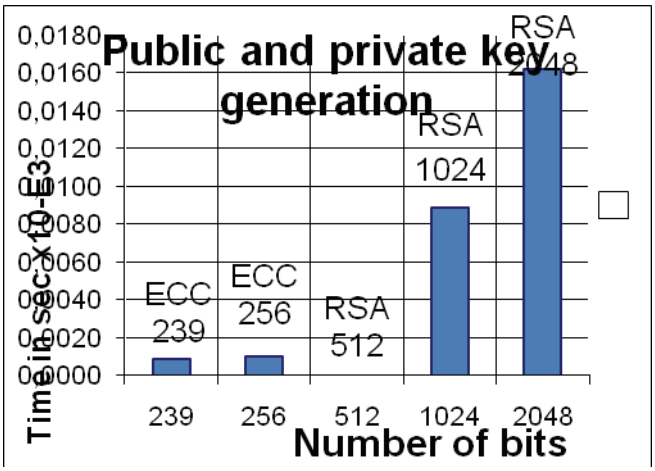


Fig5. Comparison between the time for key generation for RSA and Elliptic curves

In Fig. 5 is shown comparison between the RSA algorithms and elliptic curves implemented in cryptography and is seen the time necessary for the process of encryption. The times for the process of ECC encryption is in times smaller than the RSA even with keys of sizes of 2048 bits, which are not implemented at the moment for higher level of security (because attacks against 2048 bit keys are reported and improved to be braked).

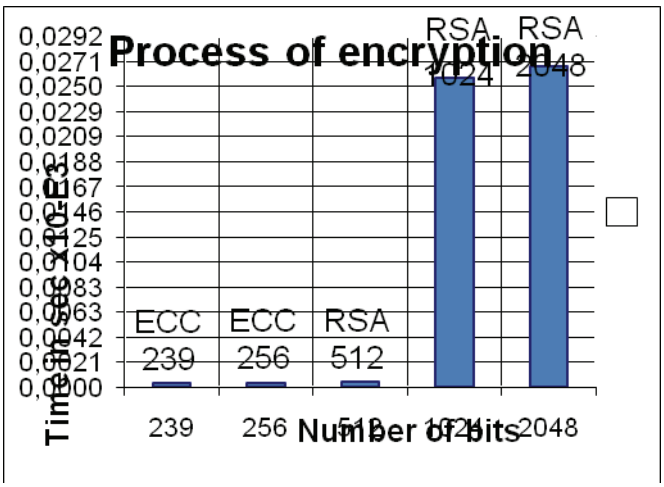


Fig6. Comparison between the times for encryption for RSA and Elliptic curves

In Fig. 6 is shown throughout the simulations comparison between times needed for generation of the keys for RSA cryptosystems and ECC system using different bit lengths. The times for the ECC239 and the ECC256 are similar to the time for RSA algorithm implementation for 512bits, which is not in use, because it is not secure.

In both graphics it can be observed that the times for generation, and encryption for the ECC crypto systems are considerably (in scales) smaller than for RSA even in the case of RSA512, that is the reason for implementation of such type of cryptography in heterogeneous networks.

In the next studied example an elliptic curve with 256 bits length is presented with all generated parameters for defining the curve as follows (the same curve is used in the previous examples):

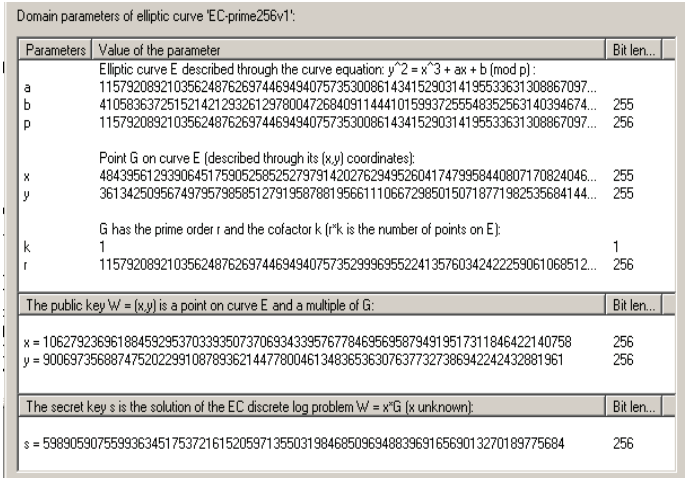


Fig. 4 Parameters of the elliptic curve

Details for the factorization of the

Input number =
77919633442827426182367047307721734498829067732826781658833934812666367

The respective next composite factor will be factorized into two factors:

1. Factorized number =
77919633442827426182367047307721734498829067732826781658833934812666367

Bit length = 239

Method: Brute Force. Time: 0.032 seconds.

First factor = 251

Bit length = 8, prime number.

Second factor =
31043678662481046287795636377578380278417955272042542493559120908871117

Bit length = 231, composite number.

2. Factorized number =
31043678662481046287795636377578380278417955272042542493559120908871117

Bit length = 231

Method: Lenstra. Time: 0.078 seconds.

First factor = 143501

Bit length = 18, prime number.

Second factor =
216330747956328153028868345012079220900327909018352084609578476 17

Bit length = 214, composite number.

3. Factorized number =
216330747956328153028868345012079220900327909018352084609578476 17

Bit length = 214

Method: Pollard. Time: 0.406 seconds.

First factor = 3293634479

Bit length = 32, prime number.

Second factor =
65681468097215699881209661399126742898153850975140974 23

Bit length = 183, prime number.

Found 4 factors in 0.516 seconds.

After generation of the public and private keys involving elliptic curve cryptology logic an example of factorization according to different methods is implemented. In the results a brute force attack is presented which is not the appropriate way for breaking the system private and public keys. A limit during the experiments is applied only for the first received factor, to be less than 1000.

IV. Conclusion

The security of a cryptosystem based on elliptic curves according to our experiments depends on the task how difficult it is to determine m given mP and P. This is referred to as the elliptic curve logarithm problem. The fastest known technique for taking the elliptic curve logarithm is known as the Pollard method and in one of our experiments we use this technique. It has been seen that a considerably smaller key size can be used for ECC compared with RSA, and DSA with curves is ten times faster than the classical RSA.

The most secure codes currently in use rely on public-key cryptography, whose security is based on the fact that computers today cannot factor very large numbers within a reasonable time period.

ACKNOWLEDGMENT

This work is supported by the Technical University of Sofia Science Fund – Grant No. 102ni191-7/2010 „Investigation of adaptive methods for QoS in communication networks”.

References

[1] Diffie, W., and Hellman, M. New directions inn cryptography, Trans. Inform. Theory IT-22, Nov. 1976, 644-654.
[2] R. L. Rivest. "Cryptology" in Handbook of Theoretical Computer Science, vol. A: Algorithms and Complexity, Elsevier and MIT Press (1990), 717-756.
[3] V.S. Miller, Use of elliptic curves in cryptography, Advances in Cryptology - Crypto '85, Springer-Verlag (1986), 417-426.
[4] T. El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory, v. IT-31, n. 4 469-472, 1985.
[5] http://www.nsa.gov.shtml
[6] FIPS 186-2
[7] According to http://www.secg.org/download/aid-385/sec1_final.pdf
[8] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.
[9] Arjen K. Lenstra, Eric R. Verheul, “Selecting Cryptographyc Key Sizes” (1994). Journal of Cryptology of International Association for Cryptologyc Research