Brief Review on Journey of Secured Hash Algorithms

Prof. Santanu Debnath[1], Dr. Abir Chattopadhyay[2], Subhamoy Dutta[3]
[1]C.S.E. Dept, Camellia School of Engineering & Technology, profsantanu@gmail.com,
[2]University of Engineering & Management, Kolkata, abir.uem@gmail.com

[3]University of Engineering & Management, Kolkata WB, India,
subhamoydutta1995@gmail.com

***Abstract*— A detailed review of a brief history, applications and contributions of the the hash functions in today's cryptology are articulated here. The secured hash functions are applied in various fields to provide a secure data transfer and authentication of messages and other user linked information through a series of algorithms. From the establishment of the first hash function MD5, followed by SHA 1, this data encryption system has undergone several upgradations and advanced to SHA 2 and SHA 3, the details of which are discussed in this paper. The importance of the secure hash algorithm in network security and also the necessity to upgrade from SHA1 and SHA2 to the modern standards of SHA3 is also highlighted here.**

Keywords—Secured Hash Algorithm; SHA1; SHA2; SHA3; MD5; Keccak;Cryptology; Network Security; NIST.

I. INTRODUCTION

Network Security ensures securing a computer network and network framework to block unauthorized access, data manipulation, device and data modification. As the internet evolves and computer networks expand, network security has become an important factors for organizations to consider. Computers today control large money transfers between banks, markets, health and medical fields, telecommunication, electrical power distribution, nuclear power plants, defense, satellites and space research. Network security cannot be compromised in these crucial fields. With increasing complexities of the network systems, they are turning vulnerable to the attacks of hackers. Therefore Network Security ensures that our private or confidential data are not prone to any type of misuse. The network computers are linked to it, so the computer security from attackers also belongs network security. It is thus an essential part of today's world.[1]

II. CRYPTOGRAPHY

Cryptography, or cryptology, is the practice and detailed study of masking data and messages. More precisely it is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Modern cryptography combines mathematics, computer science, and electrical engineering. A message transfer through cryptography involves data encryption which encodes the data, making it obscure. The data can only be decrypted using a "key" which hides the secret messages. Procedures such as microdots, merging of words and images, and various ways to cloak information in storage or passage are used in cryptography. Present-day cryptography involves the following objectives: **Confidentiality** (the message cannot be cracked by anyone except the predetermined receiver). **Non-repudiation** (the sender of the message cannot deny at a later stage his or her purpose in the transmission of the information). **Integrity** (Information cannot be modified without being detected). **Authentication** (the sender and intended receiver can confirm each other’s identity and the origin of the information).[1]

III. BRIEF TIMELINE OF CRYPTOGRAPHY

The term “crypto” is derived from the Greek “kryptos”, meaning hidden. The origin of cryptography dates back to 1900 BC, with the Egyptian practice of hieroglyphics. The first known use of a modern cipher was by Julius Caesar (100 BC to 44 BC), in which each character was replaced by a character three positions prior to it.[2] During 800, Al-Kindi, regarded as the "The Philosopher of the Arabs" developed a method where a cipher could be cracked by analyzing the frequency of occurrence of letters by frequency analysis. In 1467, Leon Battista Alberti, accomplished great success by inventing the Cipher Wheel that incorporates the first example of polyalphabetic substitution with mixed alphabets and variable period. The circular approach of the wheel made encryption and decryption much faster.[2] In 1917, Gilbert Vernam invented The Vernam Cipher, a teleprinter cipher, which used a key on paper tape to be added with regular words to encrypt phrases. Praised by the NSA as one of the greatest inventions in cryptology, the Vernam Cipher is known as the world's first unbreakable cipher. In 1975, The National Institute of Standards and Technology (NIST) developed an encryption that could be used by all US agencies. They called this the Data Encryption Standard (DES) which was a symmetric-key block cipher. The Advanced Encryption Standards (AES) was published by NIST in 2001, which greatly focused on enhancing secure encryption that is still in use. Secure Hash Algorithm (SHA), a family of cryptographic hash functions, was published in 1993, titled Secure Hash Standard, FIPS PUB 180, by U.S. government agency NIST.[3] It converts a data string into a numeric string output of fixed length. The output string is generally smaller than the original data. This version is now named SHA-0, followed by SHA-1 in 1995, SHA-2 in 2001, and SHA-3 in 2015.

IV. SECURE HASH ALGORITHM

Owing to the scarcity of scaling down of CMOS technology, study of new devices principles are utmost necessary. It is one type of hash function that has special cryptographic attributes, making it appropriate for cryptography. It is a mathematical algorithm that plots data of random size to a bit string of a fixed size hash function which is a one-way function, that cannot be inverted. The input data can only be decrypted by brute-force search of possible inputs to see if they produce a match, or use a rainbow table of matched hashes. The ideal cryptographic hash function has five main properties:

- It is a deterministic algorithm which ensure exact hash for the same message
- Hash value determination is very fast
- A hash value cannot be decoded without trying all possible combinations
- A minor change in the hash value will generate a completely different output
- Two different messages will never have the same hash value.

Cryptographic hash functions have numerous information-security applications in digital signatures, message authentication codes (MACs), and other forms of authentication. They are also used as ordinary hash functions, to index data in hash tables, for fingerprinting, to detect duplicity or uniquely identify files, and as checksums to detect accidental data corruption. Indeed, in information-security contexts, cryptographic hash values are sometimes called digital fingerprints, checksums, or hash values.

V. MD5

The “Message Digest 5” or commonly known as MD5 is widely used 128-bit hash function which was first designed in 1992 by Professor Ronald Rivest at MIT. Operations in MD5 are done by padding messages at first so as to adjust its length to 448 mod 512 thereby appending a 64-bit length value to the message, followed by the initialization of 4-word (128-bit) MD buffer (A,B,C,D). It then processes message in 16-word (512-bit) blocks using 4 rounds of 16 bit operations on the message block and buffer then adds output to buffer input to form new buffer value. The Output hash value becomes final buffer value.[1]

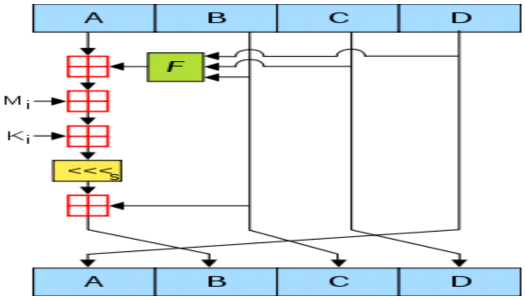

Fig.1. MD5 Operation

MD5 has helped the electronic sector by providing exclusive identifier for every data exchanged during legal discovery process, which replaced the method of Bates stamp numbering system. In 2009, an MD5 hash value was used by the United States Cyber Command as a part their official emblem. As this hash function was not related to any encryption, its security was compromised as a result of several brute-force attacks, prominently by the Flame malware in 2012. Also, the small size of the hash function makes it prone to 'birthday attack'. Such demonstration was shown in 2004 under the project name MD5CRK. Due to easy generations of collisions, two different messages may have the same checksum, which makes it victim to malicious attacks. Due to these drawbacks, NIST has excluded MD5 for password storages. Latest statistics suggests that MD5 was employed to crack 98% of the 699,494 passwords from the breach of DaFont's user accounts.[1]

VI. SHA-1

The SHA-1 algorithm was first published in 1995 in FIPS PUB 180-1 under Capstone project of the U.S. Government. SHA-1 differs by a single bitwise rotation in the message schedule from its predecessors. In SHA-1, message length is padded to 448 mod 512, following which a 64-bit length value is added to it. After that, initialization of 5-word (160-bit) buffer (A,B,C,D,E) is done to process the message in 16-word (512-bit) blocks using 4 rounds of 20 bit operations on message block and buffer, then the output is added to buffer input which yields a new buffer value, and this is regarded as the output hash value.[1][6]

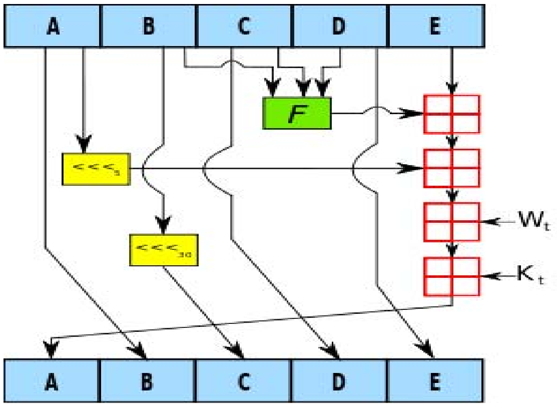

Fig.2. SHA-1 Operation

Compared to MD5, SHA-1 is slower in terms of speed but unlike 128-bits MD5, SHA-1 (160-bits) is not vulnerable to brute force attacks and it is optimized for big endian CPUs. Some prominent cryptographic protocols where SHA is used are TLS, SSL, SSH, PGP, S/MIME and IPsec. SHA-1 has proven its worth in identifying revisions and detection corrupt or illegally modified data in distributed revision control systems like Mercurial, Monotone and Git. SHA-1 was used in verification of signature in Nintendo's Wii gaming console during booting process.

The Problems with SHA-1 : Due to a number of security flaws in MD5, SHA-1 was recommended over it. However, SHA-1 is currently outdated in recent years and the increasing number of collisions has forced it to be replaced by SHA-2. NIST has already banned the use of SHA-1 effective 31st December, 2013 which suggests the risk factors involved regarding the use of SHA-1. The certificate authorities, antivirus companies, web browsers, and other entities are working together to upgrade their digital certificates and eliminate the use of SHA-1 in the upcoming years.[6]

VII. SHA-2

The SHA-2 family of cryptographic hash functions was first designed in 2001 by United States NSA and is patented under US patent 6829355. SHA-2 is an improved version of algorithm compared to the previous MD-5 or SHA-1. The SHA-2 set of algorithms consists of six hash functions with hash values of 224, 256, 384 or 512 bits, acknowledged as SHA-224, SHA-256, SHA-384, SHA-512/224, SHA-512/256.[1] The SHA-256 with 32-bits and SHA-512 with 64-bit are widely used hash functions. Although both of these hash functions have virtually identical basic structures but they differ in use of shift amounts, additive constants, and number of rounds. The generation of initial values using SHA-512/224 and SHA-512/256 are done according to the procedures described in Federal Information Processing Standards PUB 180-4. It is designed to function with enhanced security provided by the AES cipher. According to a report published in 2017, it was no longer recommended to use SHA-1 in applications that depend on collision resistance, such as digital signatures, as it was more prone to collisions than intended. But SHA-2 remained unbreakable against these attacks.[5][7]

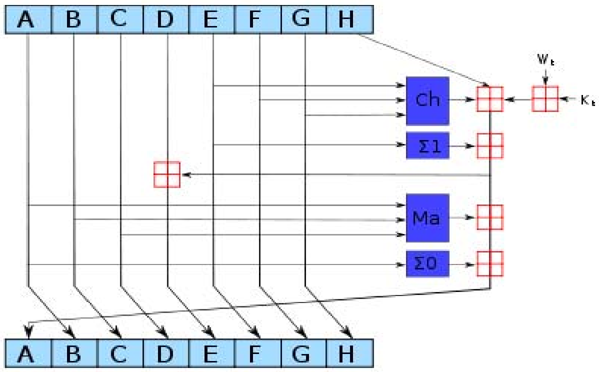

Fig.3. SHA-2 Operation

Applications : SHA-2 hash functions are widely implemented in security applications and protocols such as SSL, TSL, PGP, S/MIME, SSH and IPsec. SHA-256 is used in DKIM message signing standard and authenticating Debian software packages. SHA-512 was used to authenticate a video from International Criminal Tribunal of the Rwandan genocide. SHA-256 and SHA-512 are recommended to be used in DNSSEC, and are also used for secure password hashing in Unix and Linux. SHA-256 is used for verifying transactions and calculating proof-of-stake in several crypto-currencies like Bitcoin. SHA-2 is extensively used in cryptographic algorithms and protocols, and for protection of sensitive unclassified data by the U.S. Government.[1][7]

VIII. SHA-3

The Secure Hash Algorithm 3 (SHA-3) was first designed by Guido Bertoni, Michaël Peeters, Joan Daemen, and Gilles Van Assche. SHA-3 was never meant to replace SHA-2 at its earliest but because of successful collisions and attacks on MD-5, SHA-1, NIST anticipated the need for a dissimilar cryptographic hash function, which developed into SHA-3.[1][8] The Keccak hash family is revised version of algorithm in the SHA-3 family. On October, 2012, Keccak developed into a fully functional hash standard. In 2014, NIST published a draft FIPS 202 "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions". FIPS 202 was approved on 5th August, 2015, where it was announced by NIST that SHA-3 was designated as a new hash standard. The design depicted in Fig. 4. Shows the "Sponge Construction" of SHA-3. The design of SHA-3 uses a sponge construction where the data is absorbed into the hash function and then the result is squeezed out. In absorbing phase, data blocks are XOR-ed into a subset of the state, which then transformed as a whole. In squeeze phase, output blocks are read from the same subset of the state, alternated with state transformations. We often denote r ("rate") part as the state of read and write and c ("capacity") part as input/output which is untouched and confirms the security system where capacity is half of the security level. Every state in SHA-3 contains array of 5 X 5 of 64-bit words or total of 1600 bits. Cryptanalytic attacks can be tested by an intermediate small state sizes between 200 bits and 800 bits for lightweight practical applications. NOT, AND & XOR operations are used to permute the transformation of each block and shows comparatively faster and easier implementations of hardware and software. On certain architectures Keccak's authors tried to show authenticated encryption technology and "tree" hashing algorithm for speedy hashing system.

IX. CONCLUSION

The Keccak design of algorithm from the SHA 3 family was adopted by NIST after numerous number of complex evaluations and simulations, where it proved to be much more advanced and efficient as compared to its competitors. Although the existing SHA-2 was very strong in terms of performance and data security, which the new SHA-3 still had to uncover for better implementation. But SHA-3 had a different architectural design which provided it with outstanding accomplishments in the areas where SHA-2 could not be utilized. Keccak uses the 'sponge construction' domain extender that works on fixed permutation by adjusting exchange specific security assets for enhanced productivity, generating variable range of outputs.[4] This extends Keccak's reach over large security domain, higher efficiency in hardware usage, generic performance. Keccak is equipped with improved chaining mode that boosts genuine encryption. NIST is currently aiming to expand the present hash standard, FIPS 180-4, and include the SHA-3 algorithm by publishing a draft FIPS 180-5 for general evaluation. NIST is in consultation with the Keccak designer team and cryptographic research community for standardizing further construction based Keccak permutations for authenticated encryption mode in coming years.[9][10]

REFERENCES

[1] SHA hash functions - Wikipedia, the free encyclopaedia [online], Available: http:// en.wikipedia.org/wiki/SHA1, http://en.wiki pedia.org/wiki/SHA2, http://en.wiki pedia. org/ wiki/ SHA3

[2] The History of Cryptography - http://www.timetoast.com/timelines/the-history-of-cryptography

[3] Compact Implementation of SHA3-1024 on FPGA by S.Bhargav, Dr. Drva Sharath Kumar - http://www.ijeert.org/pdf/v3-i7/11.pdf

[4] Secure Hash Standard(SHS), FIPS PUB 180-4, FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, March 2012, http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf

[5] International Journal of Emerging Engineering Research and Technology Volume 3, Issue 7, July 2015, PP 79-86 ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

[6] SHA-1 - https://en.wikipedia.org/wiki/SHA-1

[7] Verification of a Cryptographic Primitive: SHA-256 ANDREW W. APPEL, Princeton University

[8] Cryptographic Standards and Guidelines, NIST, https://csrc.nist.gov/projects/cryptographic-standards-and-guidelines/example-values#aHashing

[9] SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions - https://csrc.nist.gov/csrc/media/publications/fips/202/final/documents/fips_202_draft.pdf

[10] Keccak Implementation Overview, Guido Bertoni, Joan Daemen, Michael Peeters, Gilles Van Assche, Rony Van Keer - https://keccak.team/files/Keccak-implementation-3.2.pdf

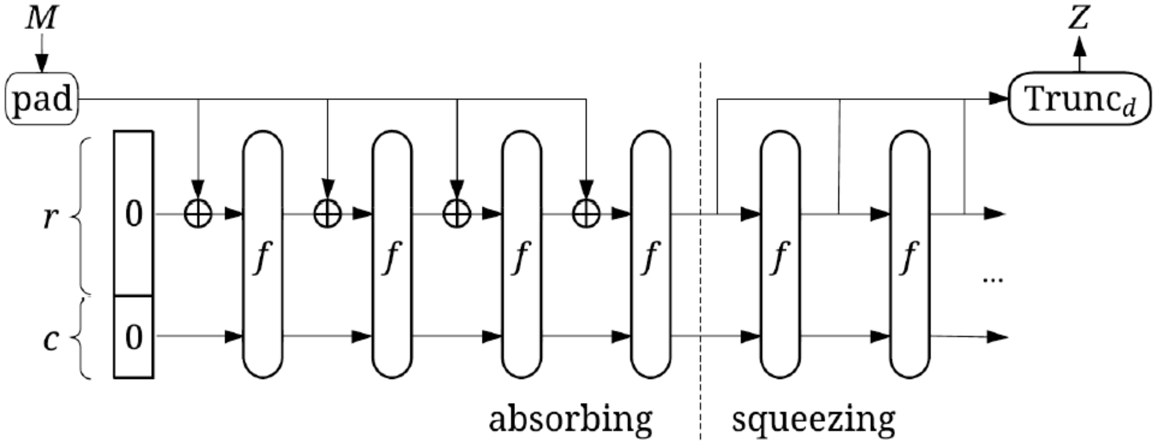

Fig.4. Sponge Construction design of SHA-3[9]