Research Intuitions of Hashing Crypto System

[1]Rojasree. V, [2]Gnana Jayanthi. J, [3]Christy Sujatha. D
PG & Research Department of Computer Science,
Rajah Serfoji Govt. College(A),
(Affiliated to Bharathidasan University),
Thanjavur-613005, Tamilnadu, India.

***Abstract*—Data transition over the internet has become inevitable. Most data transmitted in the form of multimedia. The data transmission must be less complex and user friendly at the same time with more security. Message authentication and integrity is one of the major issues in security data Transmission. There are plenty of methods used to ensure the integrity of data when it is send from the receiver and before it reaches the corresponding receiver. If these messages are tampered in the midst, it should be intimated to the receiver and discarded. Hash algorithms are supposed to provide the integrity but almost all the algorithms have confirmed breakable or less secure. In this review, the performances of various hash functions are studied with an analysis from the point of view of various researchers.**

Keywords—Hash Function; Message Authentication Code; Message Digest; Secure Hash Algorithm; Message Integrity; RIPEMD; WHIRLPOOL; BLAKE; Cuckoo; JH; HAVAL; TIGER.

I. INTRODUCTION

The advancement in mobile telecommunications and recent developments in mobile technology, make it possible to reach *any-person* at *any-where* and *any-time*. However, unlike handwritten letters, or voice messages, information in the electronic data communications through mobile communications with latest mobile devices and applications, do not provide protection of privacy or message authenticity.

Choosing a symmetric and public-key cryptography system is application dependent where (i) public-key system is widely used for protecting email and (ii) symmetric key system is used in short message applications. In spite of incorporating well developed cryptography systems (both Symmetric-key system and Asymmetric-key system) in networks, there are hackers / intruders who silently interfere and attack the entire network.

Cryptographic hashing has long played an essential role in Cyber Security. This is because both Symmetric-Key and Asymmetric-Key Cryptography use a specific key to encrypt or decrypt a message, that is reversing of plaintext from the cipher text is possible with the help of keys, whereas reversing is not possible in case of Hashing. In the cyber world, it becomes necessary to ensure authenticity of information and this can be achieved through hashing in cryptography. Later it was realized that hashing can also be a very useful building block to solve other security issues in communication and computer networks.

Protection of passwords, construction of efficient digital signature schemes, building block in protocols such as entity authentication protocols, key distribution protocols, bit commitment etc. are done by using efficient and secure hash functions that behaves as a random function implying that there is no correlation between input and output bits; and no correlation between output bits, etc.

To ensure about the originator of the message, Message Authentication Code (MAC) is used which can be implemented using either symmetric Stream cipher cryptography or Hashing cryptography techniques. However, MAC is limited with (i) Establishment of Shared Secret and (ii) Inability to Provide Non-Repudiation. Even some kind of attacks is found like Content modification, Sequence modification, Timing modification etc.

This paper is aimed to discuss the several security algorithms based on hash functions in cryptography to protect the authenticity of information. In the literature study of hash functions in cryptography, it is observed that several researches have come out with new innovations, based on the standard hashing algorithms and the types of hashing functions. Survey papers have been collected from the year ranging from 1957 through 2019 and a few of them are briefed here in this literature as below.

The paper is organized as follows. Section II concise the role of hashing in cryptography. Section III highlights a few of the renowned hashing algorithms those are in use. In section IV, various categories of hash functions are overviewed. Section V is presented with works related by implementing hashing functions. Section VI consolidates the observations and inferences from section II, III, and IV. Section VII concludes this paper with further research directions.

II. HASHING IN CRYPTOGRAPHY

(i) Background

The term “hash” means ‘chop and mix’ and hashing function is the one which chops and mixes information and derives hash results. The concept of hashing was first stated in the year 1953 after the invention of the first true electronic computer in 1950 [Luhn 1953]. In the field of computer science, for any arbitrary set of keys, a random uniform hash is the best choice.

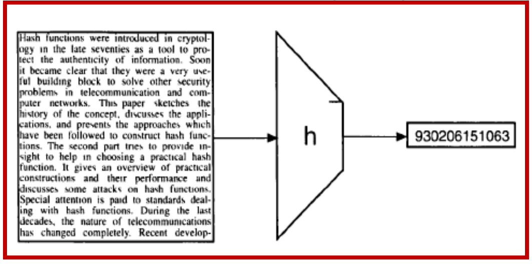
In 1957, motivated by the needs of using a random-access system with very large capacity for business applications, Wesley Peterson provided an estimation of the amount of search required for the exact location of a record in several types of storage systems, including the index-table method and the sorted-file method [Wesley, 1957]. In 1968, the word “hashing” was first used [Morris, 1968].

(ii) Definition

A hashing function is any function (·) that can be used to map an arbitrary size of data to a fixed interval [0, m]. Given a dataset containing n data points X = [x1, x2,... x_n] ∈ R^D, and a hashing function (·), the h(X) = [h(x1), h(x2) ,..., h(x_n)] ∈ [0,m] can be called the hash values or simply hashes of data points X = [x1, x2,...x_n] ∈ R^D [Alfred et al., 1996].

(iii) Hashing in Cryptography

Hashing in cryptography uses hash functions to map strings (messages) of arbitrary length to strings of a fixed and short length (message digest), ranging between 128 and 512 bits [Bart, 1994]. This hash function is called a Message Authentication Code or MAC and doesn't need. The fixed-size hash results are used for validating a original message.



Hash functions were introduced in cryptology in the late seventies as a tool to protect the authenticity of information. Soon it became clear that they were a very useful building block to solve other security problems in telecommunication and computer networks. This paper sketches the history of the concept, discusses the applications, and presents the approaches which have been followed to construct hash functions. The second part tries to provide insight to help in choosing a practical hash function. It gives an overview of practical constructions and their performance and discusses some attacks on hash functions. Special attention is paid to standards dealing with hash functions. During the last decades, the nature of telecommunications has changed completely. Recent develop-

Fig. 1 - A Hash Function

The properties of hash function include the following (i) *Compression* which means that a message of any size can be compressed to a fixed small size crypt message called Message digest. (ii) *Pre-image resistance* which means that is impossible to retrieve a message from a known hash digest. (iii) *Weak collision resistance* which means that no message produces a hash digest that is same as another message. (iv) *Strong Collision resistance* which means that no two different messages produce a same hash digests. The characteristics of a hash function are as follows (i) It should be very tough to produce the hash message. (ii) Hash function can be applied to a variable length of data. (iii) Even a small change in the message changes the hash value produced. (iv) Hash function is a one-way function that could never be inverted. (v) Hash function uses the entire data and distribute the input uniformly to the entire set. A major attack on hash function is the collision attack where an intruder could produce a message that produces a same hash value as another message and is called *birthday attack.*

III. RENOWNED HASHING ALGORITHMS

The most widely used hashing functions are (i) Message Digest (MD), (ii) Secure Hash Algorithm (SHA), (iii) RACE Integrity Primitives Evaluation Message Digest (RIPEMD), (iv) Whirlpool, (v) BLAKE, (vi) Cuckoo, (vii) JH, (viii) HAVAL, and (ix) TIGER. A study is carried out on the widely used hashing functions and glimpse of them are presented in this section.

(i). Message Digest (MD)

The Message Digest family encompasses hash functions such as MD2, MD4, MD5 and MD6 [Rivest, 1992].

MD5 is a 128-bit hash function accepted as Internet Standard RFC 1321 and is used widely for its strength in providing integrity of the message transferred. *However, in 2004, collisions were found in MD5* [Stevens et al., 2009] *and hence it is no longer recommended for use. Even though extensive vulnerabilities found with MD5, it is used as a checksum to verify data integrity, but only against unintentional corruption.*

(ii). Secure Hash Algorithm (SHA)

The Secure Hash Algorithm (SHA) family comprises of four SHA algorithms; SHA-0, SHA-1, SHA-2, and SHA-3 [Hernandez et al., 2015]. Though same family each algorithm is structurally different.

SHA-0 is a 160-bit hash function published by the National Institute of Standards and Technology (NIST) in 1993 but did not become popular because of few weaknesses in it. Later in 1995, SHA-1 was designed to correct alleged weaknesses of SHA-0 and employed in several widely used applications and protocols including Secure Socket Layer (SSL) security. However long-term employability of SHA-1 is doubtful. SHA-2 family has four further SHA variants, SHA-224, SHA-256, SHA-384, and SHA-512 where 224, 256, 384 and 512 are the number of bits in their hash value. No successful attacks have been reported on SHA-2 hash function and employed as a strong hash function. However, NIST called for new competitive hash function designs and in October 2012, the NIST chose the Keccak algorithm as the new SHA-3 standard. Keccak offers many benefits, such as efficient performance and good resistance for attacks.

(iii). RACE Integrity Primitives Evaluation Message Digest (*RIPEMD)*

The RACE Integrity Primitives Evaluation Message Digest family generally known as a family of European hash functions includes RIPEMD, RIPEMD-128, RIPEMD-160, RIPEMD-256, and RIPEMD-320-bit [Bosselaers, 2005].

RIPEMD 128-bit has overcome vulnerabilities on the original RIPEMD. RIPEMD-160 is widely used version in the family. Even though RIPEMD-256 and RIPEMD-320 have decreased the accidental collisions, they do not have higher levels of security as compared to RIPEMD-128 and RIPEMD-160 respectively.

(iv). Whirlpool

The Whirlpool family consists of WHIRLPOOL-0, WHIRLPOOL-T and WHIRLPOOL [McLoone et al., 2005].

A refined form of Advanced Encryption Standard (AES) is Whirlpool, a 512-bit hash function was chosen as part of the New European Schemes for Signatures, Integrity and Encryption (NESSIE) in February 2003 and has been included in the ISO/IEC 10118-3 standard. Whirlpool is one of the unbroken methods for constructing a hash function from a block cipher and is provably secure.

(v). BLAKE

The BLAKE family consists of BLAKE, BLAKE2, BLAKE2b and BLAKE2s [Aumasson *et al.*, 2013].

BLAKE did not win in SHA-3 competition. Then BLAKE2 has shown higher performance than BLAKE. Reduction in the number of rounds of compression from 16 to 12, BLAKE2b has come-up. Then BLAKE2s has turned up by reducing the number of initialization words from 24 to 8. BLAKE2 implements tree hashing for incremental update, verification of large files and minimal padding for messages with fast computation. *However, attacks are found in BLAKE* [Ji et al., 2009].

(vi). CUCKOO

Cuckoo, a dynamization of a static dictionary, uses two hash tables. Insertion and deletion is used to move the keys either to the table or from the table [Devroye *et al.*, 2003]. Two way hashing method are used to two instances of Chained hashing. *However, If the chain is long re-hash is needed. Finally the large amount of storage space is needed for dynamic perfect hashing.*

(vii). JH

JH hash family includes four types JH-224, JH-256, JH-384 and JH-512 using compression function. This family is introduced in 2008 to ensure high collision resistance and security because of its two features [George et al., 2013].

This function consists of five steps, which are: Padding the initial message, M, passing the padded message into message blocks, setting the initial hash value, H(0), computing the hash value H (N) and finally generating the message digest by truncating H (N).

(viii). HAVAL

HAVAL, a one-way hashing algorithm can produce output message digests with different sizes of 128, 160, 192, 224 and 256-bits [Sklavos et al., 2005]. *However, two messages collide each other when they are compressed to the same message digest. Therefore, HAVAL could not be formally proved to be secure. HAVAL operation defines at least three passes.*

(ix). TIGER

Tiger, a cryptographic hash function, proposed by Anderson and Biham in 1996, processes 512-bit blocks and produces a 192-bit hash value [Mendel et al., 2007]. Block-cipher-based compression function having key schedule block and state update transformation block, with 8-bit input produces 64-bit output and provides faster diffusion in comparison with integer arithmetic. The TIGER is prone to Man-in-the-Middle attack.

IV. TYPES OF HASH FUNCTIONS

The literature on the hash functions are also carried out on the types of Hashing. The types of hashing are classified into three major categories as (i) One-way hash function (OWHF), defined in [Merkle, 1979], [Rabin, 1978]; (ii) Collision resistant hash function (CRHF) defined in [Damgird, 1988], and (iii) Message Authentication Code (MAC) defined in [Merkle, 1990].

Based on the application OWHF or CRHF is used. MAC is used with OWHF and CRHF and is used for authentication of multi-destination messages [Mitchell, 1989].

V. LITERATURE SURVEY OF RELATED WORKS ON HASHING ALGORITHM

Based on the above three major types of hash functions, several authors have researched. As a result, many hash functions have been introduced and some of the noteworthy and recent works are briefed in sequence.

Universal One-way Hash Function (UOWHF) is used to build a signature scheme [Naor et al., 1990]. It is sufficient to have a UOWHF that compresses to a single bit from an arbitrary number of bits. Several authors have subsequently improved their construction. *A key result by J. Rompe1 is a (very inefficient) construction for a UOWHF based on any one-way function, which is the weakest possible assumption* [Rompel, 1990].

Kaliski a chief scientist from RSA Laboratories performed a survey that focuses on encryption algorithms, the low-level, step-by-step transformations on messages that address the problems of cryptographic algorithms, as well as applications that involve encryption [Kaliski, 1993]. The survey covers approved standards and work in progress. *The modifiers draft and the proposed work may contribute to some important step for an innovation thus proving that the recent cryptographic algorithms are yet to be more secure.*

Knudsen et al., presented a hash function named SMASH, which has a hash code of 256 bits and is as fast as SHA-256 [Knudsen *et al.*, 2005]. The author considered compression functions is designed from one of fixed bijective mapping f: $\{0, 1\}^n \rightarrow \{0, 1\}^n$. *As the compression functions are weak the resulting hash functions are not easy to break.*

Eli *et al.* proposed HAsh Iterative FrAmework (HAIFA) as a replacement for the Merkle-Damgard construction to support the properties of hash functions and variable hash size [Eli *et al.*, 2006]. *HAIFA performs either offline or online computation of the hash function in one pass with a fixed quantity of memory autonomously of the size of the message. If the message size is large this method becomes absurd.*

Dahmen *et al.* proposed an algorithm for Merkle authentication trees which does not need collision resistant hash functions [Dahmen *et al.*, 2008]. The signature scheme obtained from this algorithm is protected against adaptive chosen message attacks, When the underlying hash function is second pre-image resistant this signature scheme is un-forgeable, as it produces signatures that are shorter in size, and not affected by birthday attacks and *are found by recent collision algorithms too.*

Mustafa *et al.* proposed a Genetic Algorithm to efficiently hash a given set of keys which minimize the number of collisions. The proposed algorithm created Universal Hash function for hashing a given set of keys which results in the least possible number of collisions [Mustafa *et al.*, 2009]. This algorithm is suitable for the situations where the input distribution to be hashed is dynamic and where the hash function is to be changed dynamically in order to rehash the input. *However, this algorithm involved little computational overhead and could be implemented in any areas of requirements. The sieve algorithms and an improvement in the encoding of the chromosomes will make this algorithm more suitable to almost all the situations by avoiding some few case study flaws.*

Philippe *et al.* introduced SipHash, a family of hash functions to attend to the requirements for high-security short-input MACs used for network traffic authentication and hash-table lookups [Philippe *et al.*, 2012]. SipHash is simpler than MACs based on universal hashing. *The authors themselves have crypt analyzed and found that the keys the states of this hashing could be recognized but in a slower fashion when compared to other hash functions.*

Jinse *et al.* investigated the feasibility of content-based authentication using image hashing technique. The authors considered digital water marking-based approach, an alternative to the traditional Wireless Multimedia Sensor Networks (WMSNs) techniques [Jinse *et al.*, 2013]. Moreover, the author measured the performance of five selected image hashing algorithms with respect to robustness, sensitivity, and security. *But the content-based image authentication is not a hard authentication solution using classical cryptography; however some potential attacks can still exist by exploiting the strong statistical and perceptual redundancy on the image content.*

Ke Yan et al. designed DM-HASH algorithm, a dual flow matching algorithm based on hash functions in the aspect of space and time performance [Ke Yan *et al. 2013*]. The author enhanced XOR hashing algorithm to improve the randomness of the key value and then mapped the key value to a memory address. *This algorithm does not eliminate the collision, so resolving the conflict of a hashing algorithm is a problem and must be solved.*

Nasir *et al.* introduced Genetic Hash Algorithm (GHA) by merging Hash Visualization Technique with Genetic Algorithms, to improve hash visualization [Nasir *et al.*, 2014]. Hash visualization technique generates random images using a tree in which the nodes of the tree are randomly assigned mathematical operations and the leaves are assigned random values. The tree is evaluated and the resulting value is assigned to a pixel. Genetic algorithm is an optimization technique based on chromosome of living creatures which has a fitness value. In this Genetic Hash Algorithm, the trees are generated to obtain a better and larger tree that will generate images with higher security. *However, as the collisions depend on the nature and design of hash functions, the resulted number of collisions has not been considerably changed.*

Bernstein *et al.* introduced SPHINCS, to perform randomized tree-based stateless signatures, which is a high-security post-quantum stateless hash-based signature scheme that signs hundreds of messages per second [Bernstein et al., 2015]. The signature scheme is designed to provide long-term security even against attackers having with quantum computers.

Shay *et al.* introduced SPHINCS-Simpira, a modified version of SPHINCS signature scheme using Simpira as building block with post-quantum security [Shay *et al.*, 2017]. Simpira provides a set of high throughput cryptographic permutation with different range of input sizes. Still Simpira can create 1.5x speed up for generating pairs of keys. Simpira has proved a same security level in both pre-quantum and post-quantum period.

Sevilay *et al.* introduced colour hidden hash algorithm, to transform data into binary value and then matched RGBA (Red, Green, Black, Alpha) colour value where each colour code value is related with hash code function [Sevilay *et al.*, 2019]. This algorithm initially converted the input data to byte array then it transformed into four-bit groups from this array and generated a message digest length according to the system needs and requirements. Moreover, different hash code is generated as private with unique additional input value and shared between each communicated systems. *But it is impossible to get clear text from hash value.*

VI. OBSERVATIONS AND INFERENCES

After a thorough review of the plenty of hash functions like MD, SHA, BLAKE, Whirlpool, RIPEMD, HAIFA, SMASH, Simpira SPHINCS, it is clearly understood that almost all hash function algorithms are prone to birthday attacks, dictionary attacks etc. and are prone to attacks by quantum-computers. SPHINCS-Simpira and SPHINICS are able to withstand to some extent, the fast performing quantum-computerized attacks. Most algorithms require some overhead for the stand of the algorithm in the quantum-computer world. It is left to the researcher to design some new hash function that could escape the attacker equipped with a quantum computer.

VII. CONCLUSION

Hash functions also similar to asymmetric encryption algorithm is capable to withstand the adversities of post-quantum cryptography if the processes of hashing is chosen in such a way that the keys and the states of the hash function is not easily calculated by the super computer. As seen from the review in this paper, only a few hash functions are noteworthy and is also with some drawbacks to be implemented every hash function if very good employs some form of overhead. The new researchers can try to find some methods of hash function that is also devoid of collision and is free from birthday attack so that the digital signature and secret keys are still safe in the communication channel.

ACKNOWLEDGEMENT

The authors sincerely express their special thanks and sincere gratitude to Tamilnadu State Council for Higher Education (TNSCHE) and Department of Science and Technology (DST), India, for sponsoring this research works. We would also like to thank Dr.S.Albert Rabara, Dr. Mani, Mr.Arun Gnanaraj, Mrs. Christy Sujatha, Mrs.M.Manimozhi as well as other correspondents for productive discussions and improvements of early drafts of this paper, and for pointers to the literature.

REFERENCES

[1] [Wesley, 1957], W. Wesley Peterson. 1957. Addressing for random-access storage. IBM Journal of Research and Development 1, 2 (1957), 130–146.

[2] Robert Morris. 1968. Scatter storage techniques. Communications of the ACM 11, 1 (1968), 38–44

[3] [Rabin, 1978], M. 0. Rabin: Digitalized signatures. in foundations of secure computation, (R. Lipton, R. DeMillo, Eds.), Academic Press, New York, 1978, p. 155-166.

[4] [Merkle, 1979], R. Merkle, "Secrecy. aurlierrrrciition and public key systems", UMI Research Press, 1979.

[5] [Damgird, 1988], I. B. Damgird: Collision free hash functions and public key signature schemes. Advances in Cryptology, Proc. Eurocrypt'87, LNCS 304, (D. Chaum, W. L. Price, Eds.), Springer-Verlag, 1988, PP:203-216.

[6] [Mitchell, 1989], C. Mitchell: Multi-destination secure electronic mail. "The Computer Journal", Vol. 32, No. I, PP: 13- 15, 1989.

[7] [Merkle, 1990], R. Merkle: One way hash functions and DES. Advances in Cryptology, Proc. Crypto'89, LNCS 435, (G. Brassard, Ed.), Springer-Verlag. 1990, p. 428-446.

[8] [Naor et al., 1990], M. Naor, M. Yung: Universal one-wuy hashfunctions and their cnptngrophic applications. Proc. 2 I st ACM Symposium on the Thcory of Computing, PP:387-394, 1990.

[9] [Rompel, 1990], J. Rompel: One-way functions are necessary and suscient for secure vignatures. Proc. 22nd ACM Symposium on the Theory of Computing, 1990, p. 387-394.

[10] [Rivest, 1992], R. Rivest, "The MD5 Message-Digest Algorithm", Internet Engineering Task Force, RFC 1321, April 1992.

[11] [Kaliski, 1993], B. Kaliski, "A survey of encryption standards", in IEEE Micro, ISSN: 0272-1732, vol. 13, no. 6, pp. 74-81, Dec. 1993, doi: 10.1109/40.248057.

[12] [Preneel, 1994], Preneel Bart, "Cryptographic hash functions", European Transactions on Telecommunications 5.4, PP: 431-448, 1994.

[13] [Alfred et al., 1996], Menezes, Alfred J.; van Oorschot, Paul C.; Vanstone, Scott A, "Handbook of Applied Cryptography", CRC Press, ISBN 978-0849385230, 1996.

[14] [Devroye *et al*., 2003], L. Devroye, P. Morin, “Cuckoo Hashing: Further Analysis”, In the Proceedings of Information Processing Letters, Vol.86, No.4, PP:215-219, 2003.

[15] [McLoone *et al.*, 2005], Máire McLoone, Ciaran McIvor, Aidan Savage, “High Speed Hardware Architectures of the Whirlpool Hash Function”, In the Proceedings of the International Conference on Field Programmable Technology, 2005.

[16] [Sklavos *et al.*, 2005], N. Sklavos, C. Efstathiou, “On the FPGA Implementation of HAVAL Hash Function”, In the IEEE Proceedings of the International Conference on Computer as a Tool (EUROCON'05), Belgrade, Serbia & Montenegro, November 21-24, 2005.

[17] [Knudsen et al., 2005], Lars R. Knudsen, "SMASH – a cryptographic hash function", In the Proceedings of the 12th international conference on Fast Software Encryption, ISBN: 978-3-540-26541-2, PP:228–242, Feb.2005, https://doi.org/10.1007/11502760_15

[18] [Bosselaers, 2005], Bosselaers A, "RIPEMD Family", In Encyclopedia of Cryptography and Security, Springer, Boston, MA, 2005, https://doi.org/10.1007/0-387-23483-7_360

[19] [Eli et al., 2006], Eli Biham, Orr Dunkelman, "A Framework for Iterative Hash Functions: HAIFA", In Proceedings of Second NIST Cryptographic Hash Workshop, 2006. Available from: www.csrc.nist.gov/pki/HashWorkshop/2006/program_2006.htm

[20] [Mendel *et al.*, 2007], Florian Mendel, Vincent Rijmen, “Cryptanalysis of the Tiger Hash Function”, In the Proceedings of the International Association for Cryptologic Research (ASIACRYPT), PP:536–550, 2007.

[21] [Dahmen *et al*., 2008], Buchmann J., Dahmen E., Schneider M., “Merkle Tree Traversal Revisited”, In the Proceedings of the International Workshop on Post-Quantum Cryptography (PQCrypto-2008), Lecture Notes in Computer Science, E-ISBN:978-3-540-88402-6, Vol.5299, PP:63-78, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-88403-3_5

[22] [Stevens *et al.*, 2009], Stevens M, Sotirov A, Appelbaum J, Lenstra A, Molnar D, Osvik D, Weger B, “Short chosen-prefix collisions for md5 and the creation of a rogue ca certificate”, In Proceedings of the International Conference on Advances in Cryptology (CRYPTO 2009), Springer Lecture Notes in Computer Science, Vol. 5677, PP:55–69, Springer Berlin/Heidelberg, 2009.

[23] [Ji et al., 2009], Li Ji, Xu Liangyu, "Attacks on Round-Reduced BLAKE", https://eprint.iacr.org/2009/238.pdf, Last retrieved 19.Oct.2020.

[24] [Mustafa *et al., 2009]*, Mustafa Safdari, Ramprasad Joshi, "Evolving Universal Hash Functions using Genetic Algorithms", In the Proceedings of the IEEE International Conference on Future Computer and Communication, ISBN: 978-0-7695-3591-3, PP:84-87, 2009. DOI 10.1109/ICFCC.2009.66

[25] [Philippe *et al.*, 2012], Jean-Philippe Aumasson, Daniel J. Bernstein, "SipHash: a fast short-input PRF", In the Proceedings of the 13th International Conference on Cryptology, ISBN: 978-3-642-34930-0, PP:489-508, 2012, https://eprint.iacr.org/2012/351.pdf

[26] [Aumasson *et al.*, 2013], Aumasson, Jean-Philippe, et al.,"BLAKE2: Simpler, smaller, fast as MD5", In the Proceedings of the International Conference on Applied Cryptography and Network Security (ACNS 2013), Springer Berlin Heidelberg, Online ISBN:978-3-642-38980-1, PP:119-135, 2013, DOI: https://doi.org/10.1007/978-3-642-38980-1_8 2013.

[27] [George *et al.*, 2013], George S. Athanasiou, Harris E. Michail, George Theodoridis, Costas E. Goutis, “High-performance FPGA implementations of the cryptographic hash function JH”, In the Journal of IET Computers & Digital Techniques, ISSN 1751-8601, Vol. 7, Issue:1, PP:29–40, 2013.

[28] [Jinse et al., 2013], Jinse Shin, Christoph Ruland, "A Survey of Image Hashing Technique for Data Authentication in WMSNs", In the Proceedings of the International Workshop on IEEE International Conference on Internet of Things Communications and Technologies (IoT'13), ISBN:978-1-4577-2014-7, PP: 253-258, 2013.

[29] [Ke Yan *et al. 2013*], Ke Yan, Jian Chen, Bingyao Cao, Yue Zheng, Tao Hong, "Research on a low conflict flow matching hash algorithm," In the Proceedings of *IET International Conference on Smart and Sustainable City 2013 (ICSSC 2013)*, PP:234-237, ISBN:978-1-84919-707-6, Shanghai, 2013, DOI: 10.1049/cp.2013.2017.

[30] [Nasir et al., 2014], L. M. H. Nasir Eddeen, E. M. Saleh, D. Saadah, "Genetic Hash Algorithm", In the Proceedings of the 6th International Conference on Computer Science and Information Technology (CSIT), Amman, PP:23-26, 2014, DOI: 10.1109/CSIT.2014.6805974.

[31] [Hernandez et al., 2015], Hernandez, Paul, "NIST Releases SHA-3 Cryptographic Hash Standard", NIST, 2015.

[32] [Bernstein*et al.*, 2015], Daniel J. Bernstein, Daira Hopwood, Andreas Hülsing, Tanja Lange, Ruben Niederhagen, Louiza Papachristodoulou, Michael Schneider, Peter Schwabe, Zooko Wilcox-O'Hearn, "SPHINCS: Practical Stateless Hash-Based Signatures", In the Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT-2015), @ Springer LNCS, EISBN:978-3-662-46800-5, Vol.9056, PP368--397. https://eprint.iacr.org/2014/795.

[33] [Lianhua et al., 2017], Lianhua Chi, Xingquan Zhu, "Hashing Techniques: A Survey and Taxonomy", ACM Computing Surveys, ISSN: 0360-0300/2017/04, Vol. 50, No. 1, Article 11, PP:11-35, Publication date: April 2017. DOI: http://dx.doi.org/10.1145/3047307

[34] [Shay *et al.*, 2017], Shay Gueron, Nicky Mouha, "SPHINCS-Simpira: Fast Stateless Hash-based Signatures with Post-quantum Security", White Paper, 29 Jun.2017, https://eprint.iacr.org/2017/645.pdf https://csrc.nist.gov/publications/detail/white-paper/2017/06/29/sphincs-simpira/final,

[35] [Sevilay et al., 2019], Sevilay Velioğlu, Doruk Kaan Bolu, Eren Yemen, "A New Approach to Cryptographic Hashing: Color Hidden Hash Algorithm", In the Proceedings of the International Conference on Digitization (ICD), ISBN:978-1-7281-3841-1, PP:170-173, 2019, DOI:10.1109/ICD47981.2019.9105898.

[36] [Priyansh et al., 2020] Priyansh Kumar Dubey, Ajay Jangid, B. R. Chandavarkar, “An Interdependency between Symmetric Ciphers and Hash Functions: A Survey”, In Proceedings of *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, ISBN: 978-1-7281-6852-4, PP: 1-5, 2020, DOI: 10.1109/ICCCNT49239.2020.9225412