ISBN 978-93-5156-328-0
International Conference of Advance Research and Innovation (ICARI-2015)

Secure Key Exchange in Diffie Hellman Key Exchange Algorithm

Kamal Kr. Gola [a *], Rahul Rathore [a], Vaibhav Sharma [a], Manisha Kandpal [b]
[a] Department of Computer Science, Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India
[b] Department of Electronics & Communication Engineering, Jaipur National University, Jaipur, India

Article Info
Article history:
Received 9 January 2015
Received in revised form
15 January 2015
Accepted 22 January 2015
Available online 31 January 2015

Keywords
Public Key,
Private Key,
Secret Key,
RSA encryption and decryption algorithm,
RSA Digital signature Scheme

Abstract

As we know that Diffie-hellman was the first published public key algorithm that is used for secure key exchange mechanism. The purpose of this algorithm is to exchange a key that can be used for encryption and decryption. But this algorithm is no longer strong, since the key can be easily identified during communication. To overcome this problem this work proposed a technique to generate private keys and public key using RSA encryption decryption technique. In this work a secret key will be selected by one user and that key will be send to the other user which will be encrypted with two keys so it is difficult for the intruder to identify the key. For authentication purpose this work uses the concept of RSA digital signature scheme.

1. Introduction

The Diffie-Hellman key exchange protocol is a cryptographic protocol that was developed by Whitfield Diffie and Martin Hellman in 1976, although it was later alleged that it had been separately invented a few years earlier within GCHQ, the British signals intelligence agency, by Malcolm J. Williamson but was kept classified .It was the first published public key algorithm in the ground-breaking paper "New Directions in Cryptography" that define the public key cryptography [1].

This algorithm is an anonymous i.e. non-authenticated, key-agreement protocol that provides the basis for a variety of authenticated protocols which is used to provide perfect forward secrecy in Transport Layer Security's short-lived modes. It is the example of the key exchange implemented within the field of cryptography .The motive of this protocol is to enable two users that have no prior knowledge of each other to securely exchange a secret value over an insecure channel (i.e. not protected from the interception but is protected from modification) and then agree on the secret key, if both the party computes the same value for the key. And that key will be used for the encryption of the message using a symmetric key cipher [2]. Firstly both the parties agree on a non secret value i.e. public key which may be certified so that the parties can be authenticated and there may be a combination of these attributes. This algorithm is only used and limited to exchange the secret values.

2. Traditional Algorithm

The process of Diffie-Hellman algorithm is described as follow:

i. Both parties A and B agree upon two constants p and g. Where p is a prime number and g is the generator less than p.
ii. Both A and B choose their private keys a and b respectively such that they are random numbers and less than p.
iii. Let g^a mod p and g^b mod p be the public keys of A and B respectively.
iv. Then A and B exchange their public keys over insecure medium like internet.
v. Then party A computes $(g^b \bmod p)^a \bmod p$ that is equal to $g^{ba} \bmod p$.
vi. Also party B computes $(g^a \bmod p)^b \bmod p$ that is equal to $g^{ab} \bmod p$.
vii. The shared secret key K is computed as $K = g^{ba} \bmod p = g^{ab} \bmod p$.

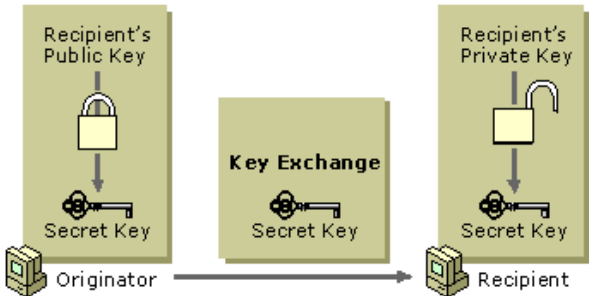

Fig: 1. Diffie- Hellman key Exchange Algorithm

3. Literature Review

Eun-Jun Yoon et al. [3] proposed an efficient Diffie-Hellman-MAC key exchange scheme providing same securities as proposed by Jeong et al. who proposed a strong Diffie-Hellman- DSA key exchange scheme providing security against session state reveal attacks as well as forward secrecy and key independence. The proposed scheme is based on the keyed MAC hash function to provide efficiency.

Emmanuel Bresson et al. [4] has investigated the Group Diffie-Hellman protocols for authenticated key exchange (AKE) are designed to provide a pool of players with a shared secret key which may later be used, for example, to achieve multicast message integrity. Over the years, several schemes have been offered.

Corresponding Author,
E-mail address: kkgolaa1503@gmail.com

SANS Institute Info Sec Reading Room [5] has investigated the overview of the Diffie-Hellman Key Exchange algorithm and review several common cryptographic techniques in use on the Internet today that incorporate diffie-Hellman. The privacy requirements for users normally described in the traditional paper document world are increasingly expected in Internet transactions today. Secure and safe digital communications are very much necessary part for web-based e-commerce, mandated privacy for medical information, etc. In simple scenario, secure and safe connections between different parties which are communicating over the Internet are now a requirement. Whitfield Diffie and Martin Hellman founded the protocol which can provide secure connection which gain popularity as "Diffie-Hellman (DH)" algorithm in 1976.

Michel Abdalla [6] discussed a Diffie-Hellman based encryption scheme, DHIES (formerly named DHES and DHAES), which is now in several (draft) standards. The scheme is as efficient as ElGamal encryption, but has stronger security properties. Furthermore, these security properties are proven to hold under appropriate assumptions on the underlying primitive. DHIES is a Diffie-Hellman based scheme that combines a symmetric encryption method, a message authentication code, and a hash function, in addition to number-theoretic operations, in a way which is intended to provide security against chosen cipher text attacks. The proofs of security are based on the assumption that the underlying symmetric primitives are secure and on appropriate assumptions about the Diffie-Hellman problem.

Vishal Garg and Rishu[7] proposed work to provide harder encryption with enhanced public key encryption protocol for security and proposed work can be implemented into any network to provide better security. This work enhanced the hardness in security by improving the Diffie-Hellman encryption algorithm by adding some more security codes in current algorithm.

Akhil Kaushik and Satvika [8] proposed algorithm is also based on Diffie-Hellman algorithm, which uses a new technique for sharing session keys. The proposed "Extended Diffie-Hellman Algorithm" uses a digital image to produce random numbers for exchanging keys over insecure network. In this paper, a new public key cryptosystem is proposed namely "Extended Diffie-Hellman Algorithm for Key Exchange". This proposed algorithm is different from the Diffie-Hellman Algorithm in two ways:

i. It suggests a new method to generate true random numbers based on images, which is quite convenient, simple and cost-effective.
ii. It recommends a new technique for calculation of shared secret key.

4. Proposed Algorithm

4.1 Key generation at the sender side (Public Key and Private Key)

a) Select p and q with the condition that p and q both prime and p does not equal to q.
b) Calculate n=p*q
c) Calculate Ø (n) = (p-1) * (q-1)
d) Select integer e gcd (Ø (n), e) =1; 1<e<Ø (n)
e) Calculate d e*d=1 mod Ø (n)
f) Public key (e, n)
g) Private Key (d, n)

4.2 Key generation at the receiver side (Public Key and Private Key)

a) Select p1 and q1 with the condition that p1 and q1 both prime and p1 does not equal to q1.
b) Calculate n1=p1*q1
c) Calculates Ø (n1) = (p1) -1 * (q1) -1
d) Select integer e1 gcd (Ø (n1), e1) =1; 1<e1<Ø (n)
e) Calculate d1 e1*d1=1 mod Ø (n1)
f) Public key (e1, n1)
g) Private Key (d1, n1)

4.3 Sharing of Secret Key

i. First Sender selects a secret key that is only known to sender. Before sending the secret key to the receiver, sender generate the digital signature using his/her private key $S = K^d \bmod n$ and then encrypt the key using receiver's public key $K_1 = K^{e1} \bmod n1$.
ii. Now sender sends the encrypted key and digital signature to the receiver.
iii. Now Receiver receives the encrypted key and digital signature and perform the decryption process using $K = K_1^{d1} \bmod n1$ where K_1 is the encrypted key, d1 is the private key of the receiver and K is the original key.
iv. Now receiver performs the verifying process using $S_1 = S^e \bmod n$. Where S_1 is a copy of the K, S is the signature and e is the public key of the sender. If $S_1 = K$ then the signature is verified.

5. Implementation

Step: 1. Key generation process (at sender side)
The first sender selects two prime numbers given as p and q that are known to sender only.
P=7 and q=17
Now it will calculate the value of n and Ø (n)
Values of n will be calculated by n=p * q.
n=7*17=119
Now calculate the value of Ø (n) = (p-1) * (q-1).
Ø (n) =6 * 16=96
Now the sender will choose public key e such that e < Ø (n) and GCD (e, Ø (n)) =1.
e =5.
Now sender calculates the private key d using given expression.
e*d=1modØ (n)
(5 * d) mod 96=1
d=77.
Step: 2. Key generation process (at receiver side)
Now the receiver selects two prime numbers p1 and q1 that are only known to the receiver only.
p1=17 and q1=11
Now it will calculate the value of n1 and Ø (n1)
Values of n will be calculated by n1=p1 * q1.
n1=17*11=187
Now calculate the value of Ø (n1) = (p1) -1 * (q1) -1.
Ø (n1) =16 * 10=160
Now the receiver will choose public key e1 such that e1 < Ø (n1) and GCD (e1, Ø (n1)) =1.
e1 =7.
Now sender calculates the private key d1 using given expression.
e1*d1=1modØ (n)

(7 * d1) mod 60=1
d1=23.

Step: 3. Sharing of Secret Key

In proposed technique public key and value of n and n1 (sender and receiver) are known to each other. Now sender generate the digital signature using own private key (d).

Encryption Process

$S=K^d \bmod n$

Where S is a digital signature at the sender side.
K is the secret key which is selected by the sender.
d is a private key of the sender.
n is the multiplication of two prime numbers.
For implementation purposes lets K=88
$S=88^{77} \bmod 119$
$(88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (885 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^5 \bmod 119) * (88^2 \bmod 119)$
= (107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 107 * 9) mod 119
= (30 * 30 * 30 * 37) mod 119
= 999000 mod 119

S= 114 (Digital signature at sender side)

Before sending the secret key to the receiver, the sender encrypts the key using receiver's public key.
$K_1=K^{e1} \bmod n1$ where K is the secret key and e1 is the public key of the receiver.
$K_1=88^7 \bmod 187 = (88^3 \bmod 187) * (88^3 \bmod 187) * (88 \bmod 187) = (44 * 44 * 88) \bmod 187$
= 170368 mod 187

K_1 = 11(Encrypted key at sender side)

Decryption process

Now Receiver receives the encrypted key and the signature and perform the decryption process using $K=K_1^{d1} \bmod n1$ where K is the original secret key, d1 is the private key of the receiver and K_1 is the encrypted key.
$K= 11^{23} \bmod 187$
$= (11^4 \bmod 187) * (11^4 \bmod 187) *(11^4 \bmod 187) *(11^4 \bmod 187) *(11^4 \bmod 187) *(11^3 \bmod 187)$
= (55 * 55 * 55 *55 *55) mod 187 *(113 mod 187)
= (33 *33*55*11*11*11) mod 187

K=88 it proved that the key is confidential.

Verifying process

Now receiver performs the verifying process using $S_1= S^e \bmod n$. Where S_1 is a copy of the K, S is the digital signature and e is the public key of the sender. If S_1=K then the signature is verified.
$S_1= 114^5 \bmod 119$
$= (114^2 \bmod 119) * (114^2 \bmod 119) * (114 \bmod 119)$
= (25 * 25 * 114) mod 19 = 71250 mod 119

S_1= 88=K its provide the authentication.

6. Result and Conclusion

In this work a new approach has been proposed for sharing the secret key as well as to generate a public key and private key. By doing implementation this scheme showed that the proposed technique provides the security during key transfer as compared to existing Diffie- Hellman key exchange algorithm. In the existing scheme, there is no security for the key. Anyone who knows the public key can access the secret key that means it does not provides the confidentiality, authentication and the key security, but the proposed scheme provides the key security as well as user authentication. The implementation results show that the proposed technique has improved the security and performance of Diffie-Hellman key exchange algorithm, while providing high quality of service and security for desired scheme.

References

[1] W. Stallings, Cryptography and Network Security Principles and Practices, 4th Edition, Pearson Education.
[2] B. A. Forouzan, Cryptography and Network Security, Special Indian Edition, The McGraw- Hill companies, New Delhi, 2007
[3] Eun-Jun Yoon, Kee-Young Yoo, An Efficient Diffie-Hellman-MAC Key Exchange Scheme, 2009 Fourth International Conference on Innovative Computing, Information and Control
[4] E. Bresson, O. Chevassut, D. Pointcheva, Jean-Jacques Quisquater, Authenticated Group Diffie-Hellman Key Exchange, Computer and Communication Security-proc of ACM CSS'01, Philadelphia, Pennsylvania, USA, ACM Press, 2001, 255-264
[5] SANS Institute Info Sec Reading Room, A Review of the Diffie-Hellman Algorithm and its use in Secure Internet Protocols
[6] M. Abdalla, M. Bellare, P. Rogaway, DHIES: An encryption scheme based on the Diffie-Hellman Problem, In Proc. of ACM CCS '01, ACM Press, 2001
[7] V. Garg, Rishu, Improved Diffie-Hellman Algorithm for Network Security Enhancement, IJCTA, 3(4), 2012
[8] A. Kaushik, Satvika, Extended Diffie-Hellman Algorithm for Key Exchange and Management, International Journal of Advances in Engineering Sciences, 3(3), 2013