Volume 4, Issue 6, June 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Enhanced Diffie Hellman Algorithm

Ekta Lamba
Research Scholar M.Tech (CSE)
YIET, Gadhola, YamunaNagar (Kurukshetra University)
Haryana, India

Lalit Garg
Assistant Professor (CSE)
YIET, Gadhola, YamunaNagar
Haryana, India

Abstract— Cryptography is the science of using mathematics to encrypt and decrypt data. Diffie Hellman algorithm is an asymmetric cryptography scheme for the encryption and decryption of data over computer network. This algorithm is meant for the generation of secret key. One major problem with Diffie Hellman algorithm is the man-in-the middle attack which can be overcome with digital signatures and public key certificates. Another problem is the brute force attack .This attack can affect only if the prime number used is small as the attack will work by finding the discrete logarithm. In this paper, we have tried to give focus on the hardness of key by using safe primes that makes it almost infeasible to calculate discrete logarithms & thus using that key for encryption and decryption of data so that we get better security. The Diffie-Hellman encryption algorithm is enhanced by adding some more security codes or changes in the current algorithm.

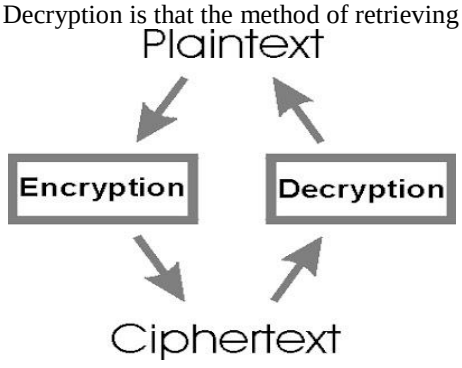
Keywords—DiffieHellmanAlgorithm,asymmetric,cryptography,encryption,decryption.

I. INTRODUCTION

Most initial computer applications had no or very little security. This continued for a number of years until the importance of data was truly realized. When computer applications were developed to handle financial and personal data, the actual need for security in computer networks was felt like never before. Therefore, various areas in security began to gain prominence. Furthermore, the internet took the world by storm, and there were many examples of what could happen if there was insufficient security built in the applications developed for the internet. For example use of credit card for making purchases over the internet. For this security of information, numerous ways have been discovered. Some admired ways are cryptography and steganography

1.1 CRYPTOGRAPHY

Cryptography is a Greek word that virtually means that the art of writing secrets. In practice, cryptography is that the task of transforming data into a type that's unintelligible, but at the same time allows the intended recipient to retrieve the original data using a secret key. Cryptographic algorithms (or ciphers, as they are often called) are special programs designed to protect sensitive data on open communication links. During encryption, ciphers transform the initial plaintext message into unintelligible cipher text. Decryption is that the method of retrieving plaintext from cipher text.



1.2 Diffie-Hellman Algorithm

The purpose of the algorithm is to enable two users to securely exchange a key that can then be used for
Subsequent encryption of messages. The algorithm itself is limited to the exchange of secret values. The Diffie-Hellman algorithm depends for its effectiveness on the difficulty of computing discrete logarithms.

There are two publicly known numbers; a prime number q and an integer a that is a primitive root of q. Suppose that the user A & B wish to exchange a key. User A selects a private random integer $XA < q$ & Calculate public $Y_A = \alpha^{XA} \bmod q$. Similarly, User B selects a private random integer $XB < q$ & calculate public $Y_B = \alpha^{XB} \bmod q$.
Each side keep X value private & makes the Y value available publicly to the other side. User A computes the key as $K = (Y_B)^{XA} \bmod q$. User B computes the key as $K= (Y_A)^{XB} \bmod q$. This K is same
$K = (Y_B)^{XA} \bmod q$

$= (\alpha^{XB} \bmod q)^{XA} \bmod q$
$= (\alpha^{XB})^{XA} \bmod q$ (by rules of modular arithmetic)
$= (\alpha^{XA})^{XB} \bmod q$
$= (\alpha^{XA} \bmod q)^{XB} \bmod q$
$= (Y_A)^{XB} \bmod q$

The attacker has only the following ingredients to work with
q, a, Y_A, Y_B.
The attacker is forced to take a discrete logarithm to determine the key.
e.g.: To determine the private key of user B,

$$X_B = \text{dlog}_{a,q}(Y_B)$$

The attacker can now computes key K in the same manner as B calculates it

$$K = (Y_A)^{XB} \bmod q$$

The security of this algorithm lies in the fact that, while it is relatively easy to calculate exponential modulo of a prime, it is very difficult to calculate discrete logarithms. For large primes, it is considered infeasible.

Figure 1 shows a simple protocol that makes use of the Diffie-Hellman calculation and exchange. Suppose that user A wishes to set up a connection with user B and use a secret key to encrypt messages on that connection. User A can generate a one-time private key *XA*, calculate *YA*, and send that to user B. User B responds by generating a private value *XB* calculating *YB*, and sending *YB* to user A. Both users can now calculate the key. The necessary public values *q* and *α* would need to be known ahead of time. Alternatively, user A could pick values for *q* and *α* and include those in the first message [8]

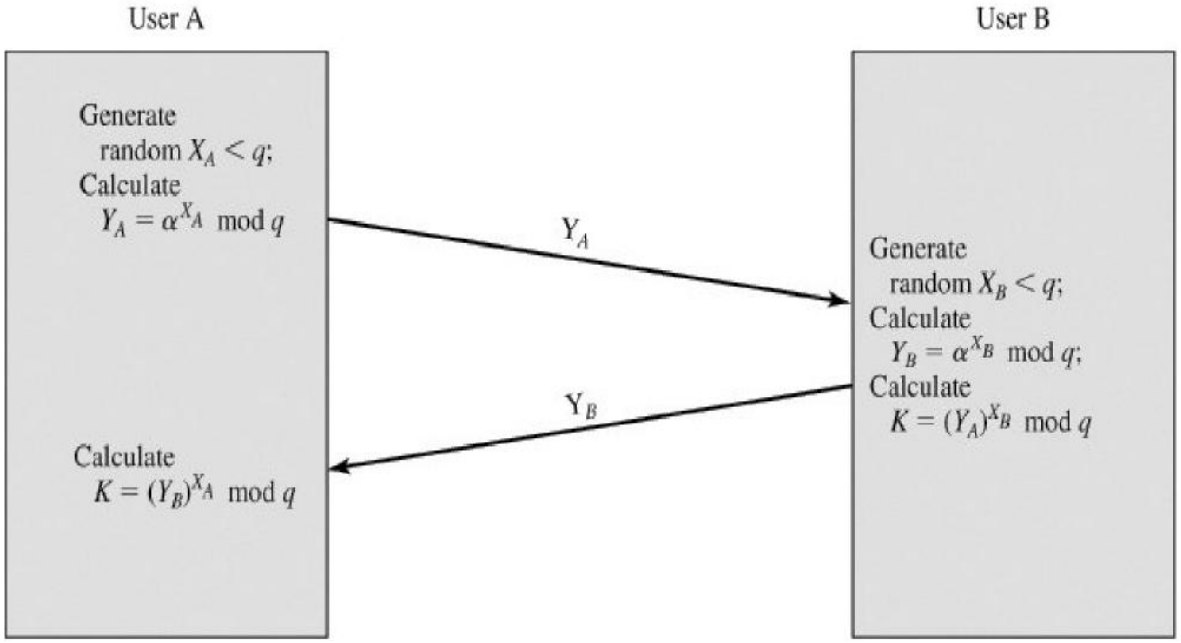


Fig 1: Diffie Hellman Algorithm

The protocol depicted in Figure 1 is insecure against a man-in-the-middle attack. The key exchange protocol is vulnerable to such an attack because it does not authenticate the participants. This vulnerability can be overcome with the use of digital signatures and public-key certificates.[8]

1.3 Entropy

The entropy of a document is an index of its information content. Entropy is an expression of insecurity as the number of Yes/No questions which have to be answered in order to clarify a message or a character. If a character has a very high probability of occurrence, then its information content is low. This would be the case, for example, with a business partner who regularly replies "Yes". This reply also does not permit any conclusions to be drawn as to understanding or attention. Replies which occur very seldom have high information content.

Extreme values of entropy

For documents which contain only upper case letters, the entropy lies between 0 bit/char (in a document which consists of only one character) and log(26) bit/char = 4.700440 bit/char (in a document in which all 26 characters occur equally often).

For documents which can contain every character of the character set (0 to 255) the entropy lies between 0 bit/char (in a document which consists of only one character) and log(256) bit/char = 8 bit/char (in a document in which all 256 characters occur equally often).

II. RELATED STUDY

In [1], P. Bhattacharya, in 2005, the author proposed two modifications of DH. The first modification is to change the domain to integer with n=2pt where Zn is still cyclic and the second modification is to change the domain to Gaussian arithmetic Z [i].

In [2], Yun Chen, in2006, the author proposes a novel key generation algorithm for DH agreement that derives computational efficiency from constructing a parallel architecture. Compared to the serial structure for traditional Binary Representation (BR) method, their algorithm is significantly more efficiency on key generation and suitable for hardware implementation in an ephemeral-static mode for DH agreement which is thought to be more secure.

In [3], Ik Rae Jeong, in May 2007, provided an overview that to provide authentication to the Diffie-Hellman key exchange, a few integrated key exchange schemes which provide authentication using the DSA signature have been proposed. It pointed out that all of the previous Diffie-Hellman-DSA schemes do not provide security against session state reveal attacks. It also suggest a strong Diffie-Hellman-DSA scheme providing security against session state reveal attacks as well as forward secrecy and key independence.

In [3], Zhen Cheng, in 2008, the author proposes the algorithm for elliptic curve Diffie-Hellman key exchange based on DNA tile self-assembly. First they give the DNA tile self-assembly model to compute the scalar multiplication, and then they can successfully implement the Diffie-Hellman key exchange over elliptic curve by extracting the result strand of the scalar multiplication. DNA tile self-assembly is looked forward to many applications in fields. They mainly propose the algorithm of elliptic curve Diffie-Hellman key exchange based on DNA tile self-assembly

In [4], Salvatore Cavalieri in 2009 deals with the problem of making secure data transmission inside Home and Building Automation environment; here, data exchanged may regard commands to actuators and/or private and secret information. The paper deals with this problem taking into account the KNX communication system, which at this moment, doesn't foresee any encryption and authentication mechanisms. A solution for data encryption and authentication will be presented and assessed, comparing it with the current state of the art.

In [5], Eun-Jun Yoon, in 2009, proposed an efficient Diffie-Hellman-MAC key exchange scheme providing security against session state reveal attacks as well as forward secrecy and key independence.

In [6], Vishnu Kumar, in 2010, the author has presented a secure energy efficient dynamic routing scheme (SEEDR) for wireless sensor networks. SEEDR uses a symmetric cryptography algorithm to support security. The dynamic key exchange protocol based on DH (Diffie-Hellman) algorithm is proposed, with non blocking OVSF codes. Their analytical model and the security analysis prove that SEEDR increase data communication security and consume less communication energy.

In [7], S. Anahita Mortazavi, in 2011, in this author proposed an efficient many-to-many group key management protocol in distributed group communication. In this protocol, group members are managed in the hierarchical manner logically. Two kinds of keys are used, asymmetric and symmetric keys. The leaf nodes in the key tree are the asymmetric keys of the corresponding group members and all the intermediate node keys are symmetric keys assigned to each intermediate node. For asymmetric key, Diffie-Hellman key agreement is introduced. To calculate intermediate node keys, members use codes assigned to each intermediate node key tree. Group members calculate intermediate node keys rather than distributed by a sponsor member. The features of this approach are that, no keys are exchanged between existing members at join, and only one key, the group key, is delivered to remaining members at leave.

In [8], Vishal Garg, in 2012, provided harder encryption with enhanced public key encryption protocol for security and proposed work can be implemented into any network to provide better security. It enhanced the hardness in security by improving the Diffie-Hellman encryption algorithm by adding some more security codes in current algorithm.

III. PROPOSED WORK

The main problem with the Diffie hellman algorithm is the man-in the middle attack that can work only when the prime number q is small because finding discrete logrithms is infeasible for large primes. So, our basic focus is on the prime numbers. In our algorithm there is no barrier on the limit of prime number q. Also here , no prior aggrement of both sender and receiver is required on the value of q and a. There is no limit for the integers X_A and X_B.. Various checks for the generation of safe primes have also been applied here. We have enhanced the diffie hellman algorithm by generating a harder encryption and decryption key so that it becomes very difficult to find key.

IV. RESULTS

4.1 ALGORITHM

At Sender's Site

1. Input a prime number.
2. Check whether it is prime number or not with a user-defined function.

3. After that again it is checked whether it is a safe prime number or not by an in-built function.
4. Two numbers p & g are generated.
5. P & g are transferred to the receiver.
6. Key generation bits are specified as 256 bits.
7. Key is generated.

At Receiver's site

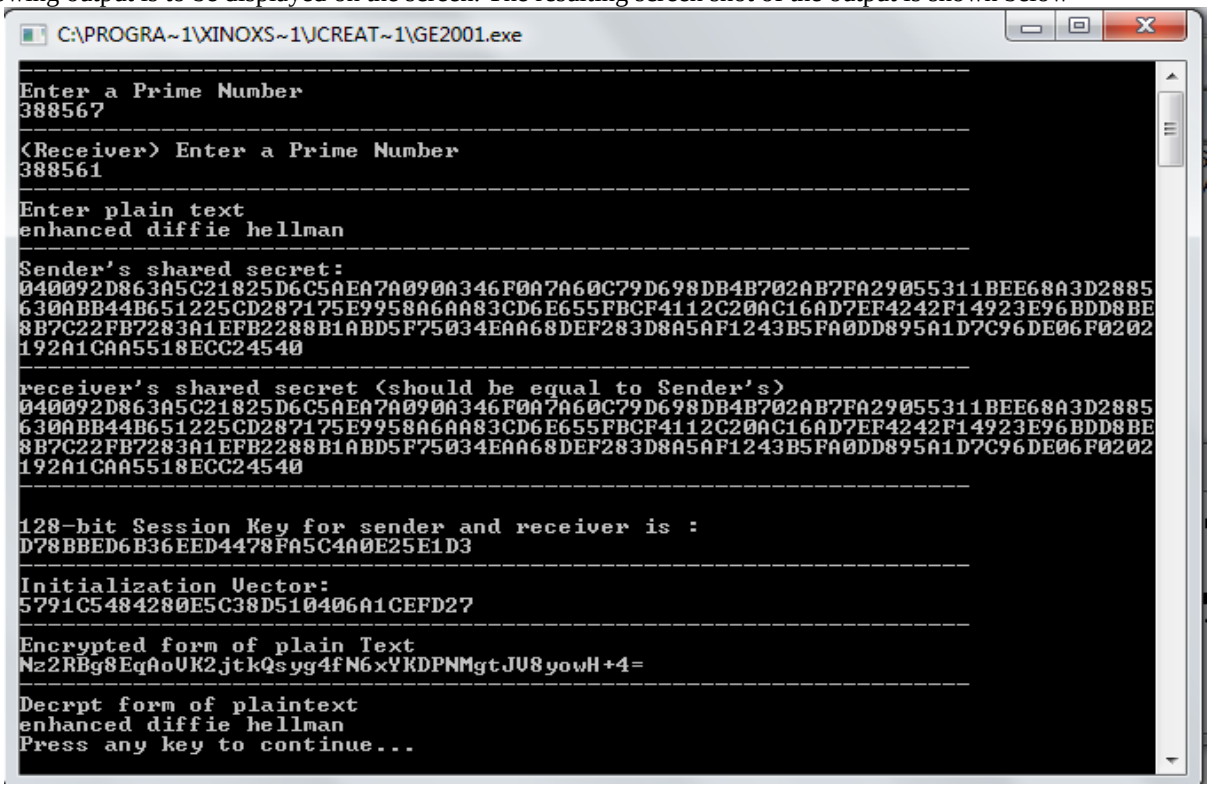
1. Input a prime number.
2. Check whether it is prime number or not with a user-defined function.
3. P & g are received from the sender.
4. Now receiver will check whether p & g are save primes or not.
5. Key generation bits are specified as 256 bits.
6. Key is generated which is same as that of sender's site.

4.2 IMPLEMENTATION

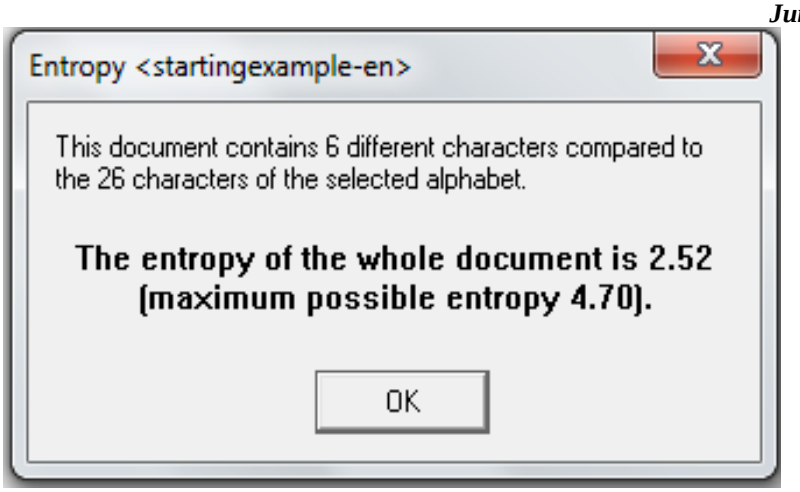
The below fig represent the implementation of proposed work in java.

```
class deffihellman {

  static {
    try {
        System.loadLibrary("chilkat");
    } catch (UnsatisfiedLinkError e) {
      System.err.println("Native code library failed to load.\n" + e);
      System.exit(1);
    }
  }


    public static byte UseKnownPrime(byte p, byte g) {
        byte aa = p, bb = g, r = 0, t;
        while (aa != 0) {
            if ((aa & 1) != 0)
                r = (byte) (r ^ bb);
            t = (byte) (bb & 0x80);
            bb = (byte) (bb << 1);
            if (t != 0)
                bb = (byte) (bb ^ 0x1b);
            aa = (byte) ((aa & 0xff) >> 1);
        }
        return r;
    }
```

The program is compiled using the default setting in jdk1.6 development kit for java. After executing the program, following output is to be displayed on the screen. The resulting screen shot of the output is shown below



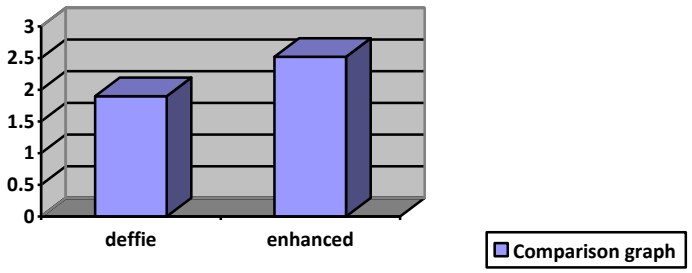
The result of the enhanced Diffie Hellman algorithm has been analysed using cryptool 1.4.30. The entropy of the enhanced diffie Hellman is 2.56.



Now we have compared the entropy of both diffie Hellman Algorithm and enhanced diffie- Hellman algorithm. From the comparison table, we see that entropy of enhanced diffie Hellman is higher than diffie Hellman .

	ENTROPY
Diffie- Hellman	**1.90**
Enhanced Diffie-Hellman	**2.52**

The bar graph between diffie-hellman algorithm and enhanced diffie-hellman algorithm has been shown as under



V. CONCLUSIONS

In this paper, a new design for enhancing the security of Diffie Hellman algorithm is proposed. This approach design will not contradict the security of the original Diffie-Hellman algorithm by keeping all the mathematical criteria of Diffie-Hellman algorithm remain unchanged. We try to improve the security of Diffie-Hellman Algorithm by making the key harder by the use of safe primes. We have also shown cryptanalysis of the results by using cryptool.

REFERENCES

[1] P. Bhattacharya, M. Debbabi and H. Otrok, "Improving the Diffie-Hellman Secure Key Exchange", International Conference on Wireless Networks, Communications and Mobile Computing in 2005.

[2] Ik Rae Jeong, Jeong Ok Kwon, and Dong Hoon Lee," Strong Diffie-Hellman-DSA Key Exchange", IEEE COMMUNICATIONS LETTERS, VOL. 11, NO. 5, MAY 2007.

[3] Zhen Cheng, Yufang Huang, Jin Xu,"Algorithm for Elliptic Curve Diffie-Hellman Key Exchange Based on DNA Tile Self-assembly", in 2008.

[4] Salvatore Cavalieri and Giovanni Cutuli," Implementing Encryption and Authentication in KNX using Diffie-Hellman and AES Algorithms", in 2009.

[5] Eun-Jun Yoon, Kee-Young Yoo," An Efficient Diffie-Hellman-MAC Key Exchange Scheme,"Fourth International Conference on Innovative Computing, Information and control, in 2009.

[6] Dongfang Zhang," A New Authentication and Key Agreement Protocol of 3G based on Diffie-Hellman Algorithm, "in 2010.

[7] S. Anahita Mortazavi, Alireza Nemaney Pour," An Efficient Distributed Group Key Management using Hierarchical Approach with Diffie-Hellman and Symmetric Algorithm: DHSA", International symposium on computer networks and distributed systems, February 23-24, 2011.

[8] Vishal Garg, Rishu,"Improved Diffie-Hellman Algorithm for Network Security Enhancement", Int.J.Computer Technology & Applications, Vol 3 (4), 1327-1331 IJCTA | July-August 2012 Available online@www.ijcta.com 1327

[9] Emmanuel Bresson, Olivier Chevassut, David Pointcheva and Jean-Jacques Quisquater, "Provably Authenticated Group Diffie-Hellman Key Exchange", in *Proc. Of ACM CCS '01,* ACM Press 2001.

[10] Michel Abdalla, Mihir Bellare, and Phillip Rogaway, "DHIES: An encryption scheme based on the Diffie-Hellman Problem", In Proc.of ACM CCS '01, ACM Press September18, 2001.

[11] Jonathan C.Herzog, "The Diffie-Hellman Key-Agreement Scheme in the Strand-Space Model", 16th IEEE Computer Security Foundations Workshop (CSFW'03), 1063-6900/03, 2003.

[12] Lein Harn, Manish Mehta and Wen-Jung Hsin, "Integrating Diffie–Hellman Key Exchange into the Digital Signature Algorithm (DSA)", IEEE COMMUNICATIONS LETTERS, VOL. 8, NO. 3, MARCH 2004

[13] Mario Cagaljm, Srdjan Capkun and Jean-Pierre Hubaux, "Key agreement in peer-to-peer wireless networks", Laboratory for Computer Communications and Applications (LCA) Ecole Polytechnique F´ed´erale de Lausanne (EPFL), CH-1015 Lausanne Networked & Embedded Systems Laboratory (NESL), University of California, Los Angeles (UCLA), November 2004.

[14] Raphael C.-W. Phan, "Fixing the Integrated Diffie-Hellman-DSA Key Exchange Protocol", IEEE COMMUNICATIONS LETTERS, VOL. 9, NO. 6, JUNE 2005.

[15] L. Harn, W.-J. Hsin and M. Mehta, "Authenticated Diffie–Hellman key agreement protocol using a single cryptographic assumption", IEEE Proc.-Commun., Vol. 152, No. 4, August 2005.

