Annals of University of Craiova, Math. Comp. Sci. Ser.
Volume 34(1), 2007, Pages 94–99
ISSN: 1223-6934

Authentication ranks with identities based on elliptic curves

Nicolae Constantinescu

Abstract. The giving paper treats the way in which a symmetric cryptographic key, which has been acknowledged by the two counterparts on a communications network, is being created. It will be treated the manner in which the two will authentify their selfs, by using secret information and arbitrary strings, which can be the names of the two. On these strings other data that will be added, competing for the final definition of the key. It will also consider the case in when we can use the access information's hierarchy.

2000 Mathematics Subject Classification. Primary 14H52; Secondary 94A60.
Key words and phrases. elliptic curves, public key cryptography, identity based cryptography.

1. Introduction

In order to facilitate the assurance of confidentiality regarding the communication, the requirements imposed by the following three basic steps must be fulfilled. The steps are:

(1) The making and distribution of a master key,
(2) The agreement on a protocol which will create a session key, used by the communicating parts for the current message session,
(3) The encryption algorithm based on the participant's acknowledged key.

The system is viable for the case in which we have an encryption, with a symmetric key, of the messages and an agreement on the session key which is based on public keys. The founding idea of the identities system belongs to Shamir [5] He wanted to create a method which will secure the e-mail. If Ann and Bob are two persons who want to communicate, each one has an e-mail address. When Ann wants to send a message to Bob, she will make a request to the server which holds Bob's public key and will encrypt the information with it. The server will manage both the public and the secret key of the participants at the dialog. The key request will be composed by sending the server a request which contains the e-mail address for whom the public key is wanted. The modifications introduced by Shamir consists of a function, χ, which generates a public key from an arbitrary string (e-mail address). In this way Ann will not ask for the key from the server any more, she will encrypt messages using the key generated by function χ. The benefits from this system is the fact of one can encrypt and send a message to another even if the Password Server is not available in that moment, and in this way the talk between Ann and the Server is not needed anymore. Throughout time, diverse algorithms which relay on this system as a base have been proposed, [4, 6], but not many of them can be put into practice because the implementations impose too heavy burden on the current calculus systems.

Received: 03 January 2008.

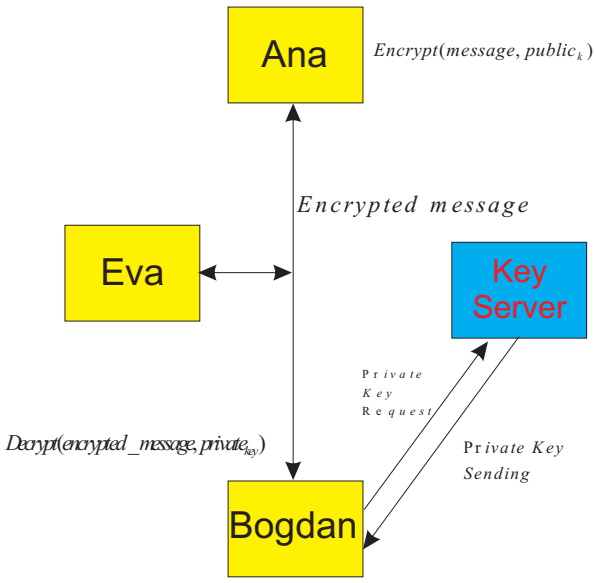

Figure 1

2. The description of the secured system, based on identities

In [2], myself and Professor dr. George Stephanides, have elaborated an encryption system based on identities. The system is composed by the following parts:

(1) *Setting up the system.*
For everyone of the participant in the intercommunications group, a control key is assigned (named ID) This key is used to ensure the communication between the Password Server and the user. We shall name the Server Public Key Generator, or PKG for short.

(2) *The encryption.*
A participant (called A), who wants to communicate with another participant (called B), will encrypt the sending message with a public key (called p_k), obtained from the morph which applies function χ, on a String s which has in it's continence an identifier of B; identifier which can be the e-mail address of B.

(3) *The decryption.*
A system user who will receive a encrypted message will access the Key Server, and due to his ID, will obtain the private key needed to decrypt the message.

The way in which this system works is illustrated in Figure 1. In the case of an eavesdrop, the eavesdropper will have access at the encrypted message.

2.1. The elliptic curves based system. For the following integration

$$\int \frac{dx}{\sqrt{4x^3 - g_2 x - g_3}} \tag{1}$$

The inverse function is called an elliptic curve. If γ_1 şi γ_2 are two constants, a function in a double period over R, then the Weierstrass function will be of the form:

$$(\alpha')^2 = 4\alpha^3 - \gamma_1\alpha - \gamma_2 \tag{2}$$

The pair (α, α') will define in space a point on the curve

$$y^2 = 4x^3 - \gamma_1 x - \gamma_2 \tag{3}$$

Definition 2.1. *If $p > 3$ is a prime integer. The elliptic curve $y^2 = x^3 + \gamma_1 x + \gamma_2$, defined over Z_p, represents a set of solutions having the form $(x, y) \in Z_p \times Z_p$ conform with the congruence relation*

$$y^2 \equiv x^3 + \gamma_1 x + \gamma_2 \ (mod\ p) \tag{4}$$

where the coefficients $\gamma_1, \gamma_2 \in Z_p$ are constants which satisfied the relation

$$4\gamma_1^3 + 27\gamma_2^2 \not\equiv 0 \ (mod\ p) \tag{5}$$

together with a special O named point to infinity.

Lemma 2.1. *If E is an elliptic curve given by the following relation*

$$Y_2 + \gamma_1 XY + +\gamma_3 Y = X^3 + \gamma_2 X^2 + \gamma_4 X + \gamma_6 \tag{6}$$

and $A_1 = (x_1, y_1)$, $A_2 = (x_2, y_2)$ two points on the curve. Then

$$-A_1 = (x_1, -y_1 - \gamma_1 x_1 - \gamma_3) \tag{7}$$

and

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1},\ \gamma = \frac{y_1 x_2 - y_2 x_1}{x_2 - x_1} \tag{8}$$

where x_1, x_2 satisfy the relation and $x_1 \neq x_2$ and, from here results

$$\lambda = \frac{3x_1^2 + 2\alpha_2 x_1 + \alpha_4 - \alpha_1 y_1}{2y_1 + \alpha_1 x_1 + \alpha_3},\ \gamma = \frac{-x_1^3 + \alpha_4 x_1 + 2\alpha_6 - \alpha_3 y_1}{2y_1 + \alpha_1 x_1 + \alpha_3}. \tag{9}$$

In case of equableness between x_1 and x_2, the points A_1 and A_2 being different from each other , will result the addition of the two points with the coordinates

$$x_3 = \lambda^2 + \alpha_1 \lambda - \alpha_2 - x_1 - x_2,\ y_3 = -(\lambda + \alpha_1) x_3 - \gamma - \alpha_3 \tag{10}$$

In compliance with this Lemma, will result the two cases:

(1) $x_2 = x_1$ and $y_2 = y_1$. In this case $A_1 + A_2 = O$
(2) For every other cases $A_1 + A_2 = B$, $B(x_3, y_3)$ where

$$x_3 = \lambda^2 - x_1 - x_2, y_3 = \lambda(x_1 - x_3) - y_1 \tag{11}$$

and

$$\lambda = \begin{cases} (y_2 - y_1)(x_2 - x_1)^{-1}, A_1 \neq A_2; \\ (3x_1^2 + a)(2y_1)^{-1}, A_1 = A_2; \end{cases} \tag{12}$$

Regarding the implementations of an information access hierarchy, a function which creates a public key must be defined. The key will be based on the conjugated information, from the concordant hierarchy layer for each user's personalized string for the communication channel. If φ is a function that applies to these rules:

$$\varphi(level, string) = \text{public key} \tag{13}$$

where *level* will represent the access level of an user and *string* represents a string of characters which characterizes the dialog's participants. The way in which an hierarchy for some users is being illustrated in Figure 2 . The basic property is to respect the access rights to information by the users of the same level. For each member of a network will be defined a point $A_i^j \in E$, where j represents the hierarchy level of each user. For each session related to the obtainment of a private key from the key server, a Diffie-Hellmann algorithm, related with the one described in [3], will be used. Trialling the emergence of a session key, a private key will be obtained. We must not forget the security issues described in [1].

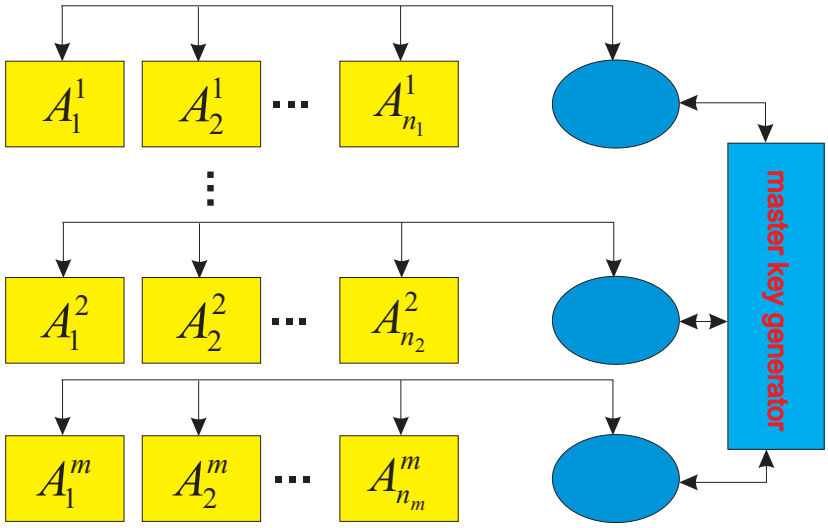

Figure 2

3. Security of elliptic curves operations

Security analysis often uses a model in order to test side-channel information leakage. While it is practically impossible to analyse all possible information leakage, often this model is aimed at specific aspects, configurations or implementations.

Before presenting the method, information leakage is considered at a lower level. Subsection 3.1 discusses special cases of point operations that should be avoided. Subsection 3.2 discusses the importance of using randomised projective coordinates and certain extended point representations. Also this subsection questions the insertion of dummy point additions to achieve uniform behaviour.

3.1. Point operations. Side-channel analysis is the operation which involves gathering of information concerning timing of computations and power consumption of such operations. Values that these indicators provide may be interpreted to obtain a certain order of operations involved in a cryptosystem. To conceal this order a careful devised algorithm should use point doubling and addition operations in order to create a uniform pattern which should be independent of the specific multiplier used in considered operations. Of course there are exceptions, and these situations should be treated in a special manner at the time of their occurrence. These are presented in the following statements:

- Point doubling $[2]A$ requires conditional statements for the case that A is the point at infinity or that A is a point of order two. If these cases are avoided, then, expressed in field operations, point doubling runs as a fixed routine.
- Point addition $A+B$ requires conditional statements for the case that one of the points is the point at infinity, or that A coincides with B, or that one point is the inverse of the other. For other cases, it too can be implemented as a fixed routine.

Details of the sequences of field operations used for point doubling and point addition depend on the underlying field (odd characteristic vs. characteristic 2) and the choice of point representations (e.g. either affine coordinates or one of multiple styles of projective coordinates, so implementations may vary widely. The essential

observation is that the respective algorithm always behaves the same as long as the above special cases are avoided.

3.2. Field operations. An important observation is that when an attacker analyses the side-channel information he does not have immediate access to the factors involved in a operation. However it is prudent to say that not all operations look the same, and this is the basic idea for side-channel attacks. Inserting randomisation techniques into one's protocol or any other cryptosystem based on elliptic curves is a good idea. This is combined with the usefulness of projective coordinates. Take for example Jacobian projective coordinates, which are triplets of the form (X, Y, Z) with $Z = 0$, they represent affine points $(X/Z^2, Y/Z^3)$; then for any field element $\epsilon \neq 0$,$(\epsilon^2 X, \epsilon^3 Y, \epsilon Z)$ is a representation of the same point on the curve. Randomisation makes it difficult for an attacker to guess the values obtained by using a randomly chosen ϵ.

Point doubling or point addition using projective coordinates results in a point represented with a Z-coordinate that is the product of the Z-coordinate(s) of the input point(s) and a short polynomial involving one or more other coordinates of the input points; thus the output point is again in a randomised representation.

Randomisation makes it difficult for an attacker to guess or imply that a certain operation involving known points is taking place at a certain point in time. Still the attacker may observe the same operation reoccuring if the same field operation is execute several times throughout the computation. Even if the attacker cannot obtain the factors involved in the operation we still want to mask this as this in some cases may be considered an important information leakage.

Point multiplication, $R = [e]P$, is performed in stages by a great part of existing algorithms, these stages are as follows:

Precomputation stage:: First, independently of the specific multiplier e, certain small multiples of P are computed and stored in a table.

Evaluation stage:: Second, the product $[e]P$ is evaluated as follows: A variable A is initialized to one of the table values; then, many times, A either is doubled or a table value is added to A, replacing the previous value of A. Finally, A contains the result $[e]P$.

Algorithms that adhere to this structure should use a table to store the value of Z^2 by using a tuple of the form (X, Y, Z, Z^2) and store it in the table, instead of the classical approach of storing (X, Y, Z) and then perform the precomputation phase when needed.Storing the tuple (X, Y, Z, Z^2) is called using *extended point representation*, if the usage of the suggested method of extended point representation and is neglected an attacker may be able to tell which point additions use the same entries in the table.

Another problem with algorithms that follow the above structure is that they need to insert dummy additions in their structure in order to achieve a fixed pattern of doublings and additions. These additions involve additions of a table value to the variable A, and then discarding the result. The main issue with dummy additions is that the value of A is never changed, this becomes a major problem when working with Jacobian projective coordinates. This is because each point operation requires squaring the Z-coordinate of A, if two consecutive point operations are performed the same value of Z will be used leading to the same squaring being performed. This is why dummy operations should be avoided.

It is possible, but inefficient, to randomise the point representation after each point operation. (If this is done, dummy additions are no longer a problem.) For point multiplication algorithms of the above form, it is more practical to use randomisation

just twice or once: If the precomputed table of multiples of P is stored in projective coordinates, then the representation of P should be randomised before the table is computed. Also at the beginning of the second stage, after the initial value has been assigned to A, the representation of A should be randomised. If the table of multiples of P is stored in affine coordinates (to speed up the evaluation stage by using mixed addition of affine and projective points), then the first randomisation obviously is not necessary.

4. Conclusions

This article presents the way in which hierarchy for the access to information in an user group on a communication channel is being shaped into a layered structured. The problems which occur in this kind of system are related to the length of the messages exchanged between users, based on the string structure. In fact, a *time-stamp* signature is used in practice, but this *time-stamp* it is not possible in all the cases, because communication in large broadband a larger then the processing power offered by systems viable from the price/performance point of view. The proposed model has an amplification factor (χ) for the number of valid messages that can be send with the same user's authenticity string. χ relays on the communication frequency between a $KeyGenerator$ and the control points of each level.

Acknowledgements. The author has been supported by CNCSIS Grant 79/2007.

References

[1] N. Constantinescu, The agreement of the common key, *The Annals of the University of Craiova - Mathematics and Computer Science series*, Vol. XXX, 2004

[2] N. Constantinescu, G. Stephanides, Identification of parts in identity-based encryption, *Research Notes in Data Security, Wessex Institute of Technology*, UK, developed with University of Bergen, Norway, ISBN 1-85312-713-2, 2004

[3] N. Constantinescu, George Stephanides, Secure Key-Exchange, *Recent Advances in Communications and Computer Science* , Vol. 7,pp. 162-166, WSEAS Press, Greece, 2003

[4] A. Miyaji, M. Nakabayashi, S. Takano, New explicit condition of elliptic curve trace for FR-reduction, *IEICE Trans. Fundamentals*, Vol. E84 A, No. 5, May 2001

[5] A. Shamir, Identity-based cryptosystems and signature schemes, *Advances in Cryptology*, LNCS, Vol 196, Springer-Verlag, pp. 47-53, 1984

[6] H. Tanaka, A realization scheme for identity-based cryptosystem, *Advances in Cryptology*, LNCS, Vol 293, Springer-Verlag, pp. 341-349, 1987

(Nicolae Constantinescu) Department of Mathematics, University of Craiova,
Al.I. Cuza Street, No. 13, Craiova RO-200585, Romania, Tel. & Fax: 40-251412673
E-mail address: nikyc@central.ucv.ro