http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 6 (2): 259-262, 2007
ISSN 1812-5638

Efficient Scheme for Obtaining Public Key Cryptosystem Using Shared Secrets

Sattar J. Aboud
Department of Computer Science, Faculty of IT, University for Graduate Studies, Amman-Jordan

Abstract: This research describes the combination of Shamir secret sharing method with the trap door characteristics of discrete logarithm as employed in the Diffie-Hellman exchange scheme. The purpose of this combination is to protect the privacy of the numbers held by the users of the shared secret, while at the same time authorizing them to broadcast data in the clear which, when published by adequacy of the users, enables mutually authenticated base key exchange between the users.

Key words: Public key encryption, Shamir secret sharing, discrete logarithm, diffie-hellman exchange key, signature scheme

INTRODUCTION

Secret sharing schemes is a method of sharing a message m among a set of n participants such that any subset consisting of t participants can reconstruct the message m, but no subset of smaller size can reconstruct. Secret sharing methods have numerous practical uses. For example, they can be employed to control the access to protect so only an authorized subgroup of bank workers can open it by grouping their shares together and rebuilding the secret combination which unlocks the protection. The key investigation regarding the understanding of a secret sharing scheme was considered by many researchers (Simmons, 1994; Jackson *et al*., 1995; Charnes *et al*., 1997) suggested various solutions for building such systems.

Threshold schemes are originally invented by (Shamir, 1979). Shamir secret sharing method suggests a method of distributing a secret number between numerous users. Simply after some minimum number of users cooperates can the secret number be rebuild. The (Diffie and Hellman, 1976), public key distribution method presents a technique for two users to agree on a secret number among them, while all their exchanges are transmitted over a public way.

In this research we propose a new scheme for obtaining public key cryptosystem using the combination of both Shamir secret sharing scheme and the trap door characteristics of discrete logarithm as employed in the Diffie-Hellman (1976) secret number exchange scheme. The object of this proposed scheme is to protect the secrecy of the numbers held by the users of the shared secret, whilst at the same time permitting them to send information in the clear which, when issued by sufficiency of the individuals enables mutually authenticated base key exchange, between the users.

THE PROPOSED SCHEME

The polynomial of degree n in one parameter is entirely defined when n + 1 points in which it sends are specified. This is factual if the coefficients related to any field and in particular to GF (p) the field of integer. If these points are (x_i, z_i), for i = 1,..., n + 1, the value of z according to any number of x is specified by the Lagrange interpolation theorem:

$$z = \sum_{i=1}^{n+1} w_i * z_i \bmod p \quad (1)$$

Such that:

$$w_i = \prod_{j=1}^{n+1} (x - x_j)/(x_i - x_j) \bmod p$$

$$\text{For } i = 1, \ldots, n+1 \text{ Where } j \neq i \quad (2)$$

In this suggested method the ordinates x are public number and since w_i base only on them, these are also public. It must be noticed that w_i, x and z, the w_i results from numerical operations, p and x are also members of the field GF (p), i.e., are both integers. The divisions needed in their derivation are achieved by employing the Euclidean method Trappe and Washington (2006). The ordinates z on the other hand, remain secret by the users and by no means directly sent by them. Instead they are employed as key exponents to raise certain public base value h to various powers to give a new set of values s. This exponentiation is performed by mod operation with another prime number q, the base value h and the out comings s`s are consequently members of the field GF (q). So

$$s_i = h^{zi} \bmod q \quad (3)$$

According to Menezenes *et al.* (1997). The exponentiations can be accomplished efficiently employing the repeated square and multiply algorithm, which is commonly employed in public key encryption and illustrated for instance by RSA (Rivest, 1978). While in the Diffie-Hellman public key distributed algorithm given q is a well selected prime knowledge of h, q and the s`s does not disclose the matching z`s because logarithm mod a well selected prime is a computationally difficult to solve. The prime numbers p and q are selected so that both are related to each other by the following formula:

$$q = g^* p + 1 \tag{4}$$

When p is selected as a prime number, g is handily chosen from the sorting order of even numbers 2, 4, 6,... is the least positive integer which produces q prime. This relationship has two aims:

- To ensure that q−1 has large prime factor which is an essential condition for guaranteeing that logarithm mod q is computationally difficult to solve. According to Menezenes Vanstone (1997). The well-known method is shanks method which needs $o(\sqrt{q})$ operations to achieve the logarithm.
- To offer a technique of efficiency achieving the operations among the exponents z needed by formula 1 via performing on the matching s numbers. Additions among z values are substituted by multiplications among s values. Multiplications of z values by known key multipliers w_i are substituted by raising the s values to the power of these public multipliers. To achieve this properly, it is essential to limit the selection of base integer h as follows:

$$h = a^g \bmod q, \text{for } 2 \le a \le q-1 \text{ and } h \ne 1 \tag{5}$$

By Fermat's Theorem:

$$a^{q-1} \equiv 1 \bmod q \text{ for any } a \ne v \bmod q \tag{6}$$

Hence

$$h^p = (a^g)^p = a^{(q-1)} = 1 = h^v \bmod q \tag{7}$$

$$\text{i.e., } h^p = h^v \bmod q \tag{8}$$

The exponents of h act as integer mod p It can be noted that this constraint still allows selection of p−1 which is another possibility working numbers for h. If h is raised correspondingly to the power of every part of formula 1, the following relationship is computed.

$$s = h^z = h^{(\sum_{i=1}^{n+1} w_i * z_i)} = \prod_{i=1}^{n+1} h^{(w_i * z_i)} = \prod_{i=1}^{n+1} (h^{z_i})^{w_i} \tag{9}$$

$$\text{i.e. } s = \prod_{i=1}^{n+1} s_i^{w_i} \bmod q \tag{10}$$

Formula 10 is a relationship among n + 2 different s`s every of which is resulting from one of the n + 2 matching z`s which demonstrate the relationship given in formula 1. Every user provided with a sum of n + 2 numbers of s can test if formula 10 is achieved. When it is achieved, then the n + 2 numbers of s and therefore their providers are real.

MUTUAL AUTHENTICATION

This application is more general knowing an exponent key distribution system with mutual authentication of all users. So in this application all users have equal situation.

Assume the total number of users is n, for i = 1 to n user i is published with two (x, z) pairs a transmission pair (x_{i1}, z_{i1}) and an authentication pair (x_{i2}, z_{i2}). From these the user can compute two (x, s) pairs, a transmission pair (x_{i1}, s_{i1}) and an authentication pair (x_{i2}, s_{i2}) .

The minimum number of operative users in a polynomial of degree n, is n + 1, if every of n + 1 operative users passes his transmission (x, s) pair, then every operative user will know n + 2 (x, s) pairs, by the n + 1 transmission pairs together with his own authentication pair. So every user individually can verify for satisfaction with formula 10 to authenticate the other users.

Having generated the authenticity of all the other users i, j can connect as in the Diffie-Hellman method employing the base number:

$$g_{ij} = h^{(z_{i1} * z_{j1})} \tag{11}$$

In which i, j computes:
Note: that s_{j1} is publicly sent by j and z_{i1} is i`s private key.

$$g_{ij} = (h^{z_{j1}})^{z_{i1}} = (s_{j1})^{z_{i1}} \tag{12}$$

And j computes:

$$g_{ij} = (h^{z_{i1}})^{z_{j1}} = s_{i1}^{z_{j1}} \tag{13}$$

Note: that s_{i1} is publicly sent by i and z`s is a private key.

By the security of the Diffie-Hellman method, it is computationally hard for any user other than i, j to solve g_{ij}. Once base numbers have been generated communicate among users, these connects can be employed if required to generate a net-wide base number. Once n + 1,s numbers are publicly known and any other s value can be

computed from them, so that the same base key should not be reemployed later. To avoid this it is recommended that the base key must combine in certain way a time stamp with date.

Example: An example of the method explained in section above is given below. It employs small prime numbers to illustrate the rule of algorithm. According to Stinson (2006) a number of 150 or more decimal digits can be employed. Assume that the number of operative users is 3 out of a total of 5. Choose a prime number p = 29. Next q = g* p + 1 = 59 compute prime with g = 2. A program is employed to choose a private polynomial of degree 2 with coefficients in GF (29). Assume it chooses:

$$z = 5 * x^2 + 8 * x + 3 \bmod 29 \quad (14)$$

Then, it chooses 10 ordinates x two for every of the 5 users. It computes privately from formula 14. The equivalent ordinates z employing the multiplicative inverse (Aboud, 2005) which is computed as follows:

	Transmission pair		Authentication pair	
User	x_{i1}	z_{i1}	x_{i2}	z_{i2}
1	16	19	14	22
2	6	28	12	7
3	5	23	9	16
4	13	24	7	14
5	8	10	19	17

The program then provides all users with records of each one's transmission pair x numbers (x_{i1}`s) and every user separately with his authentication pair x number (x_{i2}) and his private z keys (z_{i1}, z_{i2}). The program then deletes its memory to erase the details of the polynomial coefficients it has selected.

Assume that users 1, 3, 5 determine to generate base numbers for communication among them. They choose a number a = 11 and from it compute the base key:

$$h = a^g = 11^2 \bmod 59 = 3 \quad (15)$$

Then every computes his own transmission pair x_{i1}, s_{i1}, such that:

$$s_{i1} = 3^{z_{i1}} \bmod 59 \quad (16)$$

And the authentication pair x_{i2}, s_{i2}, such that:

$$s_{i2} = 3^{z_{i2}} \bmod 59 \quad (17)$$

User	Suffix	Transmission pair	Authentication pair
1	1	(16, 53)	(14, 15)
3	2	(5, 45)	(9, 26)
5	3	(8, 49)	(19, 19)

User 1 computes the authentication verification as follows. He computes w_i numbers according to his own authentication ordinates, i.e., $x_{i2} = 14$. Thus

$$w_1 = \frac{(14-5)*(14-8)}{(16-5)*(16-8)} = \frac{25}{1} = 25*30 = 25 \bmod 59 \quad (18)$$

$$w_2 = \frac{(14-16)*(14-8)}{(5-16)*(5-8)} = \frac{17}{12} = 17*17 = 28 \bmod 29 \quad (19)$$

$$w_3 = \frac{(14-16)*(14-5)}{(8-16)*(8-5)} = \frac{11}{5} = 11*6 = 8 \bmod 29 \quad (20)$$

Then user 1 computes employing formula 10 the value of his authentication s must take to calculate with the 3 transmitted s`s as follows:

$$\begin{aligned} s &= (53^{25} * (45^{28}) * (49^{8}) \bmod 59 \\ &= 29*21*15 \bmod 59 \\ &= 49 \end{aligned} \quad (21)$$

Since this matches with his authentication, user 1 can be certain that user 3 and user 5 are definitely the other users. User 3 and user 5 every one carries out similar computations to check the individualities of the other two. Then for instance user 1 and user 3 can communicate employing the base number.

$$g_{13} = 3^{(19*23)} \bmod 59 = 9 \quad (22)$$

Which user 1 computes as follows:

$$g_{13} = s_{31}^{19} = 45^{19} = 9 \bmod 59 \quad (23)$$

Note that is publicly transmitted by user 3 and the key = 19 is the user 1 private key.
And user 3 computes as follows:

$$g_{13} = s_{11}^{23} = 53^{23} = 9 \bmod 59 \quad (24)$$

Note that S_{11} is publicly transmitted by user 1 and the key = 23 is the user 3 private key.

Similarity the base numbers can be generated among user 1 and user 5 and among user 3 and user 5. Once this specific base key is employed its future utilize is no longer private. The z numbers however, hold their secrecy and can be reemployed with another base key.

DISCUSSION

In this method we assumed that the private z values are physically kept in the contributing stations. This

makes the scheme vulnerable to detained stations, thus it maybe suitable to have certain mnemonic system to allow users to remember their private keys. For the base number exchange method, it is preferable that users remember two private keys. Though, a lesser degree of security is obtainable if just one is considered.

It is possible for one user to masquerade another in the authentication operation. However, there is no benefit to him acting this, since the intruder can not figure out the base number for communication. So it is essential that all authorizing stations trust one another, if not one authorizing station could obtain a benefit by creation it appears that a new authorized a transaction rather than itself. Alternatives of the Shamir secret sharing method are studied by (Kaliaperumal, 2003) which have their similarities in the scheme examined. These give better authority to certain parties than others. It is potential to change the scheme to employ polynomials in many variables. This can give benefits of overview in practical completion and of expanding the possible uses of the system.

It is easy to enhance the base number exchange method if the minimum number of operative users is 2. Corresponding to the suggested method f (x) is a polynomial with 2 private coefficients, so we can write:

$$z = w * x + h \bmod p \tag{25}$$

Each user has two pairs and can therefore, figure out w and h and therefore the private number of all other users. To prevent this, a polynomial of the following structure can be employed:

$$z = w * x^3 + h * x^2 + a^x + d \bmod p \tag{26}$$

We mean that one have 4 private coefficients. Every user is published with 3 (x, z) pairs, two for transmission and one for authentication. Prior to transmission every user has 3 (x, z) points on a polynomial with 4 private coefficients and thus can not be working out the coefficients or other users private keys. After transmission every user has 5 (x, s) pairs and can carry out the authentication verification by employ the formula 10.

It is possible to modify the scheme to employ polynomials in more than one parameter. This can give benefits of simplification in practical implementations and of expanding the possible users of the system.

CONCLUSIONS

This research is described the Shamir secret sharing scheme combined with the trap door characteristics of discrete logarithm as employed in the Diffie-Hellman secret exchange scheme. This combination permits the users components of a shared secret key to be broadcasted over a public means in a concealed type, so that the privacy of the components and the total shared number are protected. This denotes that the same private key can be employed repeatedly, on future times. This scheme can be employed in many statuses needing the co-operation of t users out of n. The example, given illustrating that the scheme can be employed to achieve mutually authenticated base number exchange between users.

REFERENCES

Aboud, S.J., 2004. Baghdad Method for Calculating Multiplicative Inverse, international Conference on Information Technology, Las Vegas, Nevada, USA, pp: 816-819.

Charnes, C., K. Martin, J. Pieprzyk and R. Ssfavi- Naini, 1997. Remarks on the multiple assignment secret sharing scheme. Information and Communications Security, ICIS 97, Lecture Notes in Computer Science, pp: 72-80.

Diffie, W. and M. Hellman, 1976. New Directions in Cryptograph. IEEE Transactions on Information Theory, IT-22.

Jackson, W.A., K.M. Martin and C.M. O'Keefe, 1995. Efficient secret sharing without a mutually trusted authority. Advances in Cryptology, EUROCRYPT'95, Lecture Notes in Computer Science, pp: 183-193.

Kaliaperumal, S., 2003. Securing Authentication and Privacy in ad hoc Partitioned Networks. Applications and the Internet Workshops, Proceedings of Symposium, IEEE, 27-31, pp: 354-357.

Menezenes, P., V. Oorschot and S. Vanstone, 1997. Handbook of Applied Cryptography, ARC Press.

Rivest, R., A. Shamir and L. Adelman, 1978. A method for obtaining digital signatures and public key cryptosystems. Communication of the ACM., 21: 120-126.

Simmons, G.J., 1994. The consequences of trust in shared secret scheme. Advances in Cryptology, EUROCRYPT'93, Lecture Notes in Computer Science, pp: 448-452.

Shamir, A., 1979. How to share a secret. Communications of the ACM, 22: 612-613.

Stinson, D.R., 2006. Cryptography Theory and Practice, CRC 3rd, pp: 117-149.

Trappe, W. and L. Washington, 2006 Introduction to Cryptography with Code Theory, 2nd Edn., Prentice Hall.