ISSN 2229-5518

Authentication of DSS and Secrecy

Shafiqul Abidin, Dr. Kumar Balwant Singh

Abstract— The Diffie-Hellman key algorithm was the first proposed public key algorithm by which two parties can communicate with each other without having any prior knowledge of each other over an insecure communication channel. This algorithm was first published by Whitfield Diffie and Martin Hellman in 1976. It is helpful in a variety of authenticated protocols. Further, it is also found suitable in transport layers. After his investigation this protocol was integrated to digital signature algorithm. In this paper we are presenting the verification of digital signature standard with the help of Diffie-Hellman key exchange protocol by using two randomly selected integers. We also tried to show that the use of protocol can provide the secrecy and freshness of key due to continuous selection of random numbers.

Index Terms— Digital Signature Standard, Diffie Exchange Algorithm, Random Number, Authenticated Key Distribution, Phan's Integration.

1 Introduction

Very first time Arazi proposed a scheme for the integration of Diffie-Hellman key exchange protocol with digital signature [1]. After a decade Harn et.al gave some improvement in the algorithm so that it can be prevented by known key attack, key relay attack and unknown key attack [2]. Finally Phan modified this scheme with two major attributes i.e. forward secrecy and key freshness [3]. In this paper we have taken the concept of key freshness by using the randomly selected number and verified the digital signature standard.

2 Background

For this protocol we need to know that two publicly known numbers i.e. p and g which will be the primitive roots of p. Suppose there are two users A and B which want to exchange a key and unknown to each other. First time user A select a random Integer $X_A < p$ and compute $Y_A = g^{X_A} \bmod p$. Similarly, user B selects a random number $X_B < p$ and computes $Y_B = g^{X_B} \bmod p$. Each side keeps the value of X privately and make the Y value available publicly to the other side so this is called public key for both of users A and B and the process is known as the public key generation for A and B. Now user A computes the key as $K = (Y_B)^{X_A} \bmod p$ and user B computes the key $K = (Y_A)^{X_B} \bmod p$. These two values should produce similar result.

TABLE 1
PUBLIC KEY GENERATION

$$
\begin{aligned}
K &= (Y_B)^{X_A} \bmod p \\
&= (g^{X_B} \bmod p)^{X_A} \bmod p \\
&= (g^{X_B})^{X_A} \bmod p \\
&\qquad \text{(By the rule of modular arithmetic)} \\
&= g^{X_B X_A} \bmod p \\
&= (g^{X_A})^{X_B} \bmod p \\
&= (g^{X_A} \bmod p)^{X_B} \bmod p \\
&= (Y_A)^{X_B} \bmod p
\end{aligned}
$$

The result shows that the two sides can change the secret value and both values are identical with each other.

3 Related Work

The Diffie-Hellman key algorithm was the first proposed public key algorithm by which two parties can communicate with each other without having any prior knowledge of each other over an insecure communication channel proposed by Harn.et.al. Diffie-Hellman key exchange algorithm is the most famous algorithm to exchange keys over a network but it has some false and drawbacks.

For the establishment of the communication the parties require the session key which can be generated by the key establishment protocol and it can also be referred as the key agreement protocol. Diffie and Hellman developed the first most popular key agreement protocol based on asymmetric encryption or public key cryptography [3].

They proposed the two versions of protocols. In the first protocol all the entities in the communication network exchange the static public keys. In this case there is a drawback that the

- *Shafiqul Abidin is pursuing Ph. D. in computer science / IT from B.R. A. Bihar University, Muzaffarpur, India, E-mail: shafiqulabidin@yahoo.co.in*
- *Dr. Kumar Balwant Singh is currently associated with the department of Physics, Government Polytechnic, Dharbhanga, Bihar, India. E-mail: kbsphysics@yahoo.co.in*

entities A and B compute the same session key for each run of protocol. But in the second case they exchange the ephemeral public keys which are vulnerable to the man-in-middle attacks.

3.1 Phan's Integration with Digital Signature Algorithm

Phan [4] integrated these computation used in Diffie-Hellman key exchange protocol in to Digital Signature algorithm as follows:

TABLE 2
COMPUTATION FOR DIGITAL SIGNATURE ALGORITHM

Step	User A	User B
1	Select random integer v $m_A = g^v \bmod p$ $n_A = (y_A)^v \bmod p$	
	m_A, n_A →	
2		Select random integer w $K_{AB} = (m_A)^{x_B w} \bmod p = g^{x_B vw} \bmod p$ $K_{BA} = (n_A)^w \bmod p = g^{x_B vw} \bmod p$ $m_B = g^w \bmod p$ $n_B = (y_B)^w \bmod p$ $r_B = m_B \bmod q$ $s_B = ((w)^{-1}(H(m_B \| K_{BA} \| K_{AB}) + x_B r_B)) \bmod q$
		← (m_B, n_B, s_B)
3	$K_{AB} = (n_B)^v \bmod p = g^{x_B vw} \bmod p$ $K_{BA} = (m_B)^{x_A v} \bmod p = g^{x_B vw} \bmod p$ $r_B = m_B \bmod q$ Verify DSA signature (r_B, s_B) $r_A = m_A \bmod q$ $s_A = ((v)^{-1}(H(m_A \| K_{AB} \| K_{BA}) + x_A r_A)) \bmod q$	
	S_A →	
4		$r_A = m_A \bmod q$ Verify DSA signature (r_A, s_A)

4 PRPPOSED WORK

The above integration done by the Phan was for authenticated key distribution. We only review and analyze the work done by Phan and tried to prove the correctness of digital signature standard algorithm with the integration of Diffie-Hellman key exchange protocol [5]. In the DSS approach of digital signature we have the integration with Diffie –Hellman a protocol as follows.

The hash code is provided as input to a signature function along with a random number (can be selected as randomly as we did in Diffie-Hellman algorithm) K is generated for signature [6]. This signature function also depends upon the sender A's privately selected key X_A and globally selected integer g.

If the result of digital signature consists of two component s and r then the output of the verification function is a value that will be equal to the value considered in sighing function 'r' if signature is valid [7].

Suppose there are three parameters (public key elements) p, q, g.

Where

p is a prime number selected with a length bit 512 & 1024 bit. i.e. $512 \le L \le 1024$ & L is multiply of 64
q divides (p-1) & bit length 160 bits (160 prime number)
g is the $h^{(p-1)/q} \bmod p$ where h is any integer.

Now user select a private random integer number as in Diffie-Hellman protocol $\mathbf{X_A < p}$ by this we can generate the user's public key $\mathbf{Y_A = g^{X_A} \bmod p}$.In the DSS approach user can select a per-message secret number k which is randomly selected by any algorithm of random number generation also known as the nonce but it should be $0 < k < q$.

4.1 Signing Function

$r = (g^k \bmod p.) \bmod q$

$s = [K^{-1} H(M) + X_A r] \bmod q$

Signature will be (r, s)

4.2 Verifying Function

$w = (s')^{-1} \bmod q$

$v_1 = [H(M') w] \bmod q$

$v_2 = (r') w \bmod q$

$r' = [(g^{v1} Y_A^{v2}) \bmod p] \bmod q$

Where M = Message to be signed

H (M) = Hash of using SHA-1

M' r's' = Received version of M, r, s.

5 RESULT

For the verification and for the Prof. of correctness we have to prove that r = r' so that its correctness can be prove as the same message received by the receiver send by the sender. In the verifying step.

$r' = [(g^{v1} Y_A^{v2}) \bmod p] \bmod q$

$= (g^{H(M) w \bmod q} Y_A^{r w \bmod q} \bmod p) \bmod q$

$= (g^{H(M) w \bmod q} g^{X_A r w \bmod q} \bmod p) \bmod q)$

$= (g^{H(M) w + X_A r w \bmod q} \bmod p) \bmod q$

$= (g^{(H(M) + X_A) w \bmod q} \bmod p) \bmod q$

$= (g^{(((H(M) + X_A r) k H(M) + X_A r)^{-1} \bmod q} \bmod p) \bmod q$

$= (g^k \bmod p.) \bmod q$

$= r$

This correctness provides the authentication to the sender and receiver that a message which is signed is intended for the appropriate users.

6 CONCLUSION

In this paper we provide some further cryptography and analysis on the Phan's integration of DSA and DIffie- Hellman Key exchange protocol. We present an improvement on protocol with the help of two randomly selected integers which makes the protocol more secure. Also these random numbers can provide two basic attribute for key exchange protocol i.e. 1) The forward secrecy as we have chosen the different random number each and every time as X_A and X_B which are private to both of users.2) Key Freshness as the public key of both user does not depend on each other it depends on the randomly selected value by users so key freshness can be maintained i.e. with the help of these random numbers we can generate a new key for every communication.

REFERENCES

[1] A. Arazi, "Integrating a key cryptosystem into the digital signature standard," Electronic Letters, vol.29, pp. 966-967, Nov. 1993.

[2] L. Harn, Ma. Mehta, and W. J. Hsin, "Integrating Diffie-Hellman Key Exchange into the Digital Signature Algorithm (DSA)," IEEE Communication Letters, vol. 8, no. 3, Mar. 2004.

[3] W. Diffie and M. Hellman, "New Directions in Cryptography", IEEE Transactions on Information Theory, Vol. IT-l 22, No.6, November,1976, PP.644-654.

[4] R. C. W. Phan, "Fixing the integrated diffie-Hellman - DSA key exchange protocol," IEEE Communication Letters, vol. 9, no. 6, Jun. 2005.

[5] Charanjit S. Jutla and Anindya C. Patthak. Is SHA-1 conceptually sound? Cryptology ePrint Archive, Report 2005/350, 2005.

http://eprint.iacr.org/.

[6] M. Matsumoto and T. Nishimura, "Weight Discrepancy Tests on M-sequences", Bulltin of Yamagata University (Natural Science), Vol. 16, No.3, 2007, 105--112.

[7] "Cryptography and Network Security" by William Stalling Fourth Edition.