CENX586: Network Security

Dr Abdulrahman Aish Almutairi

November 2023

# A Critique on –

"A Novel Improvement with an Effective Expansion to Enhance the MD5 Hash Function for Verification of a Secure E-Document" by Dr Ammar Mohammed Ali and Dr Alaa Kadhim Farhan"

Submitted by:

Mohammed Shahzad

444105788@student.ksu.edu.sa

The research paper titled "A Novel Improvement with an Effective Expansion to Enhance the MD5 Hash Function for Verification of a Secure E-Document" by Dr Ammar Mohammed Ali and Dr Alaa Kadhim Farhan, published on 20 April of 2020 in IEEE access vol. 8, is an interesting approach to improve the security and scalability of the MD5 hash function widely used today. The authors introduce the research in section 1 and related research in section 2. MD5 main structure and weaknesses in current scenario in Section 3 a and b. They propose improvements in section 4 and 5. and then measure the performance in section 6. Discuss the results in section 7 and 8. Finally they conclude the research findings in section 9.

Their topic of choice is relevant, MD5 indeed has room for improvements in security and efficiency. Their goal is to improve performance and security of MD5 by introducing novelties like variable length hash digest. As we shall see, their approach might put off the reader before reaching the end of paper due to irregular flow of ideas, unconventional sentences usage and common grammatical errors. This may hinder the propagation of researchers' ideas to some readers.

Section 1 – Introduction, problem statement and proposal

The message-digest algorithm (MD5) as one-way cryptographic hash functions generally provides a 128-bit hash value. MD5 was designed by Ronald Rivest in 1991 to replace a more immediate hash function MD4. Most users are familiar with validating electronic documents based on a Hash function, such as the MD5 algorithm and other hash functions, to demonstrate the data integrity.

The authors of the research claim that, there are many weaknesses of the current MD5 algorithm, mainly its failures and weaknesses against varying types of attacks, such as brute force attacks, rainbow table attacks, and Christmas attacks. Dictionary Attack uses a dictionary data list. The rainbow attack is a table that is typically utilized to break down and cracking cryptographic hash functions, by reverse password hashes.

In this paper, a novel and efficient approach is presented for more collision-resistant MD5. The researchers recommend that more powerful security and elasticity can be achieved for the current MD5 128-bit algorithm by modifying length of the message digest. The researchers also propose the use of a key to eliminate threats that commonly appear in rainbow attacks and dictionary attacks. The authors discuss five methods to enhance the security and feasibility of the standard MD5 hash function.

Section 2 - Related research

The related research studied include overcoming the limitations of the MD5 function by relying on parallel coding, as the researchers combined this coding between MD5 and Blowfish function in order to increase and provide sufficient strength and security. Some present an implemented hybrid cryptography that uses both MD5 and the proprieties of an elliptic curve cryptosystem (ECC) to generate key steam. Other strategies to enhancement MD5 include, applying the concept of steganography combined with cryptography for increasing security.

Section 3 – MD5 structure and weaknesses

The message-digest algorithm (MD5) as one-way cryptographic hash functions generally provides a 128-bit hash value. MD5 was designed by Ronald Rivest in 1991 to replace a more immediate hash function MD4.

A. Main Structure of a Standard MD5 Hash Function

The conversion of text to another encrypted text by using the MD5 function, which can be summarized in five basic stages as shown in the following:

1. Appending the padding bits as preprocess. This step helped to make the length of the message corresponding to 448, modulo 512 by adding a number of bits between (1. . . 512), so that the resulting message length is a multiple of 512. (64 bits short of any multiple of 512) This will extend the message so that it is only 64 bits. The "1" bit is added to the message and then a set of 0 is appended so that the final block length matches 448.
2. Append the length of message. In this step, the extension to the original message length was added, this extension represents the 64-bit length of the original message at the end of the bits in the previous step. If the message length is a k bit, we add the value k mod 264.
3. Initialize MD buffer. Recorders are configured as a four-word buffer (A, B, C, and D) to calculate the message digest. Each of these recorders is configured to be the 32-bit length in a hexadecimal byte and low order by using the following initial values: Initialize variables: (A = 67452301, B = efcdab89, C = 98badcfe, D = 10325476).
4. Process the message This step represents the most important step in the algorithm. In this step, 16-word message SUB blocks are processed, which is equal to 512 bits. It consists generally of four cycles; each cycle has its own function: F, G, H, I, and in each cycle, there are 16 steps. In each round, 16 steps are performed on the recorders, where the output of words of this step is entered into four rounds in each round is scattered for these words.
5. Output. After processing of all blocks, plain text is converted into ciphertext or hash form as a message digest, where the final value MD5 hash is 128 bits.

B. Weaknesses In MD5 – potential attacks

The research aims to develop a MD5 algorithm that can be used to create an electronic authentication system that makes this system more secure from other authentication system. Apparently, existing MD5 is vulnerable to various attacks, including attacks with brute force, rainbow schedule, dictionary, Christmas, etc. The research focuses on eliminating the weaknesses that are inherent in the current MD5 algorithm, thereby ensuring data integrity and security.

Section 4 and 5 – Proposed modifications and improvements in md5

In this work a novel five different methods are used to generate a new different robust hash from MD5 function.

Methods 1-5

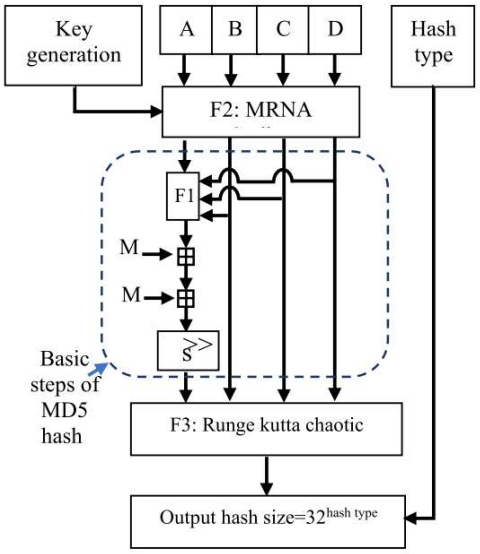


FIGURE 4. Show the general block diagram of the proposed system.

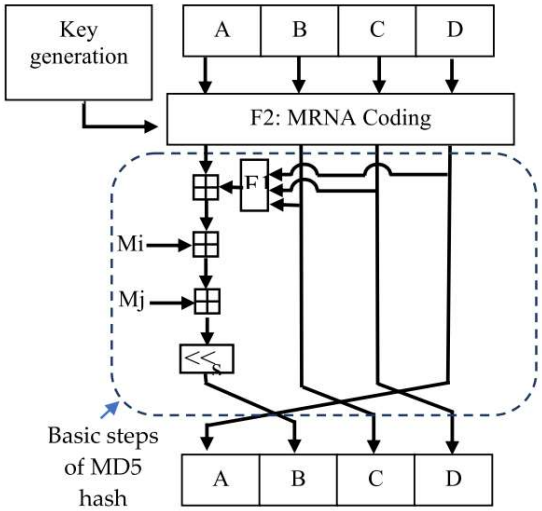


FIGURE 5. Shows the general block diagram of the proposed system (Method 2).

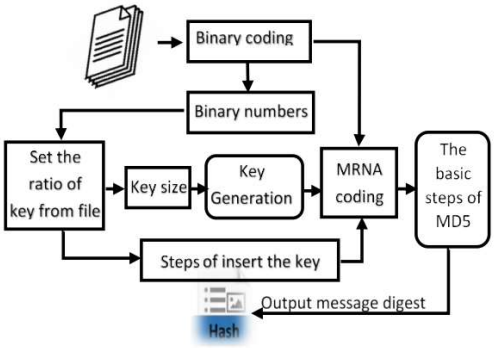


FIGURE 6. Shows the flow chart of general structure of the proposed system (Method 4).

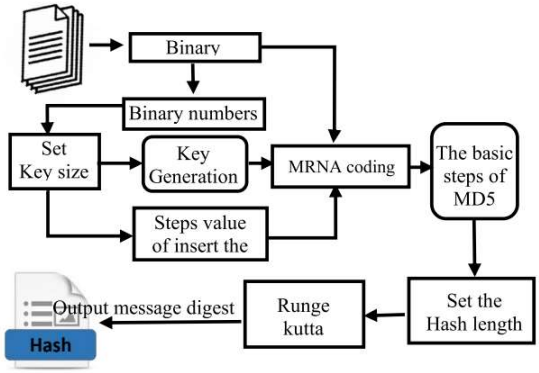


FIGURE 7. Shows the flow chart of the proposed system method 5.

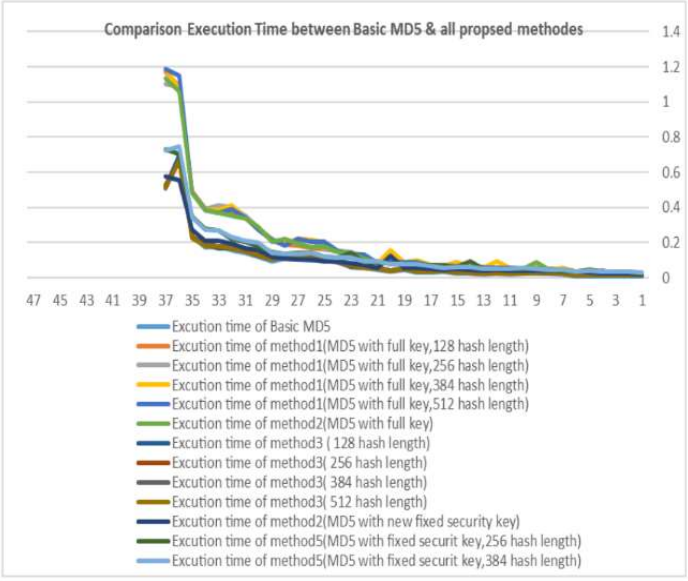


FIGURE 9. Comparison of the run time between the multiple development methods with the actual way.

Method 1: This method exposed the fusion of MD5 with chaotic theorem (Fourth-order Runge-Kutta method that used to solve the differential equation) & new security key; the key size that used in this method is the same as the file size the main structure of this method demonstrated in figure 1 above. The output hash length is (128-bit, 256-bit, 384-bit, 512-bit . . . etc).

Method 2: this method exposed the fusion of MD5 with a new security key that used in method 1 that was explained previously in the section (5. Methodology & Key generation), the length of key is the same of plain text. The output hash length from method 2 is (128 bit) and this length is the same length that created by Basic MD5 hash but with different values.

Method 3: This method creates MD5 hash value by fusion of Basic steps of MD5 hash function with chaotic theorem only. The chaotic plane depends on (Fourth-order RungeKutta method that used to solve the differential equation) that used in method1. The output hash length is (128-bit, 256-bit, 384-bit,512-bit . . . etc). The value of method 3 differs from the value generated by method 1, which can be observed clearly in experimental results.

Method 4: This method is showing a light version of method 2 that fusion of MD5 with a new security key that used in method 1 that was explained previously in the section (5. Methodology & Key generation), the length of key is fixed that is predetermined by the user. The key is engaged and injected into the text to be encrypted by performing a function XOR and the insertion key take two form (other fixed steps or fixed ratio from the original file) the same of plain text. But the output hash length from method 4 is (128 bit) and this length is the same length that created by Basic MD5 hash but with different values.

Method 5: This method is showing a light version of method 1 because it used two above methods to get the output hash value these methods are Method 3 & method 4 that fusion of MD5 with chaotic theorem & new fixed security key. The output hash length is (128-bit, 256-bit, 384-bit,512- bit . . . etc). The value of method 3 is differ from value the generated from method 1 and method 3.

## Section 6 - Performance evaluation

This section discusses the suggested approach for performance evaluation, which show the robustness of the algorithm as compared to other types of hash algorithms.

Characteristics of good performance used for the research:

### A. Weakness of confidentiality requirements for the system

In order for the system to be confidential, the designer of the system must take into account in its design to meet the basic requirements that achieve the confidentiality of the system; which is confusion and diffusion. These make sure that all logical relationships are cancelled and the cryptanalyst cannot identify the decryption algorithm and find the key.

### B. The simplicity of the key

The simplicity of the key and the ease with which it is remembered are the most important characteristics of which the key is preferred. In the proposed algorithm, the choice

of keys depends on the diversity of features that give to the key the strength and the simplicity at the same time.

C. The complication of both the encryption and decryption method

A suitable encryption system should be easy and flexible to decrypt, though, at the same time, the process of analysing and breaking the code should be difficult and not possible.

Section 7 and 8 - Experimental results and results discussion

Researchers claim that a major advantage of MD5's improved algorithm is, expansion to any size greater than 64-bit key block length so that the output is safer against any attack. The proposed improvement and development of the algorithm has been tested through several experiments shown in Table 1. Table 1 shows a comparison of multiple methods used to improve the actual MD5 algorithm. The use of this method had high flexibility in outputting the desired length with different values, as we can obtain the length of 128, 254, 384, 512, 1024 . . . etc. from the hash values.

**TABLE 1. Illustration of the different output bits in enhancing the MD5 algorithm that gives the same structure to other.**

| Type of method | Number of output Bits in new enhancement md5 | | | | The output digit of enhance md5 is the same structure to: | |
|---|---|---|---|---|---|---|
| new MD5 | 8 digits | 9 digits | 10 digits | 11 digits | 8 digits | 10 digits |
| 1 | 128 | 144 | 160 | 176 | MD4 or MD5 | SHA-1 |
| 2 | 256 | 272 | 288 | 304 | SHA-2-256 or FSB-256 | |
| 3 | 384 | 400 | 416 | 432 | SHA-2-384 or FSB-384 or ECOH-384 | |
| 4 | 512 | 528 | 544 | 2048 | SHA-2-512 or FSB-512 | |
| 8 | 1024 | 1040 | 1056 | 1072 | VSDL-1024 | |
| 16 | 2048 | 2064 | 2080 | 2096 | VSDL-2048 | |

Section 9 - Conclusion

MD5 is one of the functions or techniques of one-way segmentation that is used in many fields and in different applications to maintain data integrity by converting plain text or data into encrypted text that is generated in the form of unique hash data. The length of the output of the MD5 algorithm is often considered a weak point of the algorithm. Researchers say that MD5 algorithm suffers from attacks because of the smaller values of the hash digest. The improvement in the proposed MD5 algorithm will result in an improved collision resistance and access to the highest levels of security by providing the distribution and creation of bits such that it is difficult for attackers and hackers to predict and change individual data.

Overall, their work is an important asset to the MD5 research and the study of prospective improvements in MD5 hash function. They seem to have missed more important citations. The work was not proof-read for common grammatical and punctuation mistakes and has an unconventional flow of ideas and vocabulary, which make it difficult to follow for novice researchers and casual readers. Approach to the problem is realistic, main concept is explained very clearly. Results discussion and analysis are more than sufficient. The research work inspires further elaboration and research in improving existing hash functions.

***