6

by Nugrahini S

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

HASHING VARIABLE LENGTH APPLICATION FOR MESSAGE SECURITY COMMUNICATION

Robbi Rahim[1], Akbar Iskandar[2], Firman Aziz[3], Erwinsyah Satria[4], Wildan Mahir Muttaqin[5], S. Sujito[5], Folkes E. Laumal[6], Dwie Retna Suryaningsih[7], Nugrahini Susantinah[7], Agustinus Suradi[8] and Afiful Ikhwan[9]
[1]School of Computer and Communication Engineering, Universiti Malaysia Perlis, Arau, Malaysia
[2]STMIK AKBA, Makassar, Indonesia
[3]Universitas Pendidikan Indonesia, Bandung, Indonesia
[4]Universitas Bung Hatta, Indonesia
[5]Department of English Language Education, IAIN Surakarta, Indonesia
[6]Politeknik Negeri Kupang, Kupang, Indonesia
[7]Universitas Wijaya Kusuma Surabaya, Surabaya, Indonesia
[8]Universitas Widya Dharma, Klaten, Indonesia
[9]Muhammadiyah University of Ponorogo, East Java, Indonesia
E-Mail: usurobbi85@zoho.com

ABSTRACT

Security is still the most important priority in communicating globally on the network; all communication media such as social media today must apply various types of cryptographic algorithms to secure incoming and outgoing information. Hashing Variable Length is one algorithm that can be used to secure messages with the same length of results and also in addition to cryptography; this algorithm can also be used as message compression with very reliable security. Hashing Variable Length has an output with varying lengths and this study provides output results in the form of simulations to illustrate the results of security and compression performed.

Keywords: security, hashing variable length, compression, security message.

INTRODUCTION

Communication in the digital era is currently impossible without good security[1]-[4], various forms of security are applied to data communications so that messages or information sent do not fall into the hands of irresponsible parties[5]-[8]. Cryptography is a technique that can be done to secure messages by using various algorithms, one technique that can be used is Hashing Variable Length[9]. The use of these algorithms in addition to message security can also be used for message compression; this is possible because the output of the Hashing Variable Length algorithm is a constant length ciphertext.

Hashing Variable Length is one of several one way hash algorithms aside from the MD4, MD5 and SHA algorithms[9], [10], all of these algorithms produce the same digest message, the algorithm can be used as an authentication process and also a digital signature[11]-[13].

Hashing Variable Length was created by Zheng et.al with outputs varying from 128 bits to 256 bits and processing carried out also varies up to 5 times. The speed of the Hashing Variable Length algorithm process is based on Zheng's test 60% times faster than MD5 with 3 time's process.

The Hashing Variable Length algorithm used will be tested in different lengths of text, testing is done on application programs created using the Pascal object programming language to find out the security results of the Hashing Variable Length Algorithm. This research is expected to be able to make a real contribution from the application of the Hashing Variable Length algorithm in the form of applications.

THEORY

Cryptography

Cryptography is a field of science that studies about how to keep an important information secret in a form that cannot be read by anyone and returns it back to its original information by using various techniques that have been available so that the information cannot be known by any party who is not the owner or unauthorized[14]-[16].

Cryptography learns about mathematical techniques that relate to aspects of information security such as confidentiality, data integrity, data sender / receiver authentication, and data authentication[17]-[19]. With the development of cryptography, the division between what is included in cryptography and what has not become blurred. Today, cryptography can be considered as a combination of engineering studies and applications that depend on the existence of difficult problems[20]-[23].

For most people, cryptography is preferred in keeping communication confidential. As is well known and agreed that protection against sensitive communication has become a cryptographic emphasis so far[24]. However, this is only part of today's cryptographic application. Cryptography is a study related to 4 security aspects of an information namely confidentiality, data integrity, authentication, and non-repudiation[25], [26].

VOL. 14, NO. 1, JANUARY 2019 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Cryptography can be classified into 2 types of systems based on the type of key used, namely public key cryptography and secret key cryptography. In a secret key cryptographic system, also known as the symmetric cryptosystem, the sender and recipient together agree on a secret key that will be used in the encryption and decryption process without being known by other parties. Whereas in the public key cryptography system, known as assymmetric cryptosystem, the sending and receiving parties get a key pair of public keys and private keys where the public key is published and the secret key remains confidential[27], [28].

Cryptography application

Cryptography is now widely implemented in various applications, especially in terms of data security. Systems like this can have varying degrees of complexity. Some applications are simpler, among others; secure communication, identification, authentication, and secret sharing. More complicated applications such as systems for e-commerce, certification, secure electronic mail, key discovery and secure computer access[19], [25].

Hash function

Hashing besides being used for message authentication can also be used to generate passphrase-based keys. The value of the hash function represents a message that is shorter than the document from which the value is calculated, this value is often called a message digest. Message digest can be considered as a "digital fingerprint" from a longer document[29], [30].

The role of hash functions in cryptography is in terms of checking conditions for message integrity and digital signatures. A digest can be made public without showing the contents of the document from which digest, this is very important in digital time stamping where by using a hash function, one can obtain documents with time stamped documents without showing the contents of the document to the time stamping service provider. In the case of designing a hash function there is a compression function term, a compression function is a compression function that uses input strings of a certain length and produces shorter strings. In this process, a message of any length is broken into several blocks of length depending on the compression function and "padded" (for security reasons) so that the message size is multiplication of the block size. The blocks are then processed sequentially, by taking the results of the hash so far as the input and block of the current message

Hashing variable length

HAVAL is one of the one-way hash functions created by Zheng et al, with a maximum output length of 256 bits and can be processed as many as 3.4 and 5 times. as an example of the algorithm testing process as follows:

Message: *Keamanan Itu Sangat Penting*

Split message to w variable:

```
'Keam' -> w( 0) = 4B65616D
'anan' -> w( 1) = 616E616E
' Itu' -> w( 2) = 20497475
' San' -> w( 3) = 2053616E
'gat ' -> w( 4) = 67617420
'Pent' -> w( 5) = 50656E74
'ing ' -> w( 6) = 696E6720
```

Initial Value:

```
K0 = X0 = 243F6A88
K1 = X1 = 85A308D3
K2 = X2 = 13198A2E
K3 = X3 = 03707344
K4 = X4 = A4093822
K5 = X5 = 299F31D0
K6 = X6 = 082EFA98
K7 = X7 = EC4E6C89
```

1. First round hashing variable length

FF(X7, X6, X5, X4, X3, X2, X1, X0, W0)
FF(EC4E6C89,082EFA98,299F31D0,A4093822,03707344,13198A2E,85A308D3,243F6A88,4B65616D)
(1) Temp = F_Phi(082EFA98,299F31D0,A4093822,03707344,13198A2E,85A308D3,243F6A88)
Temp = F(85A308D3,243F6A88,03707344,299F31D0,082EFA98,13198A2E,A4093822)
Temp = (13198A2E AND 03707344) XOR (082EFA98 AND 243F6A88) XOR (299F31D0 AND 85A308D3) XOR (A4093822 AND 13198A2E) XOR A4093822
Temp = A6BD585C
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w
A7 = (A6BD585C >>> 7) + (EC4E6C89 >>> 11) + 4B65616D
A7 = 95F065EA

FF(X6, X5, X4, X3, X2, X1, X0, X7, W1)
FF(082EFA98,299F31D0,A4093822,03707344,13198A2E,85A308D3,243F6A88,95F065EA,616E616E)
(1) Temp = F_Phi(299F31D0,A4093822,03707344,13198A2E,85A308D3,243F6A88,95F065EA)
Temp = F(243F6A88,95F065EA,13198A2E,A4093822,299F31D0,85A308D3,03707344)
Temp = (85A308D3 AND 13198A2E) XOR (299F31D0 AND 95F065EA) XOR (A4093822 AND 243F6A88) XOR (03707344 AND 85A308D3) XOR 03707344
Temp = 26C872C6
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w
A7 = (26C872C6 >>> 7) + (082EFA98 >>> 11) + 616E616E
A7 = 40BCF832

FF(X5, X4, X3, X2, X1, X0, X7, X6, W2)
FF(299F31D0,A4093822,03707344,13198A2E,85A308D3,243F6A88,95F065EA,40BCF832,20497475)
(1) Temp = F_Phi(A4093822,03707344,13198A2E,85A308D3,243F6A88,95F065EA,40BCF832)

Temp = F(95F065EA,40BCF832,85A308D3,03707344,A4093822,243F6A88,13198A2E)
Temp = (243F6A88 AND 85A308D3) XOR (A4093822 AND 40BCF832) XOR (03707344 AND 95F065EA) XOR (13198A2E AND 243F6A88) XOR 13198A2E
Temp = 165BD1C4
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w
A7 = (165BD1C4 >>> 7) + (299F31D0 >>> 11) + 20497475
A7 = E27B5FFE
FF(X4, X3, X2, X1, X0, X7, X6, X5, W3)
FF(A4093822,03707344,13198A2E,85A308D3,243F6A88,95F065EA,40BCF832,E27B5FFE,2053616E)
(1) Temp = F_Phi(03707344,13198A2E,85A308D3,243F6A88,95F065EA,40BCF832,E27B5FFE)
Temp = F(40BCF832,E27B5FFE,243F6A88,13198A2E,03707344,95F065EA,85A308D3)
Temp = (95F065EA AND 243F6A88) XOR (03707344 AND E27B5FFE) XOR (13198A2E AND 40BCF832) XOR (85A308D3 AND 95F065EA) XOR 85A308D3
Temp = 065BB3FF
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w
A7 = (065BB3FF >>> 7) + (A4093822 >>> 11) + 2053616E
A7 = 22B499FC

B. Second round hashing variable length

GG(X7, X6, X5, X4, X3, X2, X1, X0, W5, 452821E6)
GG(0291618C,A89CA652,BD251E3F,45543CCE,95972321,C5117677,04D7CBF0,0A14B23E,50656E74,452821E6)
(1) Temp = G_Phi(A89CA652,BD251E3F,45543CCE,95972321,C5117677,04D7CBF0,0A14B23E)
Temp = G(45543CCE,C5117677,04D7CBF0,0A14B23E,BD251E3F,95972321,A89CA652)
Temp = (95972321 AND BD251E3F AND 0A14B23E) XOR (BD251E3F AND 04D7CBF0 AND C5117677) XOR (95972321 AND BD251E3F) XOR (95972321 AND 04D7CBF0) XOR (BD251E3F AND 45543CCE) XOR (0A14B23E AND C5117677) XOR (04D7CBF0 AND C5117677) XOR (A89CA652 AND BD251E3F) XOR A89CA652
Temp = 940ACD19
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w + c
A7 = (940ACD19 >>> 7) + (0291618C >>> 11) + 50656E74 + 452821E6
A7 = FA35F820

GG(X6, X5, X4, X3, X2, X1, X0, X7, W14, 38D01377)
GG(A89CA652,BD251E3F,45543CCE,95972321,C5117677,04D7CBF0,0A14B23E,FA35F820,20202020,38D01377)
(1) Temp = G_Phi(BD251E3F,45543CCE,95972321,C5117677,04D7CBF0,0A14B23E,FA35F820)
Temp = G(95972321,04D7CBF0,0A14B23E,FA35F820,45543CCE,C5117677,BD251E3F)
Temp = (C5117677 AND 45543CCE AND FA35F820) XOR (45543CCE AND 0A14B23E AND 04D7CBF0) XOR (C5117677 AND 45543CCE) XOR (C5117677 AND 0A14B23E) XOR (45543CCE AND 95972321) XOR (FA35F820 AND 04D7CBF0) XOR (0A14B23E AND 04D7CBF0) XOR (BD251E3F AND 45543CCE) XOR BD251E3F
Temp = B8305E51
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w + c
A7 = (B8305E51 >>> 7) + (A89CA652 >>> 11) + 20202020 + 38D01377
A7 = C6B5A7E7

GG(X5, X4, X3, X2, X1, X0, X7, X6, W26, BE5466CF)
GG(BD251E3F,45543CCE,95972321,C5117677,04D7CBF0,0A14B23E,FA35F820,C6B5A7E7,20202020,BE5466CF)
(1) Temp = G_Phi(45543CCE,95972321,C5117677,04D7CBF0,0A14B23E,FA35F820,C6B5A7E7)
Temp = G(C5117677,0A14B23E,FA35F820,C6B5A7E7,95972321,04D7CBF0,45543CCE)
Temp = (04D7CBF0 AND 95972321 AND C6B5A7E7) XOR (95972321 AND FA35F820 AND 0A14B23E) XOR (04D7CBF0 AND 95972321) XOR (04D7CBF0 AND FA35F820) XOR (95972321 AND C5117677) XOR (C6B5A7E7 AND 0A14B23E) XOR (FA35F820 AND 0A14B23E) XOR (45543CCE AND 95972321) XOR 45543CCE
Temp = CD52C4E9
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w + c
A7 = (CD52C4E9 >>> 7) + (BD251E3F >>> 11) + 20202020 + BE5466CF
A7 = 7A06D11B

GG(X4, X3, X2, X1, X0, X7, X6, X5, W18, 34E90C6C)
GG(45543CCE,95972321,C5117677,04D7CBF0,0A14B23E,FA35F820,C6B5A7E7,7A06D11B,20202020,34E90C6C)
(1) Temp = G_Phi(95972321,C5117677,04D7CBF0,0A14B23E,FA35F820,C6B5A7E7,7A06D11B)
Temp = G(04D7CBF0,FA35F820,C6B5A7E7,7A06D11B,C5117677,0A14B23E,95972321)
Temp = (0A14B23E AND C5117677 AND 7A06D11B) XOR (C5117677 AND C6B5A7E7 AND FA35F820) XOR (0A14B23E AND C5117677) XOR (0A14B23E AND C6B5A7E7) XOR (C5117677 AND 04D7CBF0) XOR (7A06D11B AND FA35F820) XOR (C6B5A7E7 AND FA35F820) XOR (95972321 AND C5117677) XOR 95972321
Temp = 6EB39372
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w + c
A7 = (6EB39372 >>> 7) + (45543CCE >>> 11) + 20202020 + 34E90C6C

VOL. 14, NO. 1, JANUARY 2019 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

A7 = D3AF3E39

GG(X3, X2, X1, X0, X7, X6, X5, X4, W11, C0AC29B7)
GG(95972321,C5117677,04D7CBF0,0A14B23E,FA35F820,C6B5A7E7,7A06D11B,D3AF3E39,20202020,C0AC29B7)
(1) Temp = G_Phi(C5117677,04D7CBF0,0A14B23E,FA35F820,C6B5A7E7,7A06D11B,D3AF3E39)
Temp = G(0A14B23E,C6B5A7E7,7A06D11B,D3AF3E39,04D7CBF0,FA35F820,C5117677)
Temp = (FA35F820 AND 04D7CBF0 AND D3AF3E39) XOR (04D7CBF0 AND 7A06D11B AND C6B5A7E7) XOR (FA35F820 AND 04D7CBF0) XOR (FA35F820 AND 7A06D11B) XOR (04D7CBF0 AND 0A14B23E) XOR (D3AF3E39 AND C6B5A7E7) XOR (7A06D11B AND C6B5A7E7) XOR (C5117677 AND 04D7CBF0) XOR C5117677
Temp = 3BA58015
(2) A7 = (Temp >>> 7) + (A7 >>> 11) + w + c
A7 = (3BA58015 >>> 7) + (95972321 >>> 11) + 20202020 + C0AC29B7
A7 = 6F7647BB

Process above will be done until encoding text will get result in ASCII = §ˆ• Ç%é9X‰öÚî□L_

RESULT AND DISCUSSIONS

Testing the message security application using the Hashing Variable Length algorithm can be seen in Figure 1 below.

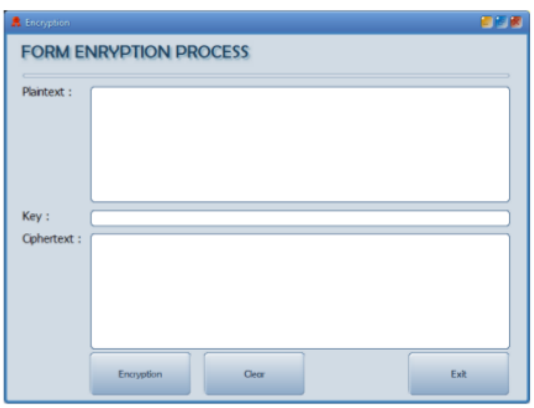

Figure-1. Main encryption form.

The encryption process using the Hashing Variable Length algorithm is done by giving a sample message to be encrypted as in Figure-2.

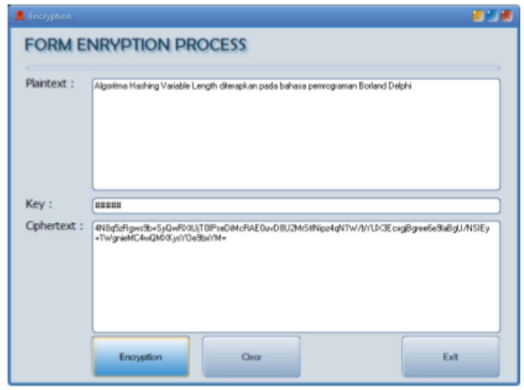

Figure-2. Encryption process.

Figure-2 displays the results of the encryption with the example message "Hashing Variable Length Algorithm applied to the Borland Delphi programming language" with a specific key.

An experiment decryption of ciphertext using the Hashing Variable Length algorithm can be seen in Figure-3.

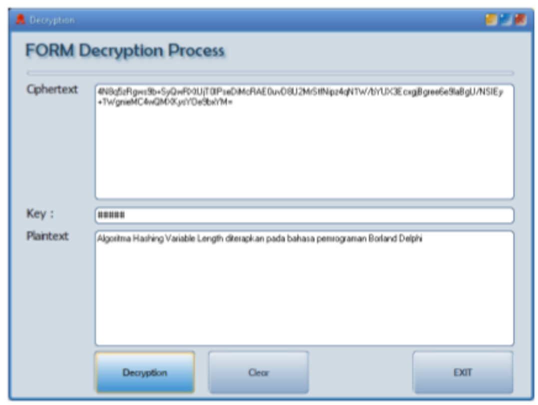

Figure-3. Decryption process.

Based on testing carried out the process of encryption and decryption using the Hashing Variable Length algorithm is very fast and with a size smaller than the size of the plaintext.

CONCLUSIONS

The Hashing Variable Length Algorithm can secure messages with and is suitable for use in communications carried out on the network and encrypted with a compressed hashing variable length algorithm so that if the message delivery process is not bandwidth intensive, the next development can be done by adding other cryptographic algorithms such as MARS, GOST, MISTY.

REFERENCES

[1] R. Ratnadewi, R. P. Adhie, Y. Hutama, J. Christian and D. Wijaya. 2017. Implementation and performance analysis of AES-128 cryptography method in an NFC-based communication system. World Trans. Eng. Technol. Educ.15(2): 178-183.

[2] H. Delfs and H. Knebl. 2007. Information Security and Cryptography.Vol. 19.

[3] R. Rahim, M. Dahria, M. Syahril and B. Anwar. 2017. Combination of the Blowfish and Lempel-Ziv-Welch algorithms for text compression. World Trans. Eng. Technol. Educ. 15(3): 292-297.

[4] R. Rahim *et al*. 2018. Internet based remote desktop using INDY and socket component. Int. J. Eng. Technol.7(2.9): 44-47.

[5] R. Rahim. 2017. Man-in-the-middle-attack prevention using interlock protocol method. ARPN J. Eng. Appl. Sci. 12(22): 6483-6487.

[6] H. Nurdiyanto, R. Rahim and N. Wulan. 2017. Symmetric Stream Cipher using Triple Transposition Key Method and Base64 Algorithm for Security Improvement. J. Phys. Conf. Ser. 930(1): 012005.

[7] S. Marrapu, S. Sanakkayala, A. kumar Vempalli and S. K. Jayavarapu. 2018. Smart home based security system for door access control using smart phone. Int. J. Eng. Technol. 7(1): 249.

[8] K. Neeraja, P. Rama Chandra Rao, D. Suman Maloji and D. Mohammed Ali Hussain. 2018. Implementation of security system for bank using open CV and RFID. Int. J. Eng. Technol. 7(2-7) 187.

[9] R. Rahim. 2017. 128 Bit Hash of Variable Length in Short Message Service Security. Int. J. Secur. It's Appl. 11(1): 45-58.

[10] Y. Zheng, J. Pieprzyk, and J. Seberry. 1992. HAVAL - A One-Way Hashing Algorithm with Variable Length of Output. Adv. Cryptol. - AUSCRYPT '92. 718(December 1992): 83-104.

[11] H. Nurdiyanto and H. Hermanto. 2016. Signature recognition using neural network probabilistic. Int. J. Adv. Intell. Informatics. 2(1): 46-53.

[12] M. Mesran, M. Syahrizal, and R. Rahim. 2018. Enhanced Security for Data Transaction with Public Key Schnorr Authentication and Digital Signature Protocol. ARPN J. Eng. Appl. Sci. 13(11): 3839-3846.

[13] H. Nurdiyanto *et al*. 2018. Authentication Security in Radio Frequency Identification with IDEA Algorithm. IOP Conf. Ser. Mater. Sci. Eng. 384: 012042.

[14] R. R *et al.* 2018. Visual Cryptography with RSA Algorithm for Color Image. Int. J. Eng. Technol. 7(2.5): 65-68.

[15] R. Rahim, D. Adyaraka, S. Sallu, E. Sarimanah and M. M. Rahman. 2018. Tiny encryption algorithm and pixel value differencing for enhancement security message. Int. J. Eng. Technol. 7(2.9): 82-85.

[16] R. Rahim, D. Hartama, H. Nurdiyanto, A. S. Ahmar, D. Abdullah and D. Napitupulu. 2018. Keylogger Application to Monitoring Users Activity with Exact String Matching Algorithm. J. Phys. Conf. Ser.954(1): 012008.

[17] R. Rahim, N. Kurniasih, M. Mustamam, L. Andriany, U. Nasution, and A. H. Mu. 2018. Combination Vigenere Cipher and One Time Pad for Data Security. Int. J. Eng. Technol. 7(2.3): 92-94.

[18] A. Putera, U. Siahaan and R. Rahim. 2016. Dynamic Key Matrix of Hill Cipher Using Genetic Algorithm. Int. J. Secur. Its Appl. 10(8): 173-180.

[19] H. Nurdiyanto and R. Rahim. 2017. Enhanced pixel value differencing steganography with government standard algorithm. in 2017 3rd International Conference on Science in Information Technology (ICSITech). pp. 366-371.

[20] S. Manna and S. Dutta. 2014. A Stream Cipher based Bit-Level Symmetric Key Cryptographic Technique using Chen Prime Number. Int. J. Comput. Appl. 107(12): 975-8887.

[21] R. I. Al-Khalid, R. A. Al-Dallah, A. M. Al-Anani, R. M. Barham and S. I. Hajir. 2017. A Secure Visual Cryptography Scheme Using Private Key with Invariant Share Sizes. J. Softw. Eng. Appl. 10(01): 1-10.

[22] S. Bruce. 1996. Applied cryptography.

[23] D. Abdullah *et al*. 2018. Super-Encryption Cryptography with IDEA and WAKE Algorithm. J. Phys. Conf. Ser. 1019(1): 012039.

[24] R. Rahim *et al.* 2018. Combination Base64 Algorithm and EOF Technique for Steganography. J. Phys. Conf. Ser.1007(1): 012003.

[25] R. Rahim, D. Adyaraka, S. Sallu, E. Sarimanah and A. Hidayat. 2018. An application data security with lempel - ziv welch and blowfish. Int. J. Eng. Technol. 7(2.9): 71-73.

[26] E. Kartikadarma, T. Listyorini and R. Rahim. 2018. An Android mobile RC4 simulation for education. World Trans. Eng. Technol. Educ. 16(1): 75-79.

[27] H. Nurdiyanto, R. Rahim, A. S. Ahmar, M. Syahril, M. Dahria and H. Ahmad. 2018. Secure a Transaction Activity with Base64 Algorithm and Word Auto Key Encryption Algorithm. J. Phys. Conf. Ser. 1028(1): 012053.

[28] S. Sriadhi, R. Rahim and A. S. Ahmar. 2018. RC4 Algorithm Visualization for Cryptography Education. J. Phys. Conf. Ser. 1028(1): 012057.

[29] R. Rahim, I. Zulkarnain and H. Jaya. 2017. Double hashing technique in closed hashing search process. IOP Conf. Ser. Mater. Sci. Eng. 237(1): 012027.

[30] R. Rahim, Nurjamiyah and A. R. Dewi. 2017. Data Collision Prevention with Overflow Hashing Technique in Closed Hash Searching Process. J. Phys. Conf. Ser. 930(1): 012012.

6

ORIGINALITY REPORT

22%	20%	17%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	id.scribd.com Internet Source	10%
2	www.slideshare.net Internet Source	6%
3	Submitted to Multimedia University Student Paper	5%

Exclude quotes	Off	Exclude matches	< 5%
Exclude bibliography	Off		