# Message Authentication Using New Hash Function

Hasanen S. Abdulah[*], Maha A. Hamood Al-Rawi[*] and Dalal N. Hammod[**1]
[*] Department of Computer Sciences, University of Technology, Baghdad-Iraq.
[**] Department of Computer Sciences, College of Sciences, University of Al-Nahrain, Baghdad-Iraq.
[1]E-mail: dal_scin81@yahoo.com.

**Abstract**

In cryptography, hash functions have very important effects when it's used in the message integrity, digital time stamping, and digital signature. Hash functions compute hash value by a set of logical (primitive) operations that perform on a (32) bit words for authentication. Authentication process becomes very important especially manipulated message undetected that can have disastrous effects in Network Management and E-Commerce. In this paper, a proposal for new hash function based on MD5 is developed. The length of message digest is 224-bit through (4) rounds that each round has a pair of (16) steps. The final number of steps is (128) that make stronger function against collision attests and more secure than MD5.

Keywords: Hash Function, Message Digest, MD5, Authentication.

## 1. Introduction

Cryptography covers a wide area since 4000 years ago the Egypt up to world wars and contemporary internet applications. Cryptography has been used especially in military services and diplomacy. In this area cryptography has been used as a tool for protecting national strategies and information [1].

Cryptographic functions consist of three kinds : functions with secret key, functions with public key, and hash function, which have the ability to compress message from any very length to a fixed length **(finger-print or message-digest)** [2].

Authentication is not encryption that used for many Requirements: (a)Masquerade– Insertion of message from fraudulent source, (b) Content Modification – Changing content of message, (c) Sequence Modification– Insertion, deletion and reordering sequence, and (d) Timing Modification – Replaying valid sessions [1].

## 2. Hash Functions

Hash functions take an input arbitrary-message and produced an output fixed-message [7]. This process is irreversible. There are many applications of hash functions such as: digital signature, message integrality, and message originality etc. Many algorithms including in hash function such: Message Digest Algorithm (MD×), Secure Hash Algorithms (SHA×), RIPE-MD, N-Hash, and HAVAL etc. Recently, MD5 is the most used because it is most secure algorithms. In spite of its security, collision could be found in MD5 through two hours by using "Mod Difference" produced by Wang [2]. MD5 used four nonlinear functions explained below, used a different one (used in each operation) for each round [9].

$$F(X,Y,Z)=(X\wedge Y)\vee(\neg X\wedge Z) \qquad 0<=iI<=15 \quad \text{.....(1)}$$
$$G(X,Y,Z)=(X\wedge Z)\vee(Y\wedge\neg Z)) \qquad 16<=iI<=31 \quad \text{.....(2)}$$
$$H(X,Y,Z)=X\oplus Y\oplus Z \qquad 32<=iI<=47 \quad \text{.....(3)}$$
$$I(X,Y,Z)=Y\oplus(X\vee\neg Z) \qquad 48<=iI<=63 \quad \text{.....(4)}$$

(Where, $\oplus$ is XOR, $\wedge$ is AND, $\vee$ is OR, $\neg$ is NOT, and iI is the step number).

Hash functions have properties and cryptographic requirements, which explain in the following section.

### 2.1 Hash Function Properties [7]

There are three properties for an un-keyed hash function (H):

1- *Pre-image Resistance:* A Hash Function (H) represented a pre-image resistant, if it is impossible to obtain original message from hash value H(M).
2- *Second Pre-image Resistance:* A Hash Function (H) represented a second Pre-image resistant if a given message (M), it is impossible to obtain another message (M') such that H(M)=H(M').

3- *Collision Resistance:* A Hash Function (H) represented a Collision Resistant, if it is impossible to obtain any two messages (M) and (M') such that H(M)=H(M') while M≠M'. For a secure hash function, the best attack to find a collision should not be better than the *birthday attack* (i.e. not better than work complexity of 2n/2 for a hash function outputting n-bit hash values).

### 2.2 The Basics of Hash Function for Cryptography [8]

The basics of hash function for cryptography are:

1- Message in various length as an input and fixed length as an output.
2- H(X) is easily to compute when a given x.
3- H(X) is a one-way function.
4- H(X) is a collision-free function.

According to these basics, we have applying a new224-bit hash algorithm based on MD5.

## 3. Related Work

There are many achievements occurred in the field of cryptographic by message digest, each suggests new method for developed MD. The most useful ones are mentioned in the following:

### 3.1 Muna Mohammed Al-Nayar, "A proposed Secure Protocol for E-Mail System Based on Authentication and Hash Function", 2011[3].

The researcher proposed a protocol based on multiple random keys to avoid key distribution or breaking problem and to ensure data authentication and integrity. The proposed protocol generated through communication session automatically and achieves more secure communication due to using multiple functions such as hash function, randomize function, splitting and merging which make it more complex unpredictable and easy to implement.

### 3.2 Alok kumar kasgar, Jitendra Agrawal, and Santosh Sahu, "New Modified 256-bit MD5 Algorithm with SHA Compression Function", March 2012[4].

The researchers proposed a modified MD algorithm that can be used for integrity message and to have a bigger size of hash (512, 768, …) like SHAs by extending the block size of compression functions or increasing number of them. In this paper combination of some function to reinforce these functions and also increasing hash code length up to 256 that makes stronger algorithm against collision attests.

### 3.3 Nidhi Garg, and Neeta Wadhwa, "Design of New Hash Algorithm with Integration of Key Based on the Review of Standard Hash Algorithms", August 2014 [5].

The researchers proposed algorithm produces a hash code of 192 bits from an arbitrary length input and serves the requirement of both the message integrity as well as source authentication. This algorithm consists of very simple steps, therefore would have lesser overhead and complexity as compared to the standard hash algorithms. The most important feature of this algorithm is that it uses a key as an ingredient to function for calculating the digest, thus providing source authentication as well as message integrity.

### 3.4 Hanumantu Rajeswari, Ramesh Yegireddi, and Vudumula GovindaRao, "Performance Analysis of Hash Algorithms and File Integrity", 2014[6].

The researchers give both an overview of the performance analysis of hash functions in cryptography and a presentation of file integrity in mobile phones and identified that the SHA-1 has best performance on 32-bit processor; the same algorithm is used in file integrity that a novel and efficient way of storing files in Android Mobile devices such that if any file is taken as evidence in the court of law.

## 4. The Proposed Hash Function

In this paper, a proposal for a new hash function has developed based on the MD5 method that represented a simple hash function and required approximately less complexity running time and kept on the same complexity class such as in the MD5. The proposed hash function used useful properties of the MD5 as follow:

1- In MD5 analysis, if the length of hash value is between (128-160) bits, it will be a good hash value but it may be broken by brute force attack and if approximately the length up to the 256 bit, it will be better hash value. In this paper we used the 224-bit (28-Byte) for hash value.
2- MD5 used to compute the message digest four-word buffer (A, B, C, D). Each of A, B, C, and D is a 32-bit buffer. These buffers are initialized to the following values:
A = 0x76543210, B = 0xfedcba98,
C = 0x89abcdef, D = 0x01234567
But the proposed method will use seven-word buffer (A, B, C, D, E, F, G) to compute the message digest. Each of A, B, C, D, E, F, and G is a 32-bit buffer. These buffers are initialized to the following values:
A=0x01234567,
B=0x89ABCDEF,
C=0xFEDCBA98,
D=0x76543210,
E=0x765432AB,
F=0x102233CD,
G=0x776655EF
3. MD5 used four rounds, each round has 64-steps (operations), and using four nonlinear functions, these nonlinear functions were explained in equations 1, 2, 3, and 4. But the proposed method used four round, each round has dual 16-stepsthis means each round has 32-steps.The final number of steps in the four rounds is 128-steps. Dual steps consist of two operations: first operation is perform on the first four variables (A, B, C, and D), and second operation is perform on the last four variables (G, F, E, and D). There are four nonlinear functions used in the four rounds which are described below:

$$\left.\begin{aligned} F(X,Y,Z) &= (X \wedge Y) \vee (\neg X \wedge Z) \\ I(F1,E,D1) &= E \oplus (F1 \vee \neg D1) \end{aligned}\right\} \quad 0 <= iI <= 15 \quad \text{----------(5)}$$

$$\left.\begin{aligned} G(X,Y,Z) &= (X \wedge Z) \vee (Y \wedge \neg Z)) \\ H(F1,E,D1) &= F1 \oplus E \oplus D1 \end{aligned}\right\} \quad 16 <= iI <= 31 \quad \text{----------(6)}$$

$$\left.\begin{aligned} H(X,Y,Z) &= X \oplus Y \oplus Z \\ G(F1,E,D1) &= (F1 \wedge D1) \vee (E \wedge \neg D1)) \end{aligned}\right\} \quad 32 <= iI <= 47 \quad \text{----------(7)}$$

$$\left.\begin{aligned} I(X,Y,Z) &= Y \oplus (X \vee \neg Z) \\ F(F1,E,D1) &= (F1 \wedge E) \vee (\neg F1 \wedge D1) \end{aligned}\right\} \quad 48 <= iI <= 63 \quad \text{----------(8)}$$

**Note: D1 is a copy from D word.**

4. MD5 computes a new value (temp) by using the following equation [9]:

$$a = b + ((a + F(b,c,d) + Mj + ti) <<< s) \text{----------(9)}$$

But the proposed method computes two values (temp, temp1) by using the following equations:

$$a = b + ((a + F(b,c,d) + Mj + ti) <<< s) \text{----------(10)}$$
$$g = f1 + ((g + F(f1,e,d1) + Mj + ti) <<< s) \text{----------(11)}$$

**Note: d1 is a copy from d word.**

5. Block size in the MD5 is 512-bits but in the proposed method is 896-bits.
6. In the MD5, padding processing is addition 1-bit followed by 0's and the last 64-bit used for representing message length but in the proposed method is addition 1-bit followed by 0's only to the end of block.
7. The result of MD5 is 16-bytes but in the proposed method is the 28-bytes.

The structure of the proposed method is explained in the figures below. Figure (1) represents the left half round, where A,B,C, and D represent variables, F represents nonlinear function (known as selection function), T represents a constant value with size of 32-bit which is computed by using sine function (the proposed method used128, 32-bit constant), Wi represents which word is used in this round (the proposed method generated a 128 word that each word is computed by XOR operation between some previous words), and <<<S represents the shift amounts in each round which have been optimized. The shifts in different rounds are distinct.

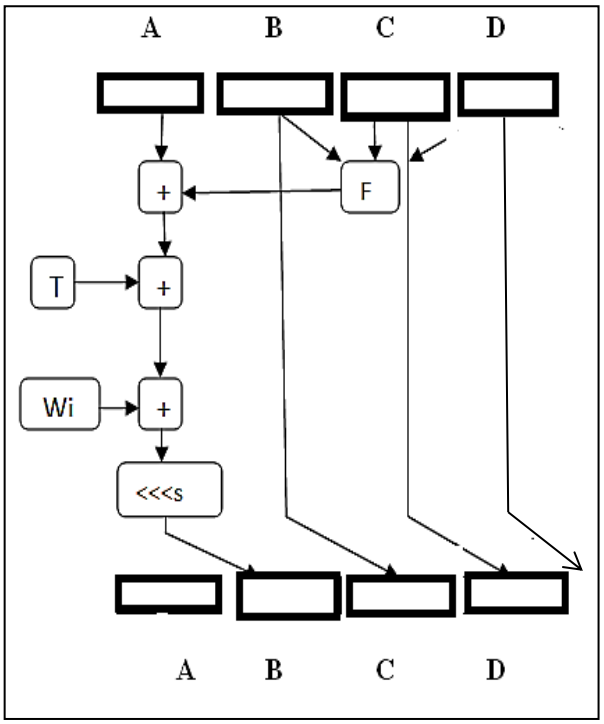


***Fig.(1): The Left Half Round.***

Fig.(2) represents the right half round, where D, E, F, and **G** represent variables, and F, T, Wi and <<<S are described in the previous section.

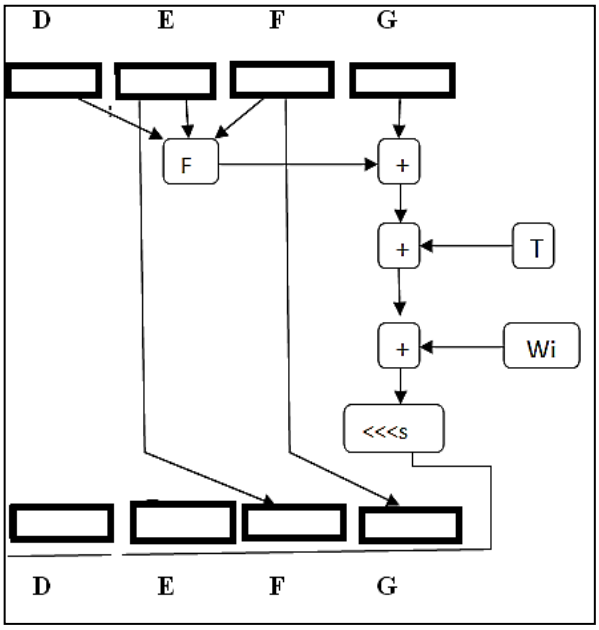


***Fig.(2): The Right Half Round***

Fig.(3) represents the complete structure of the proposed method for one round with two operations.

**Note**; for the next round, the final value of D variable computes by apply XOR operation between the D variable in the right half and D variable (D1 copy from original D variable) in the left half.

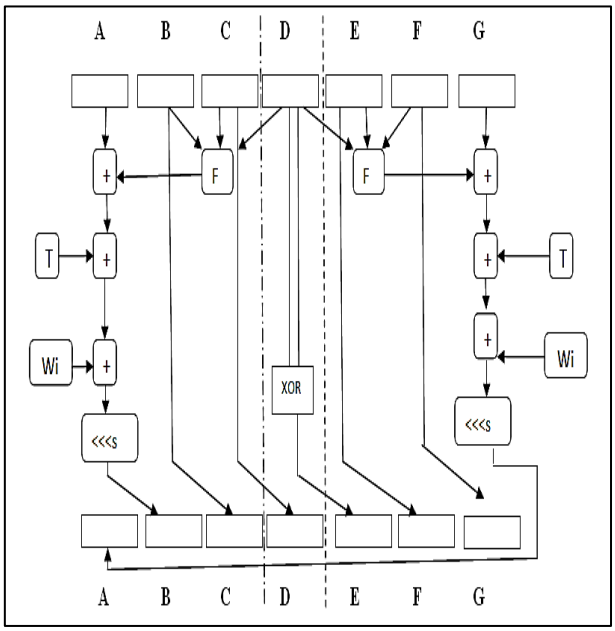


***Fig.(3): The Completely Structure of the Proposed Method.***

## 5. Discussion and Experimental Results

In testing the proposed method for a vary messages and calculated consuming time for hash value, gotten the following results and are compared with the standard MD5 hash function, as listed in Table (1).

***Table (1)***
***Compared time consuming and hash value between MD5 and Proposed Methods.***

| *Message* | *Length in byte* | *MD5* | | *Proposed Method* | |
|---|---|---|---|---|---|
| | | *Hash value in hexadecimal* | *Time in nanosec* | *Hash value in hexadecimal* | *Time in nanosec* |
| “” | 0 | D41D8CD98F00B204E9800998ECF8427E | 1248459 | EF556677CD332210AB3254761032547698BADCFEEFCDAB8967452301 | 1925801 |
| “abcc” | 4 | 26CA5BFE74F8DE88CCAAC5C0F44B349D | 1438191 | CC62FF5762FFB9573F3D841A85056F8A3BDCEA8DCCB8BDB51F6BEB55 | 2208613 |
| “ccba” | 4 | B5BA5758A3B2D6EA3D76E8A438FBBB6CB179 | 1846243 | 2DE741E0543FA96FE7F514DED87D6202E42CC4421705BD555519BFEE | 4482926 |
| “accc” | 4 | DB2E6E69B5DA8158F1875496F9B3B915 | 1602439 | 358588C5E2C49FD16B55238F8DE62D1376ADB9E7626649BD87095D4C9 | 2060278 |
| "message digest" | 14 | F96B697D7CB7938D525A2F31AAF161D0 | 1375059 | 2F1A464B9AFA77C474AC5819DE9B46280E6C2C5BED9CBA6D56146F40 | 2051039 |
| “ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789" | 62 | D174AB98D277D9F5A5611C2C9F419D9F | 1399183 | 92C28C0A0934A867F4228B87E071DC708CAAA3892BA0C8D741C6AF82 | 1905269 |
| “12345678901234567890123456789012345678901234567890123456789012345678901234567890" | 80 | 57EDF4A22BE3C955AC49DA2E2107B67A | 1343749 | 455F6EBBC95A049757845A725A91CF46169BB726AB79240CF1F73E68 | 23693194 |

The time consuming for the proposed method is more than MD5 because hash value of proposed method is longer than MD5 and when a calculation runtime complexity for XOR operation is achieved, the number of used this operation in the proposed method is (160) times whenever the number of used the same operation in the MD5 is (48) times, but the complexity class O(n) kept as the same as MD5.

Also, in the second message and third message in Table (1), the change only one character, but a different result completely. In Hash analysis, when one character changed then 50% from hash value must be changed.

Also, note treated seven variables of size 32-bit that separated into two parts each part consist of four variables of size 32-bit by D variable copies for used in the two half (right and left). The final result to D variable obtained by performs XOR between D variable in the left half and D variable in the right half. Then treated seven variables as circular rotated to right direction before starting next round.

## 6. Conclusions

In this paper, a proposal of new hash function is developed based on MD5 that has palpable advantages in security because it double the number of steps, but required time complexity more than MD5. Besides, combining two structures of MD5 and perform in the same steps (64-operations) but duplicate calculated operations in each step. The proposed method is a stronger function against collision attests because its obtained hash value in 224-bit (28- byte) by using four round and each round has dual 16-steps (operations) this means each round has 32-steps (operations). The final number of steps (operations) in the four rounds is 128-steps (128 operations). After testing this method it was obtained that a simple method and more secure than MD5 method.

**الخلاصة**

تمتلك دوال المزج المستخدمة في التشفير, تاثيرات مهمة اذ تستخدم هذه الدوال في تكامل الرسائل, بصمة الوقت الرقمية و التوقيع الرقمي. تقوم دوال المزج باحتساب قيم المزج المعتمدة على مجموعة بسيطة من العمليات المنطقية التي تطبق على 32 بت لاغراض التخويل. وقد اصبحت عملية التخويل مهمة جدا في الوقت الحاضر, خصوصاً في حالة عدم اكتشاف التلاعب بالرسائل اذ يكون له تاثير سيء في التجارة الالكترونية وادراة الشبكات. تم في هذا البحث, اقتراح طريقة مزج جديدة ومطورة تعتمد على طريقة مزج الرسالة (نسخة الاصدار الخامسة) MD5 اذ ان طول الرسالة الممزوجة يكون ٢٢٤ بت خلال ٤ دورات بحيث تحتوي كل دورة على زوج من عدد من الخطوات والبالغة ١٦ خطوة. بذلك, يكون المجموع النهائي للخطوات هو ١٢٨ خطوة هذا ما يجعل الدالة اقوى ضد احتمالية وجود رسالتين لهما قيمة مزج واحدة واكثر امناً من طريقة MD5.

## References

[1] Sarisakal M. N., Acar D., and Sevgen S., "A Secure Session Management System Using MD5 Algorithm", istanbul University, Faculty of Engineering, Department of Computer Engineering, 34850, Avcilar, Istanbul, Turkey, 1998.

[2] Chan X. and Liu G., "Discussion of One Improved Hash Algorithm Based on MD5 and SHA1", Proceedings of the World Congress on Engineering and Computer Science 2007 (WCECS 2007), 2007.

[3] Al-Nayar M. M., "A proposed Secure Protocol for E-Mail System Based on Authentication and Hash Function", Eng. & Tech. Journal, 29(16), 3291-3301, 2011.

[4] kasgar A. k., Agrawal J., and Sahu S., "New Modified 256-bit MD5 Algorithm with SHA Compression Function", International Journal of Computer Applications (0975 – 8887), 42(12), 47-51, March 2012.

[5] Garg N., and Wadhwa N., "Design of New Hash Algorithm with Integration of Key Based on the Review of Standard Hash Algorithms", International Journal of Computer Applications (0975–8887), 100(8), 11-18, August 2014.

[6] Rajeswari H., Yegireddi R., and RaoV. G., "Performance Analysis of Hash Algorithms and File Integrity", Hanumantu Rajeswarietal,/ (IJCSIT) International Journal of Computer Science and Information Technologies, 5(6), 7376-7379, 2014.

[7] Hassan N.F., Ali A. E., and Aldeen T.W., "Generate Random Image-Key using Hash Technique", Eng. & Tech. Journal, 28(2), 382-397, 2010.

[8] Stallings W., "Cryptography and Network Security Principles and Practice", Prentice Hall, 2011.

[9] Jerry Li, "MD5 Message Digest Algorithm", San Jose State University, 2003.