# An extension of the El Gamal encryption algorithm

Stelian Flonta, Dr. Liviu Miclea
Universitatea Tehnică Cluj-Napoca
Email: sflonta@colim.ro Liviu.Miclea@aut.utcluj.ro

## 1. Abstract

The encryption algorithms generally allow the encryption of a whom message by one person, we call Alice, and the decryption of the message encrypted by another person, with the generic name Bob. The algorithm presented in this paper work , is based on the following scheme:

Alice is encrypting a message which she is simultaneously sending to the persons $Bob_1$, $Bob_2$, …, $Bob_{2n+1}$ . The 2n+1 persons ( $Bob_1$, $Bob_2$, …, $Bob_{2n+1}$ ) will be able to decrypt the message received from Alice only if they are together, separately this operation being impossible for them.

## 2. Introduction

The present paper work presents an extension of the El Gamal encryption algorithm in the sense that, starting from it, a mathematical model was elaborated so that the decryption of a message can be made only by an authorized group. Here are presented: the pattern of the initial algorithm, the extended variant, two examples and three ideas of practical applications. The extendend variant is entitled the El Gamal encryption algorithm with divided private Key, suggesting the partition of the private Key, and these segments being assigned individually to the members of the authorized group.

## 3. The El Gamal encryption algorithm

3.1. The algorithm has a public Key and it can be divided into three parts : *Key generation, encryption of a message and decryption of the message* .

Within the framework of the operations that follow, in order to simplify the expression, it will be calculated modulo q without writing explicitly, where q is the one chosen in the Key generation stage. For instance $h=g^x$ means $h=g^x \pmod q$.

**1-4244-2577-8/08/$20.00 ©2008 IEEE**

3.1.1. *Key generation*

It is chosen a cyclic group of order q prime number, for which the discrete logarithm problem is difficult, and g as a generator. It is chosen a random x from {0,…,q-1} and then it is computed $h=g^x$

The public Key is {q,g,h} and the private Key is {x}.

3.1.2. *Encryption of a message*

Alice encrypts message m knowing the public Key as it follows: chooses a random element y from {0,…,q-1} and computes $c_1=g^y$, $c_2=m \cdot h^y$ then sends the encrypted message $(c_1, c_2)$ to Bob

3.1.3. *Decryption of the message*

In order to decrypt the message $(c_1, c_2)$ Bob uses q and the private Key {x} computing

$c_2/c_1^x=m \cdot h^y / (g^y)^x=m \cdot g^{xy} / g^{yx}=m$

The system is obviously indefinite: the encryption depends on $x\psi$ and another aleatory value y , chosen by *Alice*. Thus, there are more encrypted texts corresponding to a certain clear text.

3.2. **An example of encryption using the El Gamal algorithm**

*Key generation*

Let q=1013, g=610, x=319, and $h=g^x(\text{mod } 1013)=377$, thus {1013,610,377} is the public Key and {319} is the private Key.

*Encryption of a message*

Alice receives the public Key {1013,610,377} and wants to encrypt number **m**=560, for this she chooses y=335 and calculates

$c_1=g^y(\text{mod } 1013)=533$

$c_2=m \cdot h^y(\text{mod } 1013)=560 \cdot 46(\text{mod } 1013)=435$

and consenquently she obtains the encrypted message

$(c_1, c_2)$

*Decryption of the message*

Bob receives the encrypted message $(c_1, c_2)$, and in order to decrypt it he calculates

$c_2 / c_1^x(\text{mod } 1013)=435/46(\text{mod } 1013)=435 \cdot 991(\text{mod } 1013)=560=\mathbf{m}$

## 4. The El Gamal encryption algorithm with divided private Key

4.1. The algorythm can be divided in such way that an expeditor can send an encrypted message to more than one recipients and each recipient

also obtains a private Key that only he knows. As a consequence of the division of the private Key to $2n+1$ persons, these persons can decrypt the received message only if they are together .

Next it follows the presentation of the three stages of the algorithm. Within the framework of the operations that follow, in order to simplify the expression, it will be calculated modulo q without writing explicitly, where q is the one chosen in the Key generation stage. For instance $h=g^x$ means $h=g^x \pmod q$.

4.1.1. *Key generation*

It is chosen a cyclic group of order q prime number, for which the discrete logarithm problem is difficult, and g as a generator. From

$\{0,\ldots,q-1\}$ there are chosen the elements $x_1,x_2,\ldots,x_{2n+1}$, preferably distincts, then there are being calculated $h_1=g^{x1}$, $h_2=g^{x2},\ldots$, $h_{2n+1}=g^{x2n+1}$

The public Key is $\{q,g,h_1,h_2,\ldots,h_{2n+1}\}$ and the private Key is which $\{x_1,x_2,\ldots,x_{2n+1}\}$ from which are chosen $\{x_1\},\{x_2\},\ldots,\{x_{2n+1}\}$ these being the private Keys for $Bob_1$, $Bob_2$, and respectively $Bob_{2n+1}$ . Obviously each one of them knows only his own private Key .

4.1.2. *Encryption of a message*

Alice encrypts message m knowing the public Key as it follows: chooses a random element y from $\{0,\ldots,q-1\}$ and calculates

$c_1=g^y$,

$c_{21}=m \cdot h_1^y$, $c_{22}=m \cdot h_2^y,\ldots$, $c_{22n+1}=m \cdot h_{2n+1}^y$,

$c_2= c_{21} \cdot c_{23} \cdot c_{25} \cdot c_{27}\ldots./ c_{22} \cdot c_{24} \cdot c_{26}\ldots.$ then sends the encrypted message $(c_1, c_2)$ to the recipients $Bob_1, Bob_2, \ldots, Bob_{2n+1}$

4.1.3. *Decryption of the message*

In order to decrypt the message $(c_1, c_2)$ $Bob_1, Bob_2, \ldots, Bob_{2n+1}$ are using q and the private Keys $\{x_1\}$, $\{x_2\}$, respectively $\{x_{2n+1}\}$, computing together

$c_2 \cdot c_1^{x2} \cdot c_1^{x4} \cdot c_1^{x6} \ldots / c_1^{x1} \cdot c_1^{x3} \cdot c_1^{x5} \cdot c_1^{x7}\ldots=$

$(c_{21} \cdot c_{23} \cdot c_{25} \cdot c_{27}\ldots / c_{22} \cdot c_{24} \cdot c_{26}\ldots)( c_1^{x2} \cdot c_1^{x4} \cdot c_1^{x6}\ldots / c_1^{x1} \cdot c_1^{x3} \cdot c_1^{x5} \cdot c_1^{x7}\ldots)=$

$(m \cdot h_1^y \cdot m \cdot h_3^y \cdot m \cdot h_5^y \cdot m \cdot h_7^y\ldots / m \cdot h_2^y \cdot m \cdot h_4^y \cdot m \cdot h_6^y\ldots)( c_1^{x2} \cdot c_1^{x4} \cdot c_1^{x6}\ldots / c_1^{x1} \cdot c_1^{x3} \cdot c_1^{x5} \cdot c_1^{x7}\ldots)=m.$

The preceeding calculations are computed knowing that:

- $h_i^y = g^{x_i y} = g^{y x_i} = c_1^{x_i}$ , for i=1,…,2n+1
- The first fraction has n+1 factors as numerators and n factors as denominators.
- The second fraction has n factors as numerators and n+1 factors as denominators.
- The model is conceived for the division of the private Key into an uneven number of keys, it can be easily adapted to an even number of users through the assigning of two keys resulted by division to a single receiver.

4.2.1. **An example of encryption using the El Gamal algorithm with key divided by three**

*Key generation*

Let q=71, g=31, $x_1$=15, $x_2$=19 şi $x_3$=17 and $h_1=g^{x1}$ (mod 71)=41, $h_2=g^{x2}$ (mod 71)=61 , $h_3=g^{x3}$ (mod 71)=67 thus {71,31,41,61,67} is the public Key and {15,19,17} is the private Key

*Encryption of a message*

Alice receives the public Key {71,31,41,61,67} and wants to encrypt number **m**=30, for this she chooses y=10 and computes

$c_1=g^y$(mod 71)=20

$c_{21}=m\cdot h_1^y$(mod 71)=$30\cdot 41^{10}$(mod 71)=32

$c_{22}=m\cdot h_2^y$(mod 71)= $30\cdot 61^{10}$ (mod 71)=48

$c_{23}=m\cdot h_3^y$(mod 71)= $30\cdot 67^{10}$ (mod 71)=20

$c_2= c_{21}\cdot c_{23} / c_{22}$=32 · 20 / 48=37

and consenquently she obtains the encrypted message $(c_1, c_2) = (20 , 37)$

*Decryption of the message*

$Bob_1$, $Bob_2$ and $Bob_3$ receive the encrypted message $(c_1, c_2)$, and in order to decrypt it they compute using the private Keys $\{x_1\}$, $\{x_2\}$ şi $\{x_3\}$,

$(c_2\cdot c_1^{x2}) / ( c_1^{x1} . c_1^{x3} )$**(mod 71)=** $(37\cdot 30) / ( 20 \cdot 48 )$(mod 71)=45/37=

45 · 48=**30 =m**

4.2.2. **An example of encryption using the El Gamal algorithm with key divided by five**

*Key generation*

Let q=701, g=131, $x_1$=15, $x_2$=19 , $x_3$=17, $x_4$=37 , $x_5$=68

and

$h_1=g^{x1} \pmod{701}=131^{15} \pmod{701}=120$

$h_2=g^{x2} \pmod{701}=131^{19} \pmod{701}=139$

$h_3=g^{x3} \pmod{701}=131^{17} \pmod{701}=483$

$h_4=g^{x4} \pmod{701}=131^{37} \pmod{701}=201$

$h_5=g^{x5} \pmod{701}=131^{68} \pmod{701}=407$

thus {701, 131, 120, 139, 483, 201, 407} is the public Key and {15,19,17,37,68} is the private Key.

*Encryption of a message*

Alice receives the public Key {701, 131, 120, 139, 483, 201, 407} and wants to encrypt number **m**=50, for this she chooses y=210 and computes

$c_1=g^y \pmod{701}=638$

$c_{21}=m\cdot h_1^y \pmod{701}=50\cdot 120^{210} \pmod{701}=50$

$c_{22}=m\cdot h_2^y \pmod{701}=50\cdot 139^{210} \pmod{701}=244$

$c_{23}=m\cdot h_3^y \pmod{701}=50\cdot 483^{210} \pmod{701}=67$

$c_{24}=m\cdot h_4^y \pmod{701}=50\cdot 201^{210} \pmod{701}=67$

$c_{25}=m\cdot h_5^y \pmod{701}=50\cdot 407^{210} \pmod{701}=686$

$c_2=((c_{21}\cdot c_{23}\cdot c_{25}) \pmod{701} / (c_{22}\cdot c_{24}) \pmod{701}) \pmod{701} =$

$((50\cdot 67\cdot 686) \pmod{701} / (244\cdot 67) \pmod{701}) \pmod{701} =$

$(222/225) \pmod{701} = (222\cdot 620) \pmod{701} = 244$ and consenquently she obtains the encrypted message $(c_1, c_2)=(638, 244)$

*Decryption of the message*

$Bob_1$, $Bob_2$, $Bob_3$, $Bob_4$ and $Bob_5$ receive the encrypted message $(c_1, c_2)$, and in order to decrypt it, they compute using the private Keys $\{x_1\}$, $\{x_2\}$, $\{x_3\}$, $\{x_4\}$ and $\{x_5\}$,

$((c_2\cdot c_1^{x2}\cdot c_1^{x4}) \pmod{701} / (c_1^{x1}\cdot c_1^{x3}\cdot c_1^{x5}) \pmod{701}) \pmod{701}=$

$((244\cdot 638^{19}\cdot 638^{37}) \pmod{701} / (638^{15}\cdot 638^{17}\cdot 638^{68}) \pmod{701}) \pmod{701}=$

$((244\cdot 89\cdot 464) \pmod{701} / (1\cdot 464\cdot 210) \pmod{701}) \pmod{701}=$

$(50/1) \pmod{701}=50=\mathbf{m}$

## 5. Possible applications

5.1. The transmission of an exam subject to the examining board.
It is possible that a subject can be delivered to an examining board via e-mail, in secured conditions by a person, who formulates the subject, as it follows:

- The authority that rules the activity, generates the keys and assigns them, throughout a public channel or a secured, one depending on the case, to those who formulate the subject and to the members of the examining board.
- The person who formulates the subject encrypts the message and sends it to the examining board.
- The examining board decrypts the message only if they are together.

5.2. The validation of the presence of all the members by right to a virtual conference.

The speed governor of a conference generates the keys sending the secret ones to the authorized members to participate at the conference. When a session is convoqued the speed governor encrypts a message, known only by him, that he sends to the members by right. These ones are decrypting the received message together and return it to the speed governor . If the initial message coincides to the returned message , it means that all members by right are present and the conference can begin

5.3. The validation of the entire progress of a trial .
A set of keys is being generated and associated to each stage of a trial, a secret key also to the whole trial and an encrypted message by the trial manager. During the progress of the trial, partial calculations of decryption are being made. The trial is over when the result of the calculation coincides to the initial message.

## **6.** *References*

[1] A. Menezes, P. Oorschot, S. Vanstome, Handbook of Applied Cryptography. CRC Press, 1996.
http://www.cacr.math.uwaterloo.ca/hac/
[2] A. Salomaa, Criptografie cu chei publice, Ed. Militară, 1996
[3] http://www.galaxyng.com/adrian_atanasiu/cript.htm
[4] http://en.wikipedia.org/wiki/ElGamal_encryption