A NOVEL WAY OF PROVIDING CONFIDENTIALITY TO SHARED SECRET KEY AND AUTHENTICATE THE SHARES DURING RECONSTRUCTION

P Devaki [1] and G Rghavendra Rao[2]

[1]Department of Information Engineering, The National Institute of Engg., Karnataka, India
p_devaki1@yahoo.com
[2] Department of Information Engineering, The National Institute of Engg., Karnataka, India
grrao57@gmail.com

ABSTRACT

If a secret key or a password is to be used in an application, where multi users are involved, then it is better to share the key or the password to all the authorized users in such a way that a minimum or all the shares of the users put together must result in the key or the password.

This method avoids the unauthorized access to the information by a single user with out the knowledge of the other users.

So this can protect the important information and also protect the key by sharing. If the key is not shared, then always there may be a threat of key being compromised intentionally or unintentionally. If the key is shared among the authorized users, at the time of requirement to use the information, few authorized users can use their shares to reconstruct the key and use it.

This avoids the drawbacks of the single user having the key. Also it is necessary to authenticate the shares given by the users during reconstruction of the key. This facilitates the verification of the modified shares or fake shares sent by the authorized or unauthorized users.

To achieve the confidentiality of the key we are using the threshold secret sharing method where the key will be divided in to number of shares depending on the number of authorized users and represent those shares in Braille format, and to provide authentication we are using SHA-1 algorithm. The key can be text data, image.

KEYWORDS

Authentication, Braille format, Bitmap image, Confidentiality, Multiple keys, Secret key, Secret sharing.

1. INTRODUCTION

There are two aspects in the area of information security. One is the actual information which is supposed to be very confidential. Example in any business organization the business related information stored in a database needs confidentiality. Only the authorized user can access that

Natarajan Meghanathan, et al. (Eds): SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 331–338, 2012.
 DOI : 10.5121/csit.2012.2332

information. Second one is the key or the password needed to enter in to the database must also be confidential. If the key or the password is not protected properly, definitely there will be a threat for the database where the sensitive information is stored. This is true in other areas like military, banking, medical, government organizations etc where sensitive information is being maintained. There will not be any problem if the information is allowed to be accessed by only one authorized user, and then the key is known only to that user. But if the information needs to be accessed by multiple users' utmost care must be taken so that both the key as well as the information stored in the system must be safe. But in a multi user system it is very difficult to achieve confidentiality, since maintaining the key itself is very difficult. A single user can not be given the key, because if that user doesn't turn up then there is no way that the information can be accessed, and also there can always be a threat of compromising the key to the unauthorized users, loosing the key, forgetting the key. On the other hand if the key is given to all the authorized users, then any of the users can access the system with out the knowledge of the other users. A method known as secret splitting (n,n) can be used, where the key will be shared among all the users. When the key is required, it is necessary to get all the shares from all the users. Even if one share is missed due to some reason it is not possible to reconstruct the key. So to avoid all these a technique devised by Shamir[1] known as threshold secret sharing (m,n) which is based on polynomial interpolation can be used so that the key can be shared to all the authorized users by the administrator. Where n is the number of authorized users and m is the minimum number of shares required to construct the key. When the key is required, the user who needs the key will request other users to send their shares. In this method m or more than m shares are required to construct the key. With less than m shares it is not possible to construct the key, this ensures the minimum number of users required to access the system. This overcomes the misuse of key, compromising the key, forgetting the key, etc. Also it indicates the users who participated in the construction of the key. At any point of time no user will have the complete key, and with the knowledge of the partial key he can not access the system.

In this paper we are using the threshold secret sharing method to divide the key, and represent those shares using Braille format which no one has done so far. Those braille images will be given to the users. So no individual user will come to know about the key. Even the attackers/hackers will not come to know about the key. If an attacker tries to get the share, since it is in Braille format he will not be able to decode the share.

In [13] the validity will be checked for the secret, whether it is in the range or not. But it doesn't determine the participant who has sent a fake or forged share.

We are using the SHA-1 which takes binary data of any length and produces fixed number of binary digits. This can be used to authenticate the shares during the reconstruction of the key. Since the key shares need to be transmitted over the network, and we know that anybody can tap the communication channel, the key share may be modified or a fake share may be introduced in to the network by an unauthorized user.

This can be identified by using SHA-1 algorithm.

2. THRESHOLD SECRET SHARING

First secret sharing (m,n) was proposed by shamir[1] , which is based on the polynomial interpolation. There are other secret methods proposed by many people but shamir's is very simple. This method can be used for the key of text, image, and integers also.

In this method a polynomial of order m-1 will be used.

The polynomial is used to share the secret by considering the coefficients.

$F(x) = S + C1x + C2x^2 \ldots\ldots\ldots + Cmx^m$

Where S is the secret to be shared, C1, C2 … Cm are coefficients. The coefficients can be any random integer values.

When it is necessary to use the key/secret to access the system, the user requiring the key can send a request for shares to other users.

When a user receives m or more number of shares he can reconstruct the required key.

The reconstruction of the key is based on Lagrange's interpolation as shown below.

$$F(x)= \frac{(x-x1)(x-x2)}{(x0-x1)(x0-x2)} y0 + \frac{(x-x0)(x-x2)}{(x1-x0)(x1-x2)} y1 + \frac{(x-x0)(x-x1)}{(x2-x0)(x2-x1)} y2 + \ldots\ldots\ldots\ldots$$

3. BRAILLE

Braille is the internationally recognized reading and writing system for the blind and partially sighted people[10]. Braille is not a language, it is another way to read and write a language. Characters are represented by an arrangement of raised dots. Each Braille character or cell is made up of six dot positions, arranged in a rectangle comprising of two columns of three dots. A dot may be raised at any of the six positions to form many combinations as shown in Fig -1.

Many of the papers have been written to securely transmit the shares to the users, by embedding the shares in images, audio etc. this is referred to as stegnography. This method requires the image or the audio signal must be large enough to hide the secret. And also it requires significant bandwidth to transmit the share embedded in an image. We are using barille to represent the shares for the following reasons. a) Since unless people know that this is a Braille format no one will be able to read the data. a) Even if anybody understands that this is a Braille format, they need to decide about the language since Braille can be used universally. c) Since it will be represented as a bitmap image the space and bandwidth required will be lesser compared to other BMP or JPEG images. d) No need of hiding the data.

Sample example has been shown to represent English alphabets; similarly it can be used to represent any other languages and numerical values also which looks like some game.

Unless one knows that this is a Braille representation of some language, no one will be able to understand that this image is having some confidential data.

So we have explored this property in our work to represent the shares in Braille.

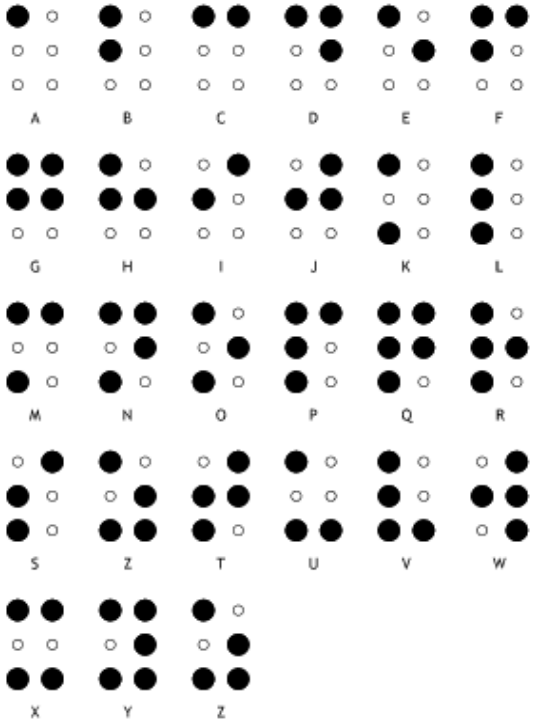

Fig – 1 Braille representation of alphabets

4. SHA (SECURE HASH ALGORITHM)

Encryption protects against passive attack where as message authentication protects against active attack. So it is necessary to protect the data against active attacks like modification of data, masquerade, replay etc.

SHA-1 provides message authentication. It provides a message digest of 160 bits for any length of input stream. For a given input of any length this algorithm generates a unique value called hash value or code. If any of the bits in the input is changed then the hash code will also be changed. But this is a one way function, by knowing the hash code it is not possible to generate the input data. this property can be used to authenticate the shares. During the reconstruction of the key one can check whether he/she has got the legitimate shares only or not by generating the hash code for the received share and compare the new and the old hash code, if they are different it indicates that that particular share has been unauthorizedly changed.

General structure of SHA-1 is shown in Fig-2.

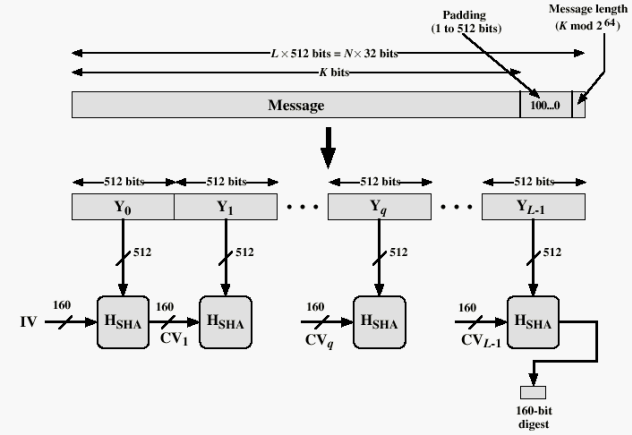

Fig – 2

Characteristics of SHA-1:

Every bit of the hash code is a function of every bit of the input. Due to the complex operations bits are well mixed and, rare chance of producing the same hash code for 2 messages with similar patterns. 2 messages having same code is on the order of 2256 operations. Difficulty of finding a message with a given hash value is on the order of 2512 operations. Using this, one can prove the integrity of the message.

5. PROPOSED WORK

As explained in section II, we are using the secret sharing scheme

We have considered (3,5) where 5 is the number of authorized users, and 3 is the threshold. The given key/secret is 222.

The polynomial is

$$F(x) = 222 + 3x + 2x^2$$

We can select 5 values for x and calculate F(x).

Y1= F(x1) = 227

Y2=F(x2)= 236

Y3 = F(x3) = 249

Y4 = f(x4) = 266

Y5 = F(x5) = 287

After obtaining the shares y1 to y5, the shares along with the values of x for each user can be given to the users as follows.

U1 = (x1, F(x1)) = (1, 227)

U2 = (x2, F(x2)) = (2, 236)

U3 = (x3, F(x3)) = (3, 249)

U4 = (x4, F(x4)) = (4,266)

U5 = (x5, F(x5)) = (5, 287)

When the 5 users get their respective shares, they can store them.

But before sending the shares they must be represented in Braille. So each pair of values will be represented in Braille as follows.

A B	CC[
C B	CIK
I B	CYJ
Y B	CKK
Q B	CS[

No doubt that Braille representation of the share gives confidentiality, but it doesn't prove authentication of the share. That means while sending the shares to users or while the user collects the shares from the other users, there may be a chance that an attacker intercepts and send a fake share or modify the share. Or even an authorized user him self can send a fake or modified share which results in wrong secret or key.

So to avoid this kind of attack which may be masquerade or modifying the share, SHA-1 is being used which generates a fixed length of code for any length of input stream.

So when the dealer generates the shares, he also generates the hash codes for all the shares.

The dealer inputs the shares to SHA-1 to generate a 160 bit code for each of the shares. While distributing the shares to the participants, dealer distributes one share and its hash value to a participant, and also the dealer sends the hash values of other shares to that participant. So each participant will receive his share and hash value along with the hash values of other shares belonging to different participants. Before storing the share and the hash values, each participant can check for the authenticity of its share by calculating the hash value for the received share, if the received hash value and the calculated hash values are same, then the participant can store that share and the hash value in its memory. When it is necessary to use the secret the participant can send a request for the shares. When the participant receives the shares from different participants, it will use the same SHA-1 algorithm to generate the hash values. These computed hash values will be compared with the stored values for each of the participants. If the values match then the shares are the correct ones which are sent by the authenticated participants only.

Even though this method increases the computation for some extent, it provides full authentication, like which participant is sending a fake or a modified share so that corrective measures can be taken on that communication channel or the participant.

Example :

Dealer generates 5 shares s1, s2, s3, s4, s5 and their corresponding hash values h1, h2, h3, h4, h5

He distributes the shares to 5 participants in the following way.

P1 s1, h1,h2,h3,h4,h5

P2 s2 , h1,h2,h3,h4,h5

P3 s3,h1,h2,h3,h4,h5

P4 s4,h1,h2,h3,h4,h5

P5 s5, h1, h2, h3, h4, h5

When the participants receive these they store their share and others hash value in the memory. It also verifies if the received share is the actual one or the modified one by calculating the hash value and compares that with the corresponding received hash value. If the values are same it will retain all the information else it can reject and inform the same to the dealer.

For example if the participant P1 received s1, h1, h2, h3, h4, h5

It calculates hash value for s1, and compares that with the corresponding received hash value h1. If they match then the P1 can store the secret and hash values in its memory else it can discard. Like this all the participants can verify their shares when they receive from the dealer. When a user wants to reconstruct the secret, he will request the other participants, when he receives the shares; he will generate the hash values and compares that with the stored hash values of the users. Thus the verification of the shares will be done by any user.

This method can be applied to any type of data text, numeric, or image.

6. SECURITY ANALYSIS

This method certainly gives security for the key in general. First of all the key is divided in to number of shares , so no single share reveals any information about the key. And also even if one share is not available still it is possible to reconstruct the key if m number of shares is available. Second since the shares are being represented in Braille definitely the attacker needs some time to understand and decode that format. Third since for each share the hash code is generated by the dealer and sent along with the respective shares to each of the users, the users can verify the shares before accepting the corresponding shares. Fourth when a user requests for shares to reconstruct the key he can also verify the shares he obtained from other users. Only if the received shares hash code match with the stored hash code of the users then only the receiver can construct the key. Otherwise he can reject those shares and inform the same to the dealer. This verifies the points of vulnerabilities.

7. CONCLUSION

This work has proved that specified number of threshold shares is sufficient to reconstruct the key; also we have used a new method to represent the share using Braille, to provide confidentiality. This method saves bandwidth and also the processing time compared to the stegnography.

Since in stegnography the image must be large enough and also there must be a method to hide the share in an image which requires sufficient processing time.

Many of the researchers deal with the new methods of providing confidentiality, but a very few have worked on authentication. Here we have used one of the strongest and simple algorithms to generate hash code for verification of the shares. Using this the users can verify the shares when they receive it from the dealer and also , users can verify the shares received from other users during reconstruction process. This clearly indicates which user is creating a fake share or which communication channel is vulnerable. So this work has proved both confidentiality and authentication of the key with out using encryption and decryption which requires processing time.

REFERENCES

[1] A. Shamir, "How to share a secret," Comn.ACM., **22**(1979), 612-613.

[2] S. Tang, "Simple Secret Sharing and Threshold RSA Signature Schemes", Journal of Information andComputational Scien e, 1, 2004, pp. 259–262.

[3] Cryptography and network security by William stallings , 3rd edition, pearson education

[4] G. Blakley, "Safeguarding Cryptographic Keys," AFIPS Conference Proceedings, **48**, 1979.

[5] C.S. Tsai, C.C. Chang, T.S. Chen, Sharing multiple secrets in digital images, J. Syst. Software 64 (2002) 163–170.

[6] Secure Distributed Key Generation for Discrete-Log Based Cryptosystems, R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, Eurocrypt 1999

[7] A. Beimel, and B. Chor, "Interaction in Key Distribution Schemes", Advances in Cryptology-CRYPTO'93, vol. 773, pp. 444-455, 1993.

[8] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable Secret Sharing and Achieving Simultaneity in the Presence of Faults", in Proc. 26th AnnualSymposium on Foundations of Computer Science, Portland, OR, 21-23 October 1985, pp. 383-395.

[9] C. Charnes, J. Pieprzyk, and R. Safavi-Naini , "Families of Threshold Schemes", in Proc. IEEE International Symposium on Information Theory. Trondheim, Norway, July 1994, p. 499.

[10] C.C. Thien, J.C. Lin, Secret image sharing , Comput. raphics 6 (5) (2002) 765–770.

[11] Blakley, G.R.: Safeguarding cryptographic keys. In: prooc. AFIPS 1979, National Computer Conference, vol. 48, pp. 313–137 (1979)

[12] Carpentieri, M.: A Perfect Threshold Secret Sharing Scheme to Identify Cheaters. Designs, Codes and cryptography 5(3), 183–187 (1995)

[13] Tompa, M., Woll, H.: How to Share a Secret with cheaters. Journal of Cryptology 1(3), 133–138 (1989)

[14] http://www.pharmabraille.co.uk/braille-alphabet.html.