International Journal For Technological Research In Engineering
Volume 1, Issue 8, April-2014 *ISSN (Online): 2347 - 4718*

SECURITY ENHANCEMENT IN DIFFIE-HELLMAN ALGORITHM

Santosh Prabhakar[1], Harshvardhan Mathur[2]
M.Tech (CSE) [1], Asst. Prof (CSE) [2]
Sobhasaria Group of Institutions, Computer Science and Engineering Department
Sikar (Raj), India.

Abstract: **Ensuring secure communication over internet is extremely challenging because of the dynamic nature of the network and the lack of centralized management. For this reason, Diffie-Hellman key agreement protocol is introduced for secure communication over network, but Diffie-Hellman key agreement protocol implementations have been plagued by serious security flaws. The attacks can be very subtle and, more often than not, have not been taken into account by protocol designers. In this Paper we discuss both theoretical attacks against the Diffie-Hellman key agreement protocol and attacks based on implementation details and also we have used a random parameter to make this algorithm more efficient. The random parameter generates new shared keys for each message that is exchanged between sender and receiver. So, different cipher text will be produced each time even for the same message. Thus, systems using this scheme will become more tolerant to various attacks.**

I. INTRODUCTION

Cryptography originated many years ago. During its early stages, two parties had to rely on a secure channel for exchanging a secret key which was used both for encryption and decryption. This scheme is known as private or symmetric key cryptography. However there were faults in the security of such a scheme. To ensure complete security, the key should be at least as long as the message as proved by Claude Shannon in the 1940s. Also, a secure channel is not always available which is why we need such an encryption scheme in the first place. These drawbacks can be overcome by using public key cryptography. This scheme enables two communicating parties to agree upon a shared secret key without exchanging it over the communication channel. Instead it is derived from parameters all of which are not publicly known. In 1976, Whitfield Diffie and Martin Hellman who were influenced by the work of Ralph Merkle on public key distribution, proposed an algorithm for key exchange which uses exponentiation in a finite field. Today, Diffie Hellman is used in a variety of protocols and services. It is used in interactive transactions, rather than a batch transfer from a sender to a receiver. The algorithm is used when data is encrypted on the Web using either SSL or TLS and in VPN. So its security is of utmost importance. However, like other security algorithm it is vulnerable to various attacks like plaintext attacks, man- in the middle attacks etc. So we propose a modification of the original algorithm so as make it more tolerant and secure by using a random parameter. Plaintext attacks are one of the most commonly used cryptanalysis methods. Samples of both the plaintext and cipher text are available to the attacker. The attacker can deduce more crucial information such as secret keys and code books from such a collection of known plaintext and cipher text. Due to the random parameter introduced in the proposed algorithm, the possibility of such a known plaintext attack is greatly reduced

II. THE DIFFIE-HELLMAN KEY EXCHANGE ALGORITHM(EQUATION FORMATION

1. Global Public Elements: Prime number q; $\alpha < q$ and α is a primitive root of q.
2. User A Key Generation:

User B Key Generation

3. Select private X_A $X_A < q$
   Select private X_B $X_B < q$
4. Calculate public Y_A $Y_A = \alpha^{XA} \bmod q$
   Calculate public Y_B $Y_B = \alpha^{XB} \bmod q$
5. Calculation of Secret Key by User A: $K = (Y_B)^{XA} \bmod q$
   Calculation of Secret Key by User B: $K = (Y_A)^{XB} \bmod q$

The result is that the two sides have exchanged a secret value. Furthermore, because XA and XB are private, an adversary only has the following ingredients to work with: q, a, YA, and YB. Thus, the adversary is forced to take a discrete logarithm to determine the key. For example, to determine the private key of user B, an adversary must compute XB = dloga, q(YB).The adversary can then calculate the key K in the same manner as user B calculates it. The security of the Diffie-Hellman key exchange lies in the fact that, while it is relatively easy to calculate exponentials modulo q prime, it is very difficult to calculate discrete logarithms. For large

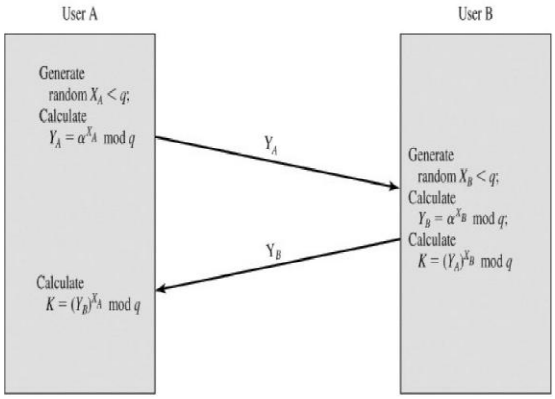

Fig. 1. Diffie-Hellman calculations and exchange.

primes, the latter task is considered infeasible Figure 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 X_A, calculate Y_A, and send that to user B. User B responds by generating a private value X_B calculating Y_B, and sending Y_B 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.

III. ENHANCED DIFFIE-HELLMAN KEY EXCHANGE ALGORITHM

Our proposed algorithm is improved from simple diffie-hellman as there are some drawback in diffie-hellman that is while the private key (K) is calculated then the result is always between 1 to 7 which can be easily decode like if we add A letter with +5 then it becomes E which can be easily understood. But in this algorithm our private key is (pow) which value is mostly higher which make change the original text with some sort of symbols. That is our private key is more secure than the simple diffie-hellman. Everyone can easily find private key in earlier algorithm but in this algorithm it is hard and difficult to find the private key.

A. Sender end

Step 1: Xa < q (user can select any random number less than q)
Step 2: Ya = a ^ Xa mod q (Ya is a public key of sender)
Step 3: K = Yb ^ Xa mod q (where Yb is a public key of receiver K is a private key)
Step 4: pow = 2 ^ K
Step 5: pow = pow + q
Step 6: Encrypt every letter of plain text using pow

B. Receiver end

Step 1: Xb < q (user can select any random number less than q)
Step 2: Yb = a ^ Xb mod q(Yb is a public key of receiver)
Step 3: Kp = Ya ^ Xb mod q (where Ya is a public key of sender K is a private key)
Step 4: pow = 2 ^ K
Step 5: pow = pow + q
Step 6: Decrypt every letter of cipher text using pow

Where ,

1) q is a prime number.
2) a is a root of prime number q.

IV. CONCLUSION

The Diffie – Hellman key exchange algorithm is an enabling technology for nearly every encryption technology in use in the Internet today, including SSL, SSH, IPSec, PKI, and everything else that depends on these protocols. The diffie – Hellman key exchange algorithm has proven to one of the most interesting key distribution schemes in use today. However, one of the fact that although the algorithm is safe against passive eavesdropping, it is not necessarily protected from active attacks (where by an intruder impersonates one of the parties involved in the exchange)

The purpose of this Research is to provide some solution to better encryption algorithms and try to provide better security to email services and to other web services etc. Our research could provide great solutions for web services like email transmission as we have enhance the encryption process and if any case someone broke into those email services, will only get a hard encrypted copy of data which very difficult to decrypt without information of our research algorithm. Our algorithm could prove to be the vision for both online and offline email security services. However our research lacks little in providing solutions to attacks like man in middle attack. But to cover up those types of issues, we develop our algorithm in such a way so that it can provide security to vendors even if they do hacked. The advantage of enhanced public key encryption algorithm is that it is hard to find the private key and it is a public key not symmetric key, every time private key is generated which is based on public key which is not fix. The security of the Diffie-Hellman key exchange lies in the fact that, while it is relatively easy to calculate exponentials modulo *q* prime, it is very difficult to calculate discrete logarithms. For large primes, the latter task is considered infeasible.

REFERENCES

[1] Zhen Cheng, Yufang Huang and Jin Xu, "Algorithm for Elliptic Curve Diffie-Hellman Key Exchange Based on DNA Tile Self-assembly", IEEE COMMUNICATIONS LETTERS, VOL. 10, NO. 9, MAY 2008.
[2] Authenticated Diffie–Hellman key agreement protocol using a single cryptographic assumption L. Harn, W.-J. Hsin and M. MehtaIEE Proc.-Commun., Vol. 152, No. 4, August 2005
[3] Standard specifications for public key cryptography, IEEE standard, pI363, 2000.
[4] Y. Kim, A. Perrig and G. Tsudik, "Tree-based Group Key Agreement,"ACM Transactions on Information and System Security (TISSEC), Vol. 7/1, Feb. 2004, pp. 60-94, doi: 10.1145/984334.984337.
[5] S. Anahita Mortazavi, Alireza Nemaney Pour and Toshihiko Kato, "Efficient Many-to-Many Group key Management Protocol", 2011 International Conference on Information and Computer Applications (ICICA 2011).
[6] Dongfang Zhang, "A New Authentication and Key Agreement Protocol of 3G based on Diffie-Hellman Algorithm", IEEE Commun. Lett., vol. 9, pp. 198-200,Feb. 2010.

