ElGamal:Public-Key Cryptosystem

Jaspreet Kaur Grewal

A paper presented for the degree of Master of Science

Math and Computer Science Department
Indiana State University
Terre Haute,IN,USA
9/30/2015

Contents

Abstract

Key exchange is any technique in cryptography by which cryptographic keys are exchanged between two parties, permitting utilization of keys in a cryptographic algorithm. The key exchange issue is the manner by which to exchange whatever keys or other data are needed so that nobody else can obtain a copy. The public key cryptography, which uses a pair of cryptographic keys, a public key and a private key. The private key is kept secret, while public key can be distributed openly, in this way refuting the need to transmit a secret key ahead of time. This paper presents ElGamal System which is a public key cryptosystem based on the Discrete-log problem. This scheme is known as ElGamal cryptosystem, it modifies the Diffie-Hellman protocol with the goal so that it can be used as an encryption and decryption protocol. Its security is also based on the difficulty of the DLP. The security of both systems depends on the trouble of figuring discrete logarithms over finite fields. To secure against mathematical and brute-force attack as well as Low-Modulus and Known-Plaintext attack on ElGamal , we go for adjusted Elgamal cryptosystem algorithm. With the wide use of ElGamal digital signature scheme, its security is usually being challenged.

ElGamal:Public-Key Cryptosystem

Jaspreet kaur grewal

29 September 2015

1 Introduction

Cryptography is a science with history that is as old as the human's knowledge of writing.There was dependably a need to cover up essential data, to make it readable only for a certain circle of individuals. The vital piece of encrypting and decoding a cipher is knowing a key- a parameter that decides the functional yield of a cipher. Without a key, the calculation would have no outcome. A key determines the change of plaintext into ciphertext, or the other way during decryption. Presently it was possible to make a safe cipher and send it to the recipient without one to one meeting, without utilizing an additional safe channel. Nonetheless, there still was no real way to securely send the key - in the event that it got to the eavesdroppers hands, information was easily decrypted.

The situation changed in an groundbreaking year 1976 when Whitfield Diffie and Martin Hellman distributed a paper where they proposed the thought of public key cryptography in which two diverse yet scientifically related keys are used-a public key and a private key. A public key framework is constructed to the point that figuring of one key (the 'private key') is computationally infeasible from the other (the'public key'), despite the fact that they are essentially related. Rather, both keys are produced secretly, as an interrelated pair.

Diffie and Hellman distributed the first public-key algorithm known as a "Diffie-Hellman key exchange" that year, at long last making exchange of the keys genuine and secure.Public key cryptographic algorithm here implies that sender encrypt with its receiver's public key and receiver decode with its own private key.The purported public key cryptosystem utilizes distinctive keys as a part of encryption and decryption, which is a cryptosystem taking into account the infeasibility in count of finding decoding key from the known encryption key.Then, after Diffie-Hellman, one of the the most well-known public key cryptographic algorithm came: the ElGamal Algorithm.

The ElGamal algorithm is used as a part of the free GNU Privacy Guard Software, late forms of PGP, and different cryptosystems. Additionally, the Digital Signature Algorithm (DSA) variant, in view of the ElGamal algorithm (called the ElGamal signature scheme), is used to sign digital documents.The ElGamal cryptosystem includes three major processes: the key generation, the encryption, and the decryption. Let us a chance to think about that as a sender called Alice needs to send a private message to the recipient Bob, and a third individual called Eve tries to know this message. The ElGamal PKC procedure works as follows: In the first step, Bob (Receiver) has to compute a public key and send it to the Alice(Sender).After receiving the Bob's public key she do the Encryption,which comprises of computing the ciphertext from the plaintext .She sends this ciphertext to Bob.Then Bob decrypts the ciphertext to compute a plaintext using his private key.In this whole process,Eve(third person) could not be able to know the secret key because its security is based on solving the Discrete Logarithm Algorithm. So,ElGamal PKC is one of many algorithms that utilizes randomization in the encryption process.

2 History

The problem of key distribution was solved in 1976 by two researchers at stanford university,Whitfield Diffie and Martin Hellman.They proposed a cryptosystem in which the encryption key and the decoding key were different. This way to deal with cryptography, known as public key cryptography, utilizes a pair of cryptographic keys,a public key and a private key. The private key is kept secret, while the general public key can be dispersed openly, in this way invalidating the need to transmit a secret key in advance.The keys are connected numerically, permitting the sender of a message to encrypt his message utilizing the receiver's public key. The message can then just be decoded utilizing the recipients's private key.Although the Diffie-Hellman Key exchange algorithm gives a technique for openly sharing an arbitrary secret key,it does not accomplish the full objective of being a public key cryptosystem, since a cryptosystem grants exchange of particular data, not only a random string of bits.So, after the Diffie-Hellman key exchange and PKC ,Taher ElGamal in 1985 described the ElGamal cryptosystem algorithm in a type of public key cryptosystem which is used over finite fields and its security is based on the Discrete Logarithm Problem(DLP). The ElGamal Cryptosystem is a very successful implementation of Diffie-Hellman algorithm Because ElGamal algorithm can be used to encrypt in one dimension without the need of second party to take actively part.

3 ElGamal Public key Cryptosystem

3.1 Definition:Cryptosystem

We characterize the cryptosystem as the 5-tuple (M, C, K, E, D) where $M \in \sum^{\star}$ is the plaintext message Alice wants to transmit to Bob. The plaintext message can be split in numerous blocks indicated as m_i. The ciphertext $C \in \sum^{\star}$ is the result of an encryption function $E_k(P) : C$, which takes the plaintext and an extra key K, and computes the ciphertext from it utilizing a satisfactory algorithm. Bob, who gets C, utilizes it with a decryption function $D_K(C) : P$, which under use of the key K results in the plaintext. The message has been transferred, and Eve was not able to read it. So, far we were using the only shared key K But for a public key cryptosystem we need to split the key into two parts. The main part is the one which gave the system its name: the public key. The second part is the private key. Since this alleged "keypair" is customized, we will mean the private key with a persons or substances introductory letter in lower case. Alice in this way claims the private key a.The public key will be written as it is figured, leaving the modulus operation implizit. Subsequently, Alice has an public key recognized by g^a. Both the private and public key are strongly connected and in this way called the "keypair".

3.2 What is ElGamal Cryptosystem

In 1984 Taher ElGamal introduced a cryptosystem which depends on the Discrete Logarithm Problem.The ElGamal encryption system is an asymmetric key encryption algorithm for public-key cryptography which is based on the Diffie-Hellman key exchange.ElGamal depends on the one way function, means that the encryption and decryption are done in separate functions.It depends on the assumption that the DL can't be found in feasible time, while the reverse operation of the power can be computed efficiently .

3.3 What was the need of ElGamal ?

The first public key system proposed by Diffie and Hellman requires association of both sides to compute a common private key. This poses issues if the cryptosystem should be applied to communication system where both sides are not able to interact in reasonable time because of deferrals in transmission or inaccessibility of the receiving party. Means that the proposed scheme by Diffie and Hellman is not a general purpose encryption algorithm as it can only provide secure secret key exchange. Thus it presents a challenge for the

cryptologists to design and provide a general purpose encryption algorithm that satisfies the public key encryption standards.So, after Diffie-Hellman, RSA public key cryptosystem came.AFter RSA, The ElGamal solved the Diffie-Hellman key exchange algorithm by presenting a random exponent type k. This exponent is a replacement for the private type of the receiving entity. Because of this simplification the algorithm can be utilized to encode in one heading, without the need of the second party to take effectively part. The key development here is that the algorithm can be utilized for encryption of electronic messages, which are transmitted by the method for public store-and-forward services.

4 Mathematical Steps:

4.1 Key Generator

In ElGamal, only the receiver needs to create a key in advance and publish it.As we disscussed above, we will now follow Bob through his procedure of key generation.Bob will take the following steps to generate his keypair:

1. Prime and Group Generation
   First,Bob needs to select a large prime p and the generator g of a multiplicative group $(Z_p)^*$ of the integers modulo p.

2. Private key selection
   Now Bob selects an integer b from the group Z by random and with the constraint $1 \leq b \leq p-2$

3. Public key Assembling
   Now,we can calculate the public key part $g^b \; mod \; p$.In ElGamal Cryptosystem, the public key of Bob is the triplet (p, g, g^b) and his private key is b.

4. Public key publishing
   Bob has to give the public key to Alice so he will publish using some dedicated keyserver or other means so that ALice is able to get hold of it.

4.2 Encryption

To encrypt a message M to Bob, Alice first needs to obtain his public key triplet (p, g, g^b) from a key server or by receiving it from him via unencrypted electronic mail. Thereis no security issue involved in this transmission, as the

only secret part, b, is sent in g^b. Since the core assumption of the ElGamal cryptosystem says that it is infeasible to compute the discrete logarithm, this is safe. For the encryption of the plaintext message M, Alice has to follow these steps:

1. Obtain the public key
   As described above, Alice has to acquire the public key part (p, g, g^b) of Bob from an official and trusted keyserver.

2. Prepare M for encoding
   Write M as set of integers $(m1, m2, \cdots)$ in the range of $1, \cdots, p-1$. These integers will be encoded one by one.

3. Select Random Exponent
   In this step, Alice will select a random exponent k that takes the place of the second party's private exponent in the Diffie-Hellman key exchange. The randomness here is a crucial factor as the possibility to guess the k gives a sensible amount of theinformation necessary to decrypt the message to the attacker.

4. Compute public key
   To transmit the random exponent k to Bob, Alice computes $g^k \, mod \, p$ and combines it with the ciphertext that shall be sent to Bob.

5. Encrypt the plaintext
   In this step, Alice encrypts the message M to the ciphertext C. For this, she iterates over the set created in step 2 and calculates for each of the m_i:
   $$c_i = m_1 \star (g^b)^k$$
   The ciphertext C is the set of all c_i with $0 < i \leq |M|$.

The resulting encrypted message C is sent to Bob together with the public key $g^k \, mod \, p$ derived from the random private exponent.

Even if an attacker would listen to this transmission, and in a second step would also acquire the public key part g^b of bob from a keyserver, he would still not be able to derive $g^{b \star k}$ as can be seen from the Discrete Logarithm problem.

Elgamal advises to use a new random k for each of the single message blocks m_i.This greatly improves security, as knowledge of one message block m_j does not lead the attacker to the knowledge of all other m_i. The reason for this ability is that if $c_1 = m_1 \star (g^b)^k \, mod \, p$ and $c_2 = m_2 \star (g^b)^k \, mod \, p$, from

knowing only m_1 the next part of the message m_2 can be calculated by the following formula:

$$\frac{m1}{m2} = \frac{c1}{c2}$$

4.3 Decryption

After receiving the encrypted message C and the randomized public key g^k, Bob has to use the encryption algorithm to be able to read the plaintext M. This algorithm can be divided in a few single steps:

1. Compute shared key
   The ElGamal cryptosystem helped Alice to define a shared secret key without Bob's interaction. This shared secret is the combination of Bobs private exponent b and the random exponent k chosen by Alice. The shared key is defined by the following equation:

   $$(g^k)^{p-1-b} = (g^k)^{-b} = b^{-bk}$$

2. Decryption
   For each of the ciphertext parts c_i Bob now computes the plaintext using

   $$m_i = (g^k)^{-b} \star c_i \, mod \, p$$

   After combining all of the m_i back to M he can read the message sent by Alice.

4.4 Examples

Alice chooses $p_A = 107$,$\alpha_A = 2$,$d_A = 67$, and she computes $\beta_A = 2^{67} \equiv 94(mod\,107)$. Her public key is $(p_A, \alpha_A, \beta_A) = (2, 67, 94)$, and her private key is $d_A = 67$.

Bob wants to send the message "B"(66 in ASCII) to Alice. He chooses a random integer k = 45 and encrypts $M = 66$ as $(r, t) = (\alpha_A^k, \beta_A^k M) \equiv (2^{45}, 94^{45}66) \equiv (28, 9)(mod\ 107)$.He sends the encrypted message (28,9)to Alice.

Alice receives the message $(r, t) = (28, 9)$, and using her private key $d_A = 67$ she decrypts to

$$tr^{-d_A} = 9.28^{-67} \equiv 9.28^{106-67} \equiv 9.43 \equiv 66(mod\,107).$$

5 Security

The security of the ElGamal scheme depends on the properties of the underlying group G as well as any padding scheme used on the messages.

- If the computational Diffie-Hellman assumption (CDH) holds in the underlying cyclic group G, then the encryption function is one-way.
- If the decisional Diffie-Hellman assumption (DDH) holds in G, then ElGamal achieves semantic security. Semantic security is not implied by the computational Diffie-Hellman assumption alone.

ElGamal encryption is unconditionally malleable, and therefore is not secure under chosen ciphertext attack. For example, given an encryption (c_1, c_2) of some (possibly unknown) message m, one can easily construct a valid encryption $(\mathrm{c}_1, 2c_2) of the message 2m$.

To achieve chosen-ciphertext security, the scheme must be further modified, or an appropriate padding scheme must be used. Depending on the modification, the DDH assumption may or may not be necessary.

Other schemes related to ElGamal which achieve security against chosen ciphertext attacks have also been proposed. The Cramer-Shoup cryptosystem is secure under chosen ciphertext attack assuming DDH holds for G. Its proof does not use the random oracle model. Another proposed scheme is DHAES, whose proof requires an assumption that is weaker than the DDH assumption.

6 Advantages and Disadvantages

Advantages:

1. One of the strength of ElGamal is its non-determinism-encrypting the same plaintext multiple times will result in different ciphertexts, since a random k is chosen each time.

2. El-Gamal encryption is used in the free GNU privacy Guard Software,recent versions ofPGP,and other cryptosystems.

Disadvantages:

1. Its need for randomness, and its slower speed(especially for signing).

2. The potential disadvantage of the ElGamal system is that message expansion by a factor of two takes place during encryption(means the ciphertext is twice as long as the plaintext.)

7 Applications

The ElGamal cryptosystem does not only support encryption and decryption, but also the electronically signing of messages M. A signature scheme has three main characteristics:

1. Creation
   Alice needs to be able to find the signature for M by using her private key a. She will then send the message together with the signature as the pair(M,S) to Bob.

2. Verification
   Bob has to be able to verify the signature by using the public key g^a.The verification of the signature assures Bob that Alice has signed the message as he received it. It does not deliver information about if Alice wrote the message herself or if she intended to send it at all. The second information Bob can draw from the verification is that the message has not been altered on the transmission path between him and Alice.

3. Forgery prevention
   It should not be possible for a malicious user to use the public key g^a of Alice to create a signature for an arbitrary message. A signature in the ElGamal cryptosystem is the pair (r, s) with $0 \leq r, s < p - 1$ defined by the equation
   $$g^M \equiv (g^a)^r (r^8) mod p$$
   The procedure of signing follows similar steps as the encryption procedure:

4. Choose random $k \in G$

5. Compute $r \equiv g^k mod p$

6. Fill the signature equation from above as $g^M \equiv g^{ar} gksmodp$ and solve it for s using $m \equiv ar + ksmod(p-1)$. This has a solution for s if k is

chosen such that $gcd(k, p-1) = 1$.

Bob received (M, r, s) and wants to verify the signature now. For this, he only needs to compute both sides of the equation 1 and check for equality.

8 Conclusion

In this paper, we have described the famous Elgamal Public-key Cryptosystem.The cryptosystem, which was developed in 1984 by Taher ElGamal. ElGamal encryption is used in the free GNU Privacy Guard software, recent versions of PGP, and other cryptosystems.We have looked at several kinds of attacks on the cryptosystem but its strength lies in the calculation of Discrete-Log problem. Here, The Algorithm is implemented using Java programming language.Topics Covered are:

- Description of cryptography and public-key cryptosystem.
- Analysis of ELGamal Public-key Cryptosystem.
- Implementation of ElGamal key Generaation, encryption and Decryption Algorithms.
- In the last, description about Signature Scheme of ElGamal Cryptosystem.

References

[1] Wikipedia
`https://en.wikipedia.org/wiki/ElGamal_encryption`

[2] Andreas V. Meier (2005 , June 08) The Elgamal Cryptosystem
`http://www14.in.tum.de/konferenzen/Jass05/courses/1/papers/meier_paper.pdf`

[3] `http://homepages.math.uic.edu/~leon/mcs425-s08/handouts/el-gamal.pdf`