USE OF GROUP THEORY IN CRYPTOGRAPHY

Priya Arora

Assistant Professor, Department of Mathematics

S.D. (P.G) College,Panipat

ABSTRACT

How group theory can be used in cryptography is described through this paper. The main purpose in cryptography is that the system developed for communication must be secure. The security of the system depends on the method on which the algorithm is based. In this paper we discuss the methods based on group theory.

Keywords: Cryptography, Algorithm, Theory

INTRODUCTION

This paper is based on cryptography. To make the system secure, It will provide some new concepts based on group theory. The literature on group-based cryptography; the paper will also provide a high level overview of the subject. Here it has been assumed that our reader already has a good knowledge of group theory, and a passing acquaintance with cryptography: the RSA and Diffie–Hellman schemes have been met before, and the difference between a public key and a symmetric key cipher is known. Although we will discuss the basics in this paper also.

The remainder of the paper is structured as follows. To start with the basics of cryptography, we introduce some of the most widely studied schemes in group-based cryptography and then we end up with conclusion.

BASICS OF CRYPTOGRAPHY

An original message is known as plain text, while the coded message is known is known as the cipher text. A process that is used for converting from plain text to cipher text is known as enciphering or encryption. Restoring the plain text from the cipher text is enciphering or decryption.

There are many schemes used for encryption constitute the area of study known as cryptography, such a scheme is known as cryptographic system or a cipher. Now the question arises how to decipher a message. The various techniques which are used for deciphering a message without any knowledge of the enciphering details fall into the area of crypt analysis. In the basic communication scenario, it is assumed that there are two parties, assume that the name of party is A and B for the sake of simplicity. Now A and B wants to communicate with each other. A third party C is a potential eavesdropper. When A wants to send a message, called the plain text, to B, she encrypts it using a method prearranged with B. Usually, the encryption method is assumed to be known to C; what keeps the message secret is a key. When B receives the encrypted message, called the cipher text , he changes it back to the plaintext using a decryption key. While doing all of these things it is assumed that C should have one of the following goals:

1. Message to be Read.
2. With the help of key read all messages encrypted with that key.
3. Corrupt A's message into another message in such a way that B will think A sent the altered message.
4. Masquerade as A, and thus communicate with B even though B believes he is communicating with A.

The main technique used in Cryptography is based on Encryption and decryption. The methods of Encryption/decryption fall into two categories: public key and symmetric key . In symmetric key algorithms, the encryption and decryption keys are known to both A and B. For example, the encryption key is shared and the decryption key is easily calculated from it. In many cases, the encryption key and the decryption key are the same. All of the classical cryptosystems are symmetric, as are the more recent Data Encryption standard (DES) and Advanced Encryption Standard (AES)

Now the question arises when the Public key algorithms was introduced. The public key algorithm was introduced in the 1970s.. Assume that A wants to communicate securely with B, but they are hundreds of kilometers apart and have not agreed on a key to use. It seems almost impossible for them to do this without first getting together to agree on a key, or using a trusted courier to carry the key from one to the other. Certainly A cannot send a message over open channels to tell B the key, and then send the cipher text encrypted with this key. Amazing fact is that this problem has a solution, called public key cryptography. The encryption key is made public, but it is computationally infeasible to find the decryption key without information known only to B. The most popular implementation is RSA.

KEY EXCHANGE BY DIFFIE-HELLMAN

In crytptography Key is required to be set up for use in cryptographic protocols such as DES or AES. Its major requirement arises when particularly two parties are widely separated. Now assume that A and B wants to create a shared key, then it can performed as follows:

1. On uniform basis A selects a random integer and proceed it to transfer to it to B.
2. Similar to that of A , B also selects a random integer and transfer it to A
3. On the basis of the points prescribed 1 and 2 above A computes ka = $(g^b)^a$ while B computes kb = $(g^a)^b$
4. The computed shared key will be $k = ka = kb \in G$.

SECURITY ISSUES DUE TO POSSIBLE ATTACKS

It is must to discuss the issue related to possible attacks during communication between A and B traced by C. The possibility of these type of attacks and arising the issue of security are as that it is possible that C has only a copy of the ciphertext. Another aspect is that C has a copy of ciphertext and the corresponding plaintext. Due to having very weak cryptosystems it may be possible to find the key easily. It is also possible to use the same pet words by A during communication and it is interpreted by the C and orignal messages is detected. Another possibility is to have the chosen plaintext i.e C gains temporary access to the encryption machine. He cannot open it to find the key, now the question arises however, he can encrypt large number of suitably chosen plaintext and try to use the resulting ciphertext to deduce the key. Last possibility regarding this issue may also be that C got the temporary access to the decryption machine and uses it to decrypt several strings and on the basis of it try to use the result to deduce the key. Another issue regarding the security issue is that how a plaintext attack can happen. It can be explained with a simple example related to army force. Suppose a airforce man wants to detect a particular airplane is friend or enemy. Now, how it can be identified. The airforce man will send a random message to the plane, which encrypt the message automatically and send it back. Only a friendly airplane is assumed to have the correct key. After it the message is to be compared with correctly encrypted message. If it is matches then the plane is friendly otherwise it is enemy. However, the other aspect of this issue is such that also the enemy sends a large number of chosen messages and look at the resulting ciphertext. If it is possible to detect the key, then the enemy can equip their planes so they can behave as friendly. Such example of plaintext attack was reported in World War-II. But modern days security is often much more demanding.

From the above facts presented , now we can easily say that modern cryptography is much broader than the traditional two party communication model we have discussed here: there is a thriving community developing the theory of multiparty communication, using such beautiful concepts as zero knowledge.

SYMETRIC AND PUBLIC KEY ALGORITHM

The methods related to Encryption and Decryption fall into two categories one is known as symmetric key and other is known as public key. Now what is the difference between these two methods. The basic difference is that in symmetric key algorithm encryption and decryption keys are known to both A and B. A particular instance of it is such that a public key is shared and decryption key is calculated from it. On the other hand public key algorithms were introduced in the 1970 and created a big boom for the cryptography. Again a particular instance of it is such that A want s to communicate with B securely but the distance between A and B is assume that is very high and there is no agreement regarding the use of a particular key. Under such circumstances, A cannot send the message over the open channel to tell B the Key and then send the ciphertext encrypted with this key. It appears under such circumstances that there is no solution of such problem. But this is not so, there is a solution, called public key cryptography. Under such cases the encryption key is made public, but on the basis of computation it is not possible to find the decryption key. It is possible with the help of RSA Algorithm.

USE OF HASH FUNCTIONS

There are various cryptography algorithms in which has function is known as a basic component. A cryptographic hash function h takes as input of any number of length and produce the output of fixed length. To do all these things certain property should be satisfied which are discussed in brief as follows:-

1. A message denoted with m is provided to a user then the message digest h(m) can be calculated very quickly.

2. Given a y, it is difficult to find an m.
3. It is difficult to find message m1 and m2 with h(m1)=h(m2).

One of the practical use of hash function is in digital signatures. All of this is due to the fact that because the length of a digital signature is often at least as long as the document being signed. Under such circumstances it is much more efficient to sign the hash of a document rather the full document. Data integrity can also be checked with the help of hash function. Now the question arises why the data integrity is required. The basic reason for this is that when the data are being transmitted to another person and the noisy communication channel introduces error to the data. The other thing when an observer rearranges the transmission in some manner before it gets to the receiver. In the both case data become corrupted. Now the question arises how to use a secure hash algorithm. What should be involved in making a real cryptographic hash function. Unlike block ciphers, where there are many block ciphers to choose from , there are just a small number of hash functions that are available. One of the popular secure hash function is known as Secure hash algorithm. The Secure hash algorithm was developed by National Security Agency(NSA) . The SHA-1 generate the output of 160 bits hash. These hash functions uses the iterative procedure. In this case the original message m is broken into a fixed size of blocks m=(m1,m2,m3…..mn). The message blocks are then processed with the help of sequence of rounds that use a compression function , which combines the current block and the result from the previous round.

RSA ALGORITHM

Before prescribing the exact steps of RSA algorithm it is must to explain the basic things related to RSA algorithm. Again assume that A wants to communicate with B. Under such cases how an RAS algorithm will work is prescribed in brief - First of all B chooses to distinct large primes number say x and y and multiplies them form a equation n=xy. At the same time an exponent used for encryption known as e is also selected such that gcd(e,(x-1)(y-1))=1. After computing these values a pair of (n,e) is sent to A. On the other hand assume that A keeps the values of x and y to be a secret one. Now A writes the message as a number to be known as m. Now if the message denoted by m written by A is larger than n then the message is cut off into blocks such that each of the cut off block is less than n. Now the situation of both are required to be prescribed. Assume that under particular circumstances the equation is true such that m<n. Then A computes c= m (mod n) and the computed value c is sent to B. Now B can compute (x-1)(y-1) and therefore can find the decryption exponent d. Now the brief summary of the algorithm can be written as follows:

1. B find out the secret primes and the value of n is computed by applying n=xy
2. B selects the exponent e with gcd(e,(x-1)(y-1))=1
3. B computes d with de=1 (mod(x-1)(y-1)).
4. B makes n and e public, and keeps x,y,d secret
5. Encryption by A for the message m as c=m (mod,n) and sends c to B.
6. B decrypts it by applying the equation m=c(mod n)

In the above procedure there are certain aspects which are must to consider. These issues are how does B chooses x and y values . These should be chosen random and independently to each other. What should be the size of it that always depends on the type of security needed. Another aspects that is must to consider is that why does B reaquires gcd(e.(x-1)(y-1)=1)). All these issues are required to be resolved first for the success of encryption and decryption technique to be followed for a message to obtain DES key.

SECURITY FEAR OF RSA

If RSA algorithm is implemented correctly then the algorithm is effective and useful in cryptography. But there are some other issues also related to fear of security of RSA algorithm used in Cryptography. A well known attack on RSA is discovered by Paul Kocher. He demonstrate that it is possible to discover the decryption exponent by carefully timing the computation times for a series of decryptions. Kocher attacks clearly shows that a system could still have unexpected weakness. The main point related to timing attack are as that assume C is competent to see B decrypt several ciphertexts.

GROUPS BASSED CRYPTOGRAPHY

Now a days there are various cryptographic protocols based on groups. The first proposal to use nonabelian groups in public key cryptograph. By using Group theory we can construct variants of the Diffie–Hellman key agreement protocol. Since the protocol uses a cyclic subgroup of a finite group G, one approach is to search for examples of groups that can be efficiently represented and manipulated. All the proposals discussed above use representations of abelian (indeed,cyclic) groups. What about non-abelian groups? The first proposal to use non abelian groups that we are aware of is due to Wagner and Magyarik in 1985.

CONJUGACY SEARCH PROBLEM

Assume that G be a non-abelian group to discuss the problem related to Conjugary search problem. Assuming that we can find a group where the conjugacy search problem is hard (and assuming the elements of this group are easy to store and manipulate), one can define cryptosystems that are similar to cryptosystems based on the discrete logarithm problem. Ko et al. proposed the following analogue of the Diffie–Hellman key agreement protocol.

DIFFIE-HELLMAN KEY EXCHANGE

How to find out the keys for use in cryptography protocols such as DES is the main problem when the two parties say A and B are widely separated. Public Key methods as used in the RSA Algorithm provides one solution. On the other hand there are several technical implementation issues related to any key distribution scheme. Now the question arises how A and B set up the private key K to be used to obtain the original message. The brief algorithm steps in the Diffie-Hellman algorithm are as follows:-

1. A or B any one can selects a large, secure prime number p and a primitive root say t and t1. Both of these t and t1 are also made public
2. Now A selects a secret random x with $1<x<t1-2$ and B selects a secret random y with $1<y<p-2$
3. A sends t (mod t1) to B and B sends to A
4. After receiving the message by A and B they can calculate the session key K.

Now both of them i.e A and B have the same number K , they can use some prearranged procedure to produce the key. For example they can decide to select middle bits of K

CONCLUSION

The various algorithms were used and indicated with example to show the secure Cryptography and it clearly shows that the protocols of Ko et al is a good idea to implement the groups successfully in the process of cryptography. Now the question here arises is that can such a platform group be found? Here, we need a candidate group whose elements can be manipulated and stored efficiently. Although major emphasis is paid by the various algorithms on the infinite group based cryptography. Efforts can be made to use finite group based cryptography. Although finite group based cryptography have many difficulties during implementation but it has the more advantageous then the infinite group cryptography.

REFERENCES

1. Iris Anshel, Michael Anshel and Dorian Goldfeld, An algebraic method for public-key cryptography, Math. Res. Lett. 6 (1999), 287-291.
2. Iris Anshel, Michael Anshel, Dorian Goldfeld and Stephane Lemieux, Key agreement, the Algebraic Eraser^{TM}, and lightweight cryptography, Contemp. Math. 418 (2006) 1–34.
3. Norman Biggs, The critical group from a cryptographic perspective, Bull. London Math. Soc. 39 (2007) 829–836.
4. Simon R. Blackburn, Cryptanalysing the critical group Cryptology 8 (1995), 157–166.
5. Mar´ıa Isabel Gonz´alez Vasco, Spyros Magliveras and Rainer Stein-wandt, Group-theoretic cryptography, Chapman & Hall / CRC Press, to appear.
6. Mar´ıa Isabel Gonzalez Vasco, Martin R¨otteler and Rainer Steinwandt, On minimal length factorizations of finite groups, J. Exp. Math. 12 (2003), 1–12.
7. Spyros S. Magliveras and Nasir D. Memon, The algebraic properties of cryptosystem PGM, J. Cryptology 5 (1992), 167–183.
8. S. S. Magliveras, D. R. Stinson and Tran van Trung, New approaches to designing public key cryptosystems using one-way functions and trap-doors in finite groups, J. Cryptology 15 (2002) 167–183.
9. Sean Murphy, Kenneth Paterson, and Peter Wild, A weak cipher that generates the symmetric group, J. Cryptology 7 (1994) 61–65.
10. National Institute of Standards and Technology, Secure Hash Standard, Federal Information Processing Standards Publication (FIPS) 180-1, 1995.