Data Authentication in Distribution Automation System Using Optimized MD5 Hash Function

MORVARID SEHATKAR[1] FARAMARZ FAGHIHI[2]

1- Faculty of Engineering
University of Tehran
Amirabad, Tehran
IRAN

2- Electrical Engineering Department
Iran University of Science and Technology
Narmak, Tehran, 16844
IRAN

Abstract: - The quick expanding of distribution systems has caused that traditional maintenance, exploitation and protection methods wouldn't be possible anymore, therefore extracting the distribution systems' data, embodying them and using the automation system is an obvious and inevitable task .The Distribution automation system software must be capable of extracting data from systems, monitoring data to users properly and exerting the necessary control orders. In order not to make disruption in distribution system, receiving the data and sending the orders to circuit breaker, sectionalizing switches and ... Must be done absolutely proper which demonstrate the necessity of certificating the separated data from channel. In this paper in order to certificate controlling data and receiving the proper orders, it's suggested to use one-way function with complicated structure based on simple nonlinear functions' repetition. In these forms of functions the MD5 hash function has been chosen. In software implementation of this function, the proper selection of parameters such as Minimum memory capacity engagement and main body functions non coincidence has led to reach a suitable speed and security in data certification. Data certification in given way is higher in speed than the way discussed in [5] which was consist of digital signature based on elliptic curve, but in security wise the latter method is preferred. While the collision occurrence for the implemented hash function is not simply feasible.

Key-words: - Distribution Automation System, Hash Function, MD5, Cryptography, Authentication

1 Introduction

Distribution automation system makes it possible to have surveillance and control over distribution systems and also makes it possible for distribution companies to have instantly telesurveillance, coordination and order implementation on facilities [2].

Hereupon, in addition to distribution substations with digital control it is also possible to automate the medium volt substations in distribution section, so that we can simply reach the data and regional situations of a zone. But because a set of orders is sent to the substation by the surveillance point it's possible that the orders be sent or received conversely and resulted in untimely power outage or joint which would resulted in twice worse harms. Therefore the foregoing data certification is remarkably important.

In [5] a signature proposal based on elliptic curve had been proposed which was highly secure but that algorithm's slowness, made a small delay in data certification [1, 9].

In this paper it's suggested to use MD5 hash function [3, 10], but considering this function parameters it's been tried to reach higher speed and security in implementation, which is obvious in implementation results.

The following is how the subjects are adjusted: In chapter 2, the message authentication in distribution automation system has been discussed. In chapter 3, the hash functions' principles and features for authenticating would be discussed. Chapter 4 is about the MD5 hash function implementation. Chapter 5 is discussing the MD5 hash function security. Chapter 6 compares the digital signature based on elliptic curve with MD5 hash function and finally chapter 7 covers Conclusions.

2 Message Authentication in Distribution Automation System

From the IEEE vision, Distribution automation system is a system which in a real time enables a distribution company to have telesurveillance,

coordination and controlling of distribution equipments from a distant place. Hereupon, the communication channels must be used.

Using the automation system has a lot of features for distribution companies such as: dropping in extortionate protection and maintenance expenses, having perpetual reach on a large amount of data and making charts and statistical reports. But it's clear that in all cases the accuracy of received data is highly important, perhaps giving false data affects instant operations and future plans. Instant operation can be false cutting or joining the circuit breaker or sectionalizing switches and future plan can be considered the load increase and yearly consumption programming that each one can result in rampant damages.

3. Hash Function

For message authenticate, it's possible to use a function called "Hash Function" [4]. Hash function is a function in the form of: " H=h (M) " which has 3 features:

1. Calculating the function value from the given message is easy.
2. Understanding the given message from function value is difficult.
3. It's difficult to reach another message (M') by having the message (M) under the H (M) = H (M') condition.

In prior definition, the terms of "easy" and " difficult" are from the computational complexity aspect.

Usually the hash function output value is much less than given message value that makes it easy to be sent through Communication channels.

In order to make a hash function, first a logical mathematical function must be considered, then by using a given value, the $h_i = f(m_i , h_{i-1})$ recursive algorithm would be done until the final value is resulted.

The functions that are used in body of hash functions can be classified in two common groups:

- Based on block cipher cryptography.
- Consisting ofnonlinear simple function repetition.

In this paper the MD5 hash function resulted from some simple function repetition, is implemented.

4 MD5 Hash Function Implementation

MD5 hash function produces a 128-bit hash value from 512-bit given message. This function is consisted of a four step main loop. The following is Hash function algorithm:

1. Input text is processed in 512-bit blocks, that each block is divided to sixteen 32-bit sub blocks. The first step is to pad the message to a multiple of 512 bits. This is done by following the message with between 1 and 512 padding bits, the first of which is 1, the rest of which are 0s, and then following that with a 64-bit integer that is the orthogonal message length in bits. And this allows messages of arbitrary length up to 2^{64} bits.
2. The output of the algorithm is set of four 32-bit blocks, which concatenate to form a single 128-bit hash value.
3. Four 32-bit variables are initialized:

A = 0x01234567
B = 0x 89abcdef
C = 0x fedcba98
D = 0x76543210

4. For each of 512-bit blocks of message, main loop of the algorithm would be repeated.
5. The value of A, B, C and D variables are copied in a, b, c and d variables.
6. Each operation performs a nonlinear function on three of four 32 bit variables.

$F(x, y, z) = (x \wedge y) \vee (\sim x \wedge z)$
$G(x, y, z) = (x \wedge y) \vee (y \wedge z)$
$H(x, y, z) = x \oplus y \oplus z$
$I(x, y, z) = y \oplus (x \vee \sim z)$

7. The result is added to a sub block and a constant value.
8. A ‘s’ bits left rotation would be done (‘s’ is a variable) and finally the result is added to the fourth variable.
9. After all this work, the original values of variables have been thoroughly mangled in a way that, while completely dependent on the message bytes, provides no algorithmic way to find out what those massage bytes were. The mangled digest is now added to the digest value that existed prior to the current stage, and that become the new digest value. The algorithm now proceeds to digest the next 16 bytes of the message until there is no more to be digested; the output of the last transformation is the final messages digest.

If for example we consider the F (b, c, d) as a function, ‘a’ will be added to F (b, c, d). Then the result would be added to M_j and t_j, (M_j is the jth message sub block and t_i is a constant value that in the ith step it's equal to [2^32 abs (sin i)] in which i is in Radian unit), then the result would be rotated to left and be added to ‘b’ and be stored in ‘a’.

$FF(a,b,c,d,M_j,S,t_i)$ denotes $a = b + ((a+ F(b,c,d) + M_j+ t_i)^{<<<s})$.

GG(a,b,c,d,M_j,S,t_i) denotes a = b + ((a + G(b,c,d) + M_j+ t_i)$^{<<<s}$).

HH(a,b,c,d,M_j,S,t_i) denotes a = b + ((a + H (b,c,d) + M_j+ t_i)$^{<<<s}$).

I I(a,b,c,d,M_j,S,t_i) denotes a = b + ((a + I (b,c,d) + M_j+ t_i)$^{<<<s}$).

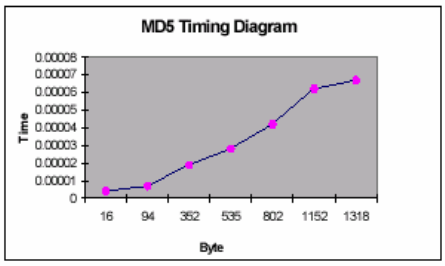

Fig.1. MD5 Timing Diagram with main functions

In the implementation time, proper using of 'F', 'G', 'H' and 'I' functions is highly important and the algorithm's speed and security can be enormously affected by that. Different types of these functions were tested that for each one a different time was resulted although they were totally closed to each other. One type of these functions with low performance and our optimized functions those are suitable for capacity of transmitted data in distribution automation systems are according to below:

Low performance function (**G2**):

$$F(x, y, z) = (x \wedge \sim y) \vee (\sim x \wedge z) \vee (y \wedge z)$$
$$G(x, y, z) = (x \wedge \sim y) \vee (y \wedge \sim z)$$
$$H(x, y, z) = (\sim x \oplus y \oplus z) \wedge z$$
$$I(x, y, z) = y \oplus (\sim x \vee \sim z)$$

Optimized functions (**G3**) (for data with 0-1500 bytes capacity)

$$F(x, y, z) = (x \wedge y) \vee (\sim x \wedge z)$$
$$G(x, y, z) = x \oplus (y \wedge z)$$
$$H(x, y, z) = (x \oplus y \oplus z) \wedge x$$
$$I(x, y, z) = \sim y \oplus (x \vee \sim z)$$

G1 is introduced the main functions of MD5.

In table 1 and figure 2 suggested functions are compared.

Table1 Execution Time of functions

Byte	Time(G1)	Time(G2)	Time(G3)
16	0.000004	0.000005	0.000003
94	0.000007	0.000007	0.000006
352	0.000019	0.000021	0.000016
535	0.000028	0.000031	0.000025
802	0.000042	0.000043	0.000035
1152	0.000062	0.000062	0.000052
1318	0.000067	0.000068	0.000056

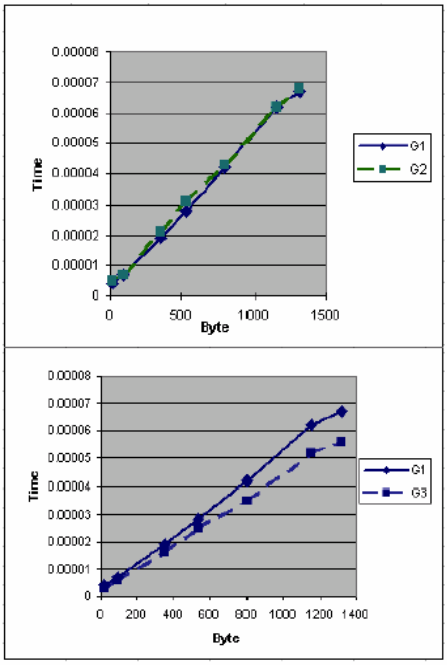

Fig.3. Comparing Submitted Functions with Main Functions

From this chart we can result that:

1-Totally, MD5 is a fast hash function, which in comparison with another hash functions is much faster.

2- The algorithm's execution time will grow proportionally by increasing of message length, which shows that for the messages with very long length the MD5 wouldn't be effective anymore. Nevertheless this problem has no matter in the distribution automation systems because of the kind of sending orders.

3- The optimized time is resulted by using functions that have following conditions:

- The total structure of four functions must be varied.
- There must not be any symmetry in every function.
- There must be a proper padding algorithm.

Naturally the prior parameters not only provide a proper speed for MD5 function but also provide the necessary security feature for it, which is discussed hereinafter.

5 MD5 Hash function security

In the family of hash functions, which are based on nonlinear simple functions repetition, the N-Hash and MD4 functions were not resistant against collision attack. In 1995, Dobbertin in an article showed that MD4 hash function was not resistant against collision attack [7, 8, 11]. Kasselman did a faster attack on this

function in 1997 [6]. The presented attack by Dobbertin is divided in two parts:
- Internal collision attack
- Differential attack and right initial values selection

The differential attack will succeed when some internal collision happen. So to attack a MD4, first step is to find an internal collision. About this case, the proper selection of initial values is a very important step in decreasing the number of logical existent equations for finding collision.

But about the MD5 security, there are the following aspects:
- One step is added to MD4 steps, so the complexity is more.
- Due to using the nonsymmetrical functions in main algorithm and more optimal functions that we mentioned in this article, the collision happening would be more difficult.

So we can say that the MD5 hash function has proper security features and can be used in all usual authentication needs.

6 The comparison between MD5 Hash function and Digital Signature based on Elliptic Curve

In [5] a new proposal about digital signature based on elliptic curve was proposed in order to authenticate the send and received signals in distribution automation system, because the elliptic curve is in NP-Complete problem class, surely it has an admissible security but inherently has a slow algorithm so the distribution automation system speed is not as high as it's expected [12, 13]. But by this article suggestion about using MD5 hash function the speed problem will be solved.

Also MD5 hash function is so much resistant against collision and statistically reaching to collision is not possible in a short period of time, so using the implemented MD5 hash function to authenticate received signals is much more effective than the elliptic curve method.

7 Conclusion

In this article after discussing about the necessity of authentication in distribution automation system, using optimized MD5 hash function was suggested. After some changes, this function was implemented and performed by C++ language. Reaching to the proper times and concerning the proper complexities, the problem about the slowness of suggested algorithm in [5] has been solved and the data authentication in automation system with proper speed and secure algorithm has become possible.

Refrences

[1] R. Rauscher and F. Bohnsack, Result of an Elliptic Curve Approach for Use in Cryptosystem, *IEEE Transaction on Information Theory*, 1999, pp. 415-422.

[2]A. Pahwa, Flexible Control of Distribution System, *Lecture Notes*, Kansas state university, 1999.

[3]A.J. Menezes, P. Cvan, O. Scott and A.Vanstone, *Handbook of applied Cryptography*, CRC press, 1996.

[4]D. Stinson, *Cryptography, Theory and Practice*, CRC press,2002.

[5]F. Faghihi, M. Esmaeili and M. Sehatkar, Information Security in Distribution System by Elliptic Curve, *17th International Power System Conference* ,Tehran – Iran , Proceedings(4), Control , Protection, Telecommunication & IT, 28-30 Oct. 2002, pp.89-97.

[6]. P.R. Kasselman, A Fast Attack on the MD4 Hash Function, *Information Theory, IEEE*, 1997.

[7]. H. Dobbertin, Cryptanalysis of MD4, In *Proceedings of the 3rd Workshop on Fast Software Encryption*, Cambridge, U.K., pages 53-70, Lecture Notes in Computer Science 1039, Springer-Verlag, 1996.

[8] E. Abdel-azeem, R. Seiregisamir, and I. Shaheen, Cryptographic Security Evaluation of MD4 Hash Function, *Proceeding of the 13th national radio science conference*, March 19-21, 1996, cario, Egypt.

[9]M. Y. Hunag, Investigation of the Elliptic Curve Cryptosystem for Multi-application smart card, 1998 second international conference on knowledge-Based Intelligent Electronic System, 21-23 April 1998, Adelaide, Australia.

[10] R. L. Rivest, The MD4 message digest algorithm, Advances in Cryptology, CRYPTO 90, Vol. LNCS 537, 1991, pp. 303-311.

[11] B. Denbore and A. Bosselaers, An attack on the last two rounds of MD4, Advances in Cryptology, CRYPTO 91, Vol. LNCS 576, 1992, pp. 194-203.

[12] T. Kobayashi, H. Morita, and F. Hoshine, Fast Elliptic Curve Algorithms Combining Frobenius Map and Table Reference to Adapt to Higher Characteristic, EUROCRYPT 99, Vol. LNCS 1592, 1999, pp. 176-189.

[13] Y. Han, P. Leong, and T. J. Zhang, Fast Algorithms for Elliptic Curve Cryptosystem over Binary Finite Field, Crypto 2000, pp. 75-85.