ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 9, May 2016

Comparison of MD5 and Blowfish Algorithm

Deepthi[1], Pradyumna G.R[2]

P.G. Student, Dept of ECE, NMAM Institute of Technology, Nitte, Udupi, Karnataka, India[1]

Assistant Professor, Dept of ECE, NMAM Institute of Technology, Nitte, Udupi, Karnataka, India[2]

ABSTRACT: The need for data has increased with the increase in the number of users and data. So many cryptographic algorithms are implemented to provide better security at lower cost. Thus in this paper, cryptographic algorithms MD5 and Blowfish are studied. They are then compared based on different parameters and are then considered for FPGA implementation.

KEYWORDS: Encryption, Decryption, MD5, Blowfish.

I. INTRODUCTION

Cryptography is a technique to store and transmit information in such a form which can be read and processed only by the intended receiver. It is the art of changing the plain text into unreadable message for a secure communication. The objectives of cryptography are data confidentiality, integrity, non-repudiation and authentication. Cryptography is completely based on mathematical concept and science. So, practically it is very difficult to break the message by any third party. Before, cryptography was only concerned with encryption i.e., message confidentiality. The earliest form of cryptographic algorithms involved more than secret writing as people couldn't read. Recently, cryptography has involved a lot and there is beyond confidentiality like digital signature, authentication etc [1]. The modern cryptography is divided into different areas like private-key cryptography, public-key cryptography, cryptanalysis, cryptographic hashes and so on.

Cryptographic algorithms are classified into two parts: Symmetric-key cryptography and Asymmetric-key cryptography. Symmetric-key as its name suggests it uses same key for both encryption process and decryption process of the data. The strength of this Symmetric-key cryptography is based on the key used. If weak keys are used then the data can be easily decrypted. Symmetric algorithms are divided as block cipher and stream cipher. Examples of Symmetric algorithm are Advanced Encryption Standard (AES), Digital Encryption Standard (DES), Blowfish, and RC4. Asymmetric-key cryptography, here two keys are required; public-key for encryption process and private-key for decryption.Examples of Asymmetric algorithm are RSA, Diffie-Hellman algorithm.

The objectives of cryptography are authentication, data confidentiality, integrity and non-repudiation. Authentication means making sure whether the message came from the intended receiver or no. Confidentiality is that the message can be accessed only by authorized persons and not by everyone. Integrity is that the message cannot be altered by anyone in between the transmission. Non-repudiation is that making sure neither the receiver nor the sender can deny the transmission.In cryptography, Message Authentication (MAC) is an important method to verify whether the received message came from the claimed source and are not altered ones [2]. The use of a message authentication code is the key element for authentication schemes. One of the methods to produce MAC is to use hash functions.

II. RELATED WORK

J. Deepakumara et al., 2001 [2] in their paper "FPGA Implementation of MD5 Algorithm" stated that MD5 is used in Internet Protocol Security, as the basis for an HMAC. There is a drastic increase in the interest for high-speed cryptographic accelerators in IPSEC applications, for example Virtual Private Networks. He explained that it is cheaper to build cryptographic accelerators using hardware implementations of HMACs based on hash algorithm such as MD5.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 9, May 2016

D. Cao et al., 2010 [3] in their paper "Design and Implementation for MD5-based data integrity checking system" provided the method of data integrity checking using MD5 in order to make sure that the data of information is in correct format and state. He also explains the principle of MD5 algorithm, researches a way to use data integrity checking method on the protection of information system.

R. Saeed et al., 2013 [4] in their paper "VHDL Processor for Covering Information Using MD5 Algorithm" focused on security factor. The efficiency of the MD5 algorithm urged us to use it to cover the information which needs to be secure. MD5 is used with many covering algorithms in four scenarios which proposed to increase the degree of security.
A. Noaman, 2013 [5] in his paper "A VHDL Module for Implementation of MD5 Algorithm" proposed the application of VHDL to implement and to computing MD5 for data integrity checking method and to make sure that the data of an information systems are in a proper state.
T. Aurora et al., [6] in their "Blowfish Algorithm" paper gave a brief description about Blowfish by designing and implementing with the help of algorithm.

III.BASIC TERMS

1). Plain Text
The actual information to be transmitted to a specified receiver is the plain text.Like, if a sender wants to send a message "good morning" to another person, then here "good morning" is the plain text.
2). Cipher Text
The meaningless message that is obtained at the receiver side after the transmission is the cipher text. For example, the cipher text for the plain text "good morning" would be "55tlh%%fg@t1".
3). Encryption
Conversion of the plain text into cipher text at the sender side is called as encryption. Here for the encryption process two things are required: an algorithm and a key.
4). Decryption
Decryption is reverse process of encryption; the cipher text is converted back into its original form (plain text). Even decryption method requires analgorithm and a key. Usually this algorithm will be similar for both encryption and decryption with slight modifications if necessary.
5). Key
A key is used during the encryption process for the conversion of plain text into cipher text and during the decryption process for the conversion of the cipher text into plain text. The key size can be of any length based on the algorithm used.
6). Hash Functions
Hash is a process, signature, function that translates information into cryptic value.

IV.CRYPTOGRAPHIC ALGORITHMS

There are many cryptographic algorithms that are used for encryption process and decryption process. These algorithms come under any one for these categories: symmetric-key cryptosystem or asymmetric-key cryptosystem. Symmetric-key algorithm uses only a single key for both the encryption process and decryption process. Hence the strength of these algorithms is based on the key used. In asymmetric-key cryptography different keys are being used for both encryption and decryption.Hash is an algorithm where a block of data is taken and fixed size bit string is returned. It is a one way process. In hashing, the data to be encrypted is called message and the result after has function is called the message digest. Different hash functions are MD5, SHA.

A. MD5 ALGORITHM

MD5 is a cryptographic hash algorithm widely used in the field of authentication security [3]. MD5 algorithm takes an input of any length and produces 128 bits output.The core of MD algorithm is a hash function, which compresses an input of random length for an input of constant length. It should have the properties such as:

i. Computation should be easy.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 9, May 2016

ii. Should be hard enough for the message to be computed from its digest.
iii. Should be difficult to find another input with same digest [5].

MD5 contains an operation number of 64 that is grouped in 4 rounds of 16 operations. F is a non-linear function. One of its functions is used in each round. The 32 bits constant is denoted by K_i, the input message of 32 bits block is denoted by M_i which is different for each operation [4]. The left bit rotation by s places is denoted by s. Four functions are possible, a different one used for each round. The following steps illustrate how to calculate the message digest of the message:

1. Appending the padding bits.
2. Appending the length.
3. Initialization of MD buffer.
4. The message is processed in 16 word blocks.
5. Output.

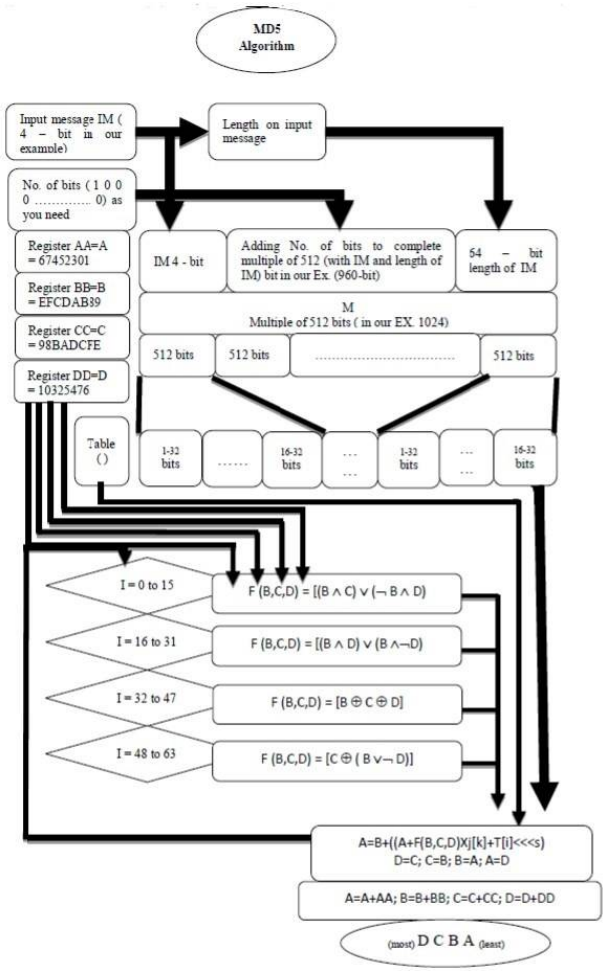

Figure 1: Flowchart of MD5 Algorithm

The implementation details of the algorithm are as follows:
MD5 algorithm flowchart shown in Figure 1 consists of the following steps:

1. Check the no. of bits (length) of the original message (NOM).

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 9, May 2016

2. Add No. of bits to the input message (IM), so a total length of the result data equal to the multiple of 512 (the add bit are 1 0 0 …….0).
3. Add 64 bits which represent the length of the IM to the result of point 2 the final will represent by M, where M (multiple of 512 bits).

4. Divided M to blocks (B), each one has 512 bits.
5. Divided (B) to 16 blocks (X) each one have 32 bits.
6. The algorithm steps have 4 rounds and each round has 16 steps. The total no. of steps is 64.
7. There are four 32 bits shift registers, each one has initial (Hex.) values as:
   a. $Reg.A = [6\,7\,4\,5\,2\,3\,0\,1]32 - bits\ [A] = [D]'$
   b. $Reg.B = [e\ f\ c\ d\ a\ 8\ 9]32 - bits\ [B] = [C]'$
   c. $Reg.C = [9\,8\ b\ a\ d\ c\ f\ e]32 - bits\ [C] = [B]'$
   d. $Reg.D = [1\,0\,3\,2\,5\,4\,7\,6]32 - bits\ [D] = [A]'$
8. The values of *A, B, C* and *D* are stored temporarily in *AA, BB, CC,* and *DD* registers respectively.
9. This algorithm includes 4 "rounds" of processing. These rounds have same structure but each has different functions F, G, H and I. The operation in single step involve the functions:

$$A = B + \left(\left(A + F(B,C,D) + X_J[K] + T[i]\right) <<< s \right. \quad (1)$$

$$D = C; C = B; B = A;\ A = D \quad (2)$$

Where:
$X_J[k]$ - denote the k^{th} 32 bits word of X_J
$<<< s$ -circular shift left by s bits.
In the end of four rounds, the output is added to the input of first round

$A=A+AA;\ B=B+BB;\ C=C+CC;\ D=D+DD$ (3)

10. Then, the outputs are 128 bits will arrange as follows:

$$\underset{(most)}{D\ C\ B\ A}\ \underset{(least)}{}$$

B. BLOWFISH ALGORITHM

Blowfish is an encryption algorithm which uses variable key size of 32bits to 448 bits. Blowfish was designed by Bruce Schneier in 1993. A good encryption rate is obtained in software [7].The use of different key size up to the length of 448 bits is an advantage to Blowfish. Blowfish is a symmetric algorithm, which means it uses the same key for both encryption process and decryption process. Even though the algorithm is simple, the implementation of it is complex.Blowfish algorithm is a Feistel network that takes an input of 64 bits for encryption using a key of any length form 32 bits to 448 bits. The total number of rounds is 16 rounds.

In any implementation, there are many separate circuit pieces that can be identified:

- Encryption core
- Function F(xL)
- Generated array of sub-keys
  The sub-keys that are to be used must becomputed before any process.The P-array consists of one18 entry sub-keys: P1, P2,..., P18. Each sub-keyis 32 bits [9].
- Key-dependent S-boxes
  There are 4 S-boxes with each having 256 entries as:
  S1,0, S1,1,….., S1,255;
  S2,0, S2,1,….., S2,255;
  S3,0, S3,1,....., S3,255;
  S4,0, S4,1,…..., S4,255.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 9, May 2016

These S-boxes are of 32 bits.

- Control logic

Consider a 64 bits input 'x'. This input bit is divided into two parts: xL and xR, each 32 bits. Then for each round,
xL = xLXOR Pi
xR = F(xL) XOR xR
Then swapingxL and xR. After the16th round, undo the last swap. After that
xR = xR XOR P17
xL = xL XOR P18.
Finally, combine the 32 bits xL and xR to get the 64 bits output (cipher-text).

V. COMPARISON OF ALGORITHMS

Table 1: Comparison of Algorithms

Algorithm	Block Size	Rounds	Key	Type	Possible Attacks
MD5	128 bits	4 Rounds	No key is used	Hashing Algorithm	Birthday Attacks
Blowfish	64 bits	16 Rounds	32-448 bits	Symmetric-key	Reflection Attacks

Table 1 illustrates the comparison of MD5 and Blowfish algorithm based on block size, key and number of rounds [8].

VI. SIMULATION RESULTS

The simulation of these two algorithms was done. From this implementation, maximum performance of each algorithm was optimized. For MD5, the total fuse memory usage is 32,480 KB and the total fuse CPU usage is 937 ms.For Blowfish, the total fuse memory usage is 37,748 KB and the total fuse CPU usage is 1,796ms.

Figure 2: Simulation Results of MD5 Algorithm Using Xilinx.

Figure 2 shows the simulation results of MD5 algorithm using Xilinx ISE.Here the input given is a string "abc". The ASCII value of "abc" is given as the input, "616263". The MD5 hash output for "abc" is "900150983cd24fb0d6963f7d28e17f72".

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 9, May 2016

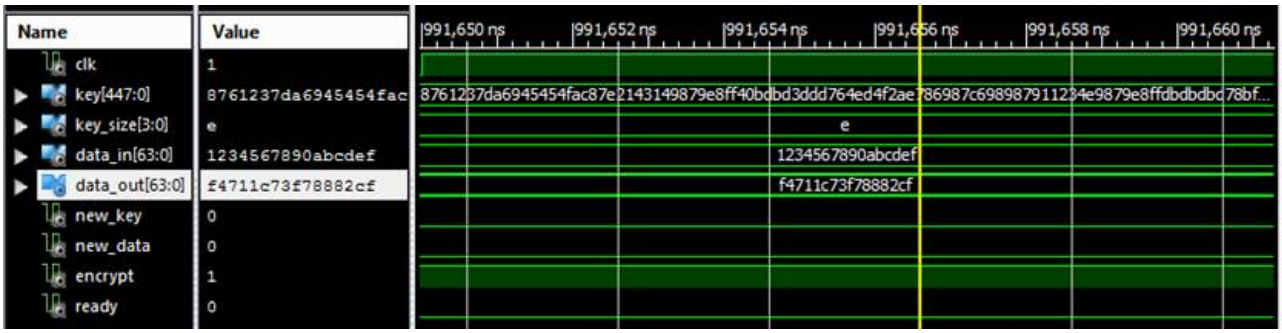

Figure 3: Simulation Result of Blowfish Algorithm using Xilinx.

Figure 3 shows the simulation results of Blowfish algorithm using Xilinx ISE.The input given is a 64 bit data_in, and is encrypted using a 448 bit key to get the 64 bitencrypted output data_outwhen the clock signal clk is "1".

VII. APPLICATIONS

MD5 is used widely in the field of software for better security. It is being used to give a clear idea that the information that was sent was received by the intended receiver. Its application can also be in the field of electronic discovery, where an identifier can be given to each file that is being sent during legal process. Blowfish algorithm is fast but is slow only when changing keys. But this slow process when changing keys is also an advantage where applications uses slow key scheduling, such as in 'OpenBSD'.

VIII.CONCLUSION

The obtained results show that MD5 is superior to Blowfish algorithm in terms of the total fuse usage, CPU fuse usage. MD5 algorithm is considered the best algorithm compared to Blowfish algorithm. Blowfish would be the strongest algorithm if implemented in hardware as it does not have any known cryptanalysis up to date.

REFERENCES

[1] William Stallings, "Cryptography and Network Security Principles and Practices", Fourth Edition, Pearson Education, Prentice Hall, 2009.
[2] J. Deepakumara, H.M. Heys, and R. Venkatesan, "FPGA Implementation of MD5 Hash Algorithm", Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering, pp.919-924, 2001.
[3] Danyang Cao, Bingru Yang, "Design and Implementation for MD5-based Data Integrity Checking System", 2nd IEEE International Conference on Information Management and Engineering,pp.608-611, 2010.
[4] Thamir R. Saeed, Hashmea S. Dakel, Najmah A. Beeb, Ivan A. Hashim and Jawad K. Ali, "VHDL Processor for Covering Information Using MD5 Algorithm", International Journal of Engineering & Science Research, pp. 402-408, 2014.
[5] Noaman M. A., "A VHDL Model for Implementation of MD5 Hash Algorithm", Engineering & Technology Journal, pp. 1107-1116, 2013.
[6]Tanjyot Aurora, Parul Aurora, "Blowfish Algorithm", International Journal ofComputer Science and Communication Engineering, pp.238-243, 2013.
[7] AnkitaDespande, P.S. Choudhary, "FPGA Implementation of BlowfishCryptographic Algorithm", International Journal of Advanced Research inComputer Science and Software Engineering, pp.542-547, 2014.
[8] L. KranthiKiran, J.E.N. Abhilash, P. Suresh Kumar, "FPGA Implementation ofBlowfish Cryptosystem Using VHDL", International Journal of EngineeringResearch & Technology, pp. 1-5, 2013.