International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 4, Issue 1, January-February 2015 **ISSN 2278-6856**

Analysis of Diffie Hellman Key Exchange Algorithm with Proposed Key Exchange Algorithm

Mr. Randhir Kumar[1], Dr. Ravindranath C. C[2]

[1]Jagran Lakecity University, School of Engineering & Technology,
Mughaliyachhap, Bhopal MP, India

[2]Trinity Collage, Department of Electrical Engineering,
Raisen Road, Bhopal MP, India

Abstract

Asymmetric cryptography or public-key cryptography is cryptography in which a pair of keys is used to encrypt and decrypt a message so that it arrives securely. Initially, a network user receives a public and private key pair from a certificate authority. Any other user who wants to send an encrypted message can get the intended recipient's public key from a public directory. They use this key to encrypt the message, and they send it to the recipient. When the recipient gets the message, they decrypt it with their private key, which no one else should have access to. This paper shows the comparative analysis of Diffie Hellman algorithm and generation of public key over the insecure transmission in network. In proposed algorithm the time complexity for generating the public key, collision attack has been measured and compared with Diffie Hellman algorithm.

Keywords:- Diffie-Hellman ,collision,security, attacks, complexity ,Encryption, Decryption.

1. INTRODUCTION

Diffie Hellman is a specific method of exchanging keys. It is one of the earliest practical examples of Key exchange implemented within the field of cryptography. This key can then be used to encrypt subsequent communications using a symmetric key chipper. The Diffie Hellman Algorithm is used to generate the public key. Public key algorithm for key exchange allows two users to exchange a secret key over an insecure medium without any prior secrets [2]. The Diffie Hellman Functionality is limited to key exchange only. This algorithm cannot be used for Encryption/Decryption and does not provide authentication to the communication parties. The algorithm is Vulnerable to man-in the middle attack [3]. In proposed algorithm time complexity and analysis will be measured as well as Diffie Hellman key will be used as encryption and decryption using RSA. Both RSA and Diffie-Hellman are public key encryption algorithms strong enough for commercial purposes [13]. The minimum recommended key length for encryption systems is 128 bits, and both exceed that with their 1,024-bit keys. Both were invented in the late 1970s and have yet to be cracked. The nature of the Diffie- Hellman key exchange, however, makes it susceptible to man-in-the-middle (MITM) attacks, since it doesn't authenticate either party involved in the exchange [16]. The MITM maneuver can also create a key pair and spoof messages between the two parties, who think they're both communicating with each other. Mutually authenticating both parties can defeat attempts at MITM attacks.

2. DIFFIE HELLMAN

Asymmetric Encryption of data requires transfer of cryptographic private key. The most challenging part in this type of encryption is the transfer of the encryption key from sender to receiver without anyone intercepting this key in between [1]. This transfer or rather generation on same cryptographic keys at both sides secretively was made possible by the Diffie-Hellman algorithm. The Diffie-Hellman algorithm was developed by Whitfield Diffie and Martin Hellman in 1976. This algorithm was devices not to encrypt the data but to generate same private cryptographic key at both ends so that there is no need to transfer this key from one communication end to another [7]. Though this algorithm is a bit slow but it is the sheer power of this algorithm that makes it so popular in encryption key generation. Diffie-Hellman was the first published public key algorithm that is used for secure key exchange mechanism. The purpose of algorithm is used to enable users to security exchange a key that can be used for subsequent encryption. This cryptographic problem ensure A (resp. B) that no other participants aside from B (resp. A) can learn any information about the agreed value and often also ensure A and B that their respective partner has actually computed this value. But this algorithm is no longer strong, since the key can be easily identified by discrete logarithmic approach. Hence, in order to strengthen this algorithm, we are going to reduplicate the Diffie Hellman algorithm with different methodology. In this paper we will make comparative study over Diffie Hellman and our proposed algorithm approach.

3.COLLISION ATTACK ON DIFFIE HELLMAN

The algorithm is insecure against man in the middle attack as follows, suppose there are a user C who is going intercept the secret key shared between user A and user B. C generates two random private keys xd1and xd2 and then computes corresponding public keys y d1 and y d2. User A transmits public key ya to user B. in the meanwhile user C intercepts y a and transmits his public key y d1 to user B. user C also calculate K2= y d2 xa mod n. B receives y d1 and calculate secret key K1=y *d*1 xb mod n. Now, user B transmits his private key xa to A. Now, C intercepts and transmits his own public key y d2 to A. Now, A receives yd2 and calculates corresponding K2. Since the algorithm is easily cracked by discrete logarithm approach as above, we have to strength the algorithm to avail better security key transmission.

4.ALGORITHM OF DIFFIE HELLMAN

The General Algorithm for Diffie Hellman is given below:

Step I: Select two prime value p and g for both parties during key generation.

Step II: Generating two random numbers

a = rand () % 50;

b = rand () % 50;

Step III: Generation of two random key with mod operation

r1 = mod (g, a, p); // g^a mod p

r2 = mod (g, b, p); // g^b mod p

Step IV: Exchange of key to both the party

k1 = mod (r2, a, p); // r2^a mod p

k2 = mod (r1, b, p); // r1^b mod p

5 PROPOSED ALGORITHM OF DIFFIE HELLMAN

The proposed algorithm for key generation has been shown in two methods.

Method-I:

Step-I: Select two prime value p and g for both parties during key generation.

Step-II: Generating two random numbers

a = rand () % 100;

b = rand () % 100;

Step-III: Generation of two random key with mod operation

r1 = (g << a) * p

r2 = (g <<b) * p

Step-IV: Exchange of key to both the party

k1 = (r2 <<a) *p

k2 = (r1 << b) * p

Method –II:

Step-I: Select two prime value p and g for both parties during key generation.

Step-II: Generating two random numbers

a = rand () % 100;

b = rand () % 100;

Step-III: Generation of two random key with mod operation

r1 = (g * a) * p

r2 = (g * b) * p

Step-IV: Exchange of key to both the party

k1 = (r2 * a) * p

k2 = (r1 * b) * p

6 TIME COMPLEXITY AND ANALYSIS FOR DIFFIE HELLMAN AND PROPOSED ALGORITHM

The time complexity of the general algorithm is shown in table-I.

Table-I:

Xa	Xb	G	N	Time in (ms)
3	7	5	7	2621961879710
3	5	23	53	2621961594354
7	11	11	13	2621961965370
7	11	17	23	2621962111512
41	11	31	41	2621962202524

Xa and Xb is two prime number namely p and g, G and N is the two random key generation, based on this random key generation public key is use to be created in Diffie Hellman algorithm. The time complexity of the proposed algorithm for Diffie Hellman in shown in Table-II

Xa	Xb	G	N	Time in (ms)
3	7	21	37	1375168097953
3	5	34	86	1375168126859
7	11	74	39	1375168162687
7	11	22	97	1375168204328
41	11	92	13	1375168232906

Xa and Xb is two prime number namely p and g, G and N is the two random key generation, based on this random key generation public key is use to be created in Diffie Hellman algorithm. in above time complexity is measured in terms of execution with Diffie Hellman and proposed Algorithm. The time complexity of the proposed algorithm for Diffie Hellman in shown in Table-III

Xa	Xb	G	N	Time in (ms)
3	7	9	66	1375168270640
3	5	43	3	1375168296765
7	11	11	50	1375168331062
7	11	51	86	1375168353859
41	11	37	32	1375168384734

Xa and Xb is two prime number namely p and g, G and N is the two random key generation, based on this random key generation public key is use to be created in Diffie Hellman algorithm. in above table time complexity is measured in terms of execution with Diffie Hellman and proposed Algorithm. The proposed algorithm is use to generate the key which is too long and it is very hard to attack the key using discrete approach algorithm.

7 FREQUENCY MEASUREMENT AFTER ENCRYPTION

Diffie Hellman with Encryption

A	E	I	O	U
212	192	146	228	272

Proposed Algorithm with Encryption

A	E	I	O	U
195	74	129	152	153

RESULT

Figure-I (Proposed Key Exchange Algorithm with Diffie Hellman)

In above Result there is two prime values has been selected 13 and 7, based on that public key of Diffie Hellman is calculated that is 12 for both the party to transmit the secure information. In the case of Diffie Hellman Algorithm key size is always generated with fixed length.

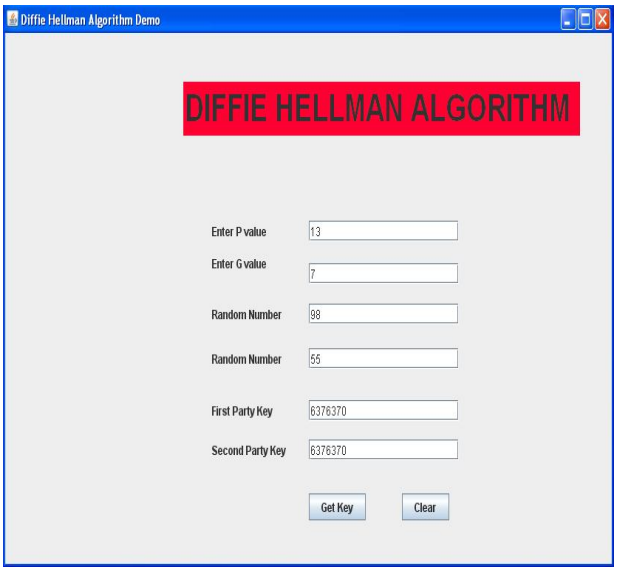

Figure II (Proposed Key Exchange Algorithm with Diffie Hellman)

In above Result there is two prime values has been selected 13 and 7, based on that public key of Diffie Hellman is calculated that is 6376370 for both the party to transmit the secure information. In the case of Diffie Hellman Proposed Algorithm key size can vary so it is hard to have middle attack.

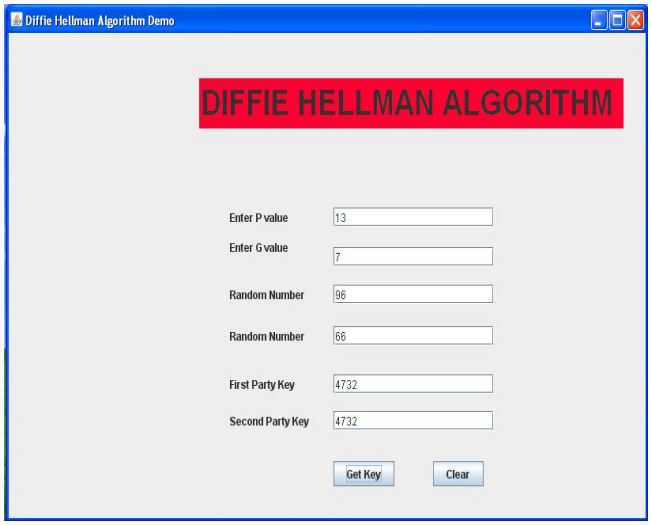

Figure III (Proposed Key Exchange Algorithm with Diffie Hellman)

In above Result there is two prime values has been selected 13 and 7, based on that public key of Diffie Hellman is calculated that is 4732 for both the party to transmit the secure information. In the case of Diffie Hellman Proposed Algorithm key size can vary so it is hard to have middle attack.

8.CONCLUSION AND FUTURE ENHANCEMENT

Diffie Hellman Key can be used with RSA, DES Public key and AES public key to encrypt and decrypt the message. The Diffie Hellman Key and Encryption Techniques can be used with all types of media. Diffie Hellman Key can be used with Elliptical Curve Cryptography using Encryption and Decryption.

ACKNOWLEDGEMENT

The authors wish to thank the management of Jagran Lakecity University Bhopal and Dr. Ravindranath C. C, Principal, Trinity Bhopal for their constant encouragement for completion of this work.

References

[1]. Y. Amir, Y.Kim, C. Nita-Rotaru, " Secure communication using contributory key agreement", IEEE Transactions on Parallel and Distributed systems, pp. 468-480,2009.

[2]. Ram Ratan Ahirwal, Manoj Ahke, Elliptic Curve Diffie-Hellman Key Exchange Algorithm for Securing Hypertext Information on Wide Area Network, IJCSIT

[3]. Yufang Huang, "Algorithm for elliptic curve Diffie-Hellman key exchange based on DNA title self

assembly In Proceedings of 46th IEEE Theories and Applications, pp.31-36, 2008.

[4]. A. M. Fiskiran and R. B. Lee. "Workload characterization of elliptic curve cryptography and other network security algorithms for constrained environments". IEEE International Workshop on WWC-5, 2009

[5]. B. Kaliski, M. Liskov and Y. L. Yin. "Efficient finite field basis conversion techniques". Proposal for Inclusion in IEEE P1363, 2009

[6]. V.S. Miller, "uses of elliptic curves in cryptography," in Advances in Cryptology, CRYPTO'85, ser. Lecture Notes in Computer Science, vol. 218, Springer, 1986. pp. 417-428.

[7]. N. Koblitz, "Elliptic curve cryptosystems," Mathematics of Computation, vol. 48, no.177, pp.203-209, Jan 1987.

[8]. D. Hakerson, A. Menezes, and S. Vanston, "Guide to Elliptic Curve Cryptography," Springer-Verlag, NY (2004).

[9]. H. Cohen, A Miyaji and T. Ono, "Efficient elliptic curve exponentiation using mixed coordinates," Lectures Notes in Computer Science, 1514, 51-65 (1998).

[10]. Dimitrov V., L. Imbert, and P. K. Mishra, "Efficient and secure elliptic curve point multiplication using double-base chains,"Lectures Notes in Computer Science, 3788, 59-78 (2005).

[11]. M. Ciet, M. Joye, K. Lauter and P.L. Montgomery,"Trading inversions for multiplications in elliptic curve cryptography," Designs, Codes, and Cryptography, 39, 189-206 (2006).

[12]. Y. Chen, J.S. Chou, C.H. Huang , "Comments on five smart card based password authentication protocols," International Journal of Computer Science and Information Security 8 (2) (2010)129–132.

[13]. Ch. Suneetha, D. Sravana Kumar and A. Chandrasekhar, "Secure key transport in symmetric cryptographic protocols using elliptic curves over finite fields," International Journal of Computer Applications, Vol. 36,1 November 2011.

[14]. Mohsen Machhout eat.al., "coupled FPGA/ASIC Implementation of elliptic curve crypto- processor," International Journal of Network Security & its Applications Vol. 2 No. 2 April 2010.

[15]. M. Kumar, "An enhanced remote user authentication scheme with smart card," International Journal of Network Security 10(3) (2010) 175–184.

[16]. Ram Ratan Ahirwal, Manoj Ahke Elliptic Curve Diffie-Hellman Key Exchange Algorithm for Securing Hypertext Information on Wide Area Network, IJSCIT, Vol 4(2).