Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.199

IJCSMC, Vol. 9, Issue. 2, February 2020, pg.111 – 116

A Robust Review of SHA: Featuring Coherent Characteristics

Ghazala Shaheen
King Khalid University, KSÁ
Email: ghazalghazi@hotmail.com; gshaheen@kku.edu.sa

ABSTRACT: Secure hashing algorithms produces a message digest based on principles similar to those used in the design of the MD4 and MD5 message digest algorithms, but has a more conservative design. The four SHA algorithms are structured differently and are named SHA-0, SHA1, SHA-2, and SHA-3. SHA series appears to provide greater resistance to attacks, supporting the NSA's assertion that the change increased the security. This is a review study which includes the comparisons of different secure hashing algorithms with respect to the attributes that are considered as performance pillars of cyber security systems.

Keywords-Secure Hashing Algorithms (SHA-0, SHA-1, SHA-2), hash functions, Cryptography, Cyber security, Encryption

I. INTRODUCTION

SHA as it is abbreviation for "Secure Hash Algorithm" developed by NIST are widely used for data authentication and validation.Starting with the fact that to secure messages over the internet hashing algorithms transform a text string into an alphanumeric string, here both encryption and hashing techniques are prevailed. These two are considered similar sometimes in a way that they both take a string of useful text and convert it into something very different. However, unlike encryption a hash value cannot go back into its original message.
This study consists of comparisons based on chronological series of secure hashing algorithms. As each algorithm takes some specific amount of the time for the computation of hash value, there are some internal states, output size, block size, capacity and total rounds it takes to give a hash output. This study refers to account of SHA series algorithms in terms of performance throughput by monitoring the time consumed in computing hash value and finds algorithm which finishes off the task of user authentication in less amount of time compared to generic time taken for computation.Hash functions can be either cryptographically secure or not. Non-secure hash functions are useful for tasks like string mapping, error detection in communications protocols, so they still have a place. But they're not useful for cryptographic purposes, because they have certain weaknesses.

To be secure, a hashing function needs to have exceptional traits like flowing the change all over, that is if a one bit change in the message occurs it should result in an unpredictable change of approximately 50% of the bits in the digest. Secondly it must present non reversible outputs that one can't recreate a message given the hash. Thirdly it must be collision resistant. It must be out of question to come across two messages with the same hash value.Future study of SHA algorithms will compare secure hashing algorithm with best of the times network security algorithm to understand the real security needs of current era.

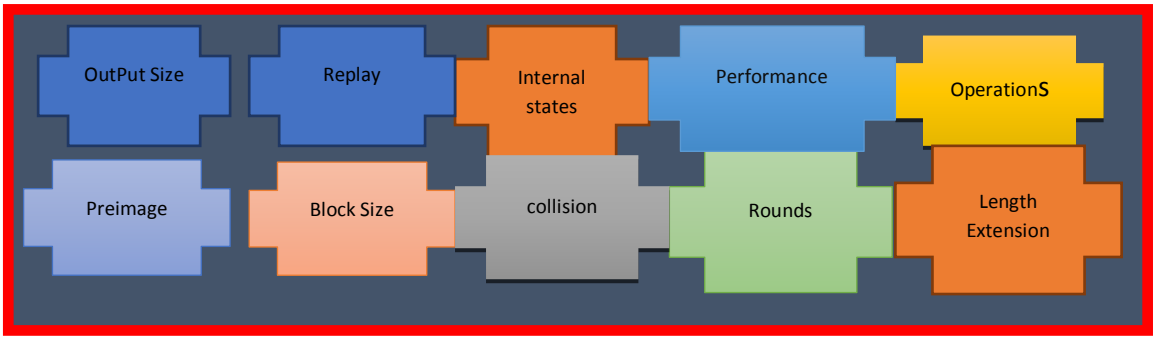

Figure-1: **building blocks of a hash function**

A). SHA Background

It is a fact that SHA-0 and SHA-1 are already breached years back but successful attacks on an algorithm are not enough to determine overall capability of an algorithm as the standards for cyber security are marked to be based on the factors that what is the average of Breaching Instances occurred and how much it takes to detect a breach and what is diligent fixation Time. After maintaining performance on above mentioned paradigms, the next valuable measure is estimate of overhead per event along with Uptime/Downtime of the system. A successful security algorithm is the one that deals with these consequences well in time with negligible losses. [3]

B). SHA- Attacks History

Algo	Launch	Input	Output	C.A	C.C	L.E.A
SHA-0	1993	Variable	160 bit	In 2004 42/160 bits are equal collisions SHA-0: 62/ 80 rounds	collision had complexity 2^{51} and took about 80,000 CPU hours	Yes
SHA-1	1995	variable	160 bit(20 byte)	2005SHA	2^{69} hashing operations 2^{63} operations	Yes
SHA-2 SHA-256 and SHA-512. SHA-224	2001	Variable	SHA-224:OP: 224 bits (28 bytes)SHA-256 : 32 bytes SHA-384: 48 bytes SHA-512: 64 bytes	N/A	N/A	N/A
SHA-3 SHAKE-128/256	2009	Variable	224, 256, 384 and 512 bits.	N/A	N/A	N/A

Table-1:Here is a list of attacks on SHA algorithms that seemed to be successful too.[2]

C). Shattered Attack

This is the attack in which they generated two different PDF files with the same SHA-1 hash in roughly $2^{63.1}$ SHA-1 evaluations. Over here the attacker is required "the equivalent processing power of 6,500 years of single-CPU computations and 110 years of single-GPU computations".

II. Problem Statement

This study is based on the statement that which features of Secure Hashing Algorithms determine their strength and existence in cyber security procedures and what are the factors that became vital cause of successful collision attack. Furthermore, second prime object is insisting on expanding the strengths to the extent where loopholes and limitations could damage the system the least.

III. Methodology

The methodology adopted for this study is based on the information available in online repositories, annual reports, proceedings and journal papers. All most all famous forums considerably serving the market for hashing are kept into account. Articles searched from the security solution provider's portfolio are included in the study as well. In addition to that cyber security giants are asked for some quires in the form of precise questions sent through emails and that specific pieces of information are also the part of paper's analysis work. As part of the policy, author is not mentioning the names of resources consulted online for this review study. Almost ninety articles are found on the topic that are referred here. This paper is the outcome of finally screened evidences from a big lot of accessed material. Furthermore, the readings for capacity analysis are copied from the sources performing computations on Skylake micro architecture system.

IV. Discussion & Analysis

A). Collision Attacks

As advanced cryptographic hash functions construct a diverse hash for every file [1]. A hash collision conveys a common hash for two separate files. SHA-1 collision was completed by creating a PDF prefix,[4] which was anticipated exclusively to allow to generate two documents with were based on haphazardly distributed distinct contents, but that would hash to the same SHA-1 digest [5].This collision was attained by combining many special cryptanalytic techniques in complex ways. The worth noticing point here is that even after this collision attack, falsifying digital digest is still not a piece of cake because of proper implementation of edging on the message digest with 20 random bits.

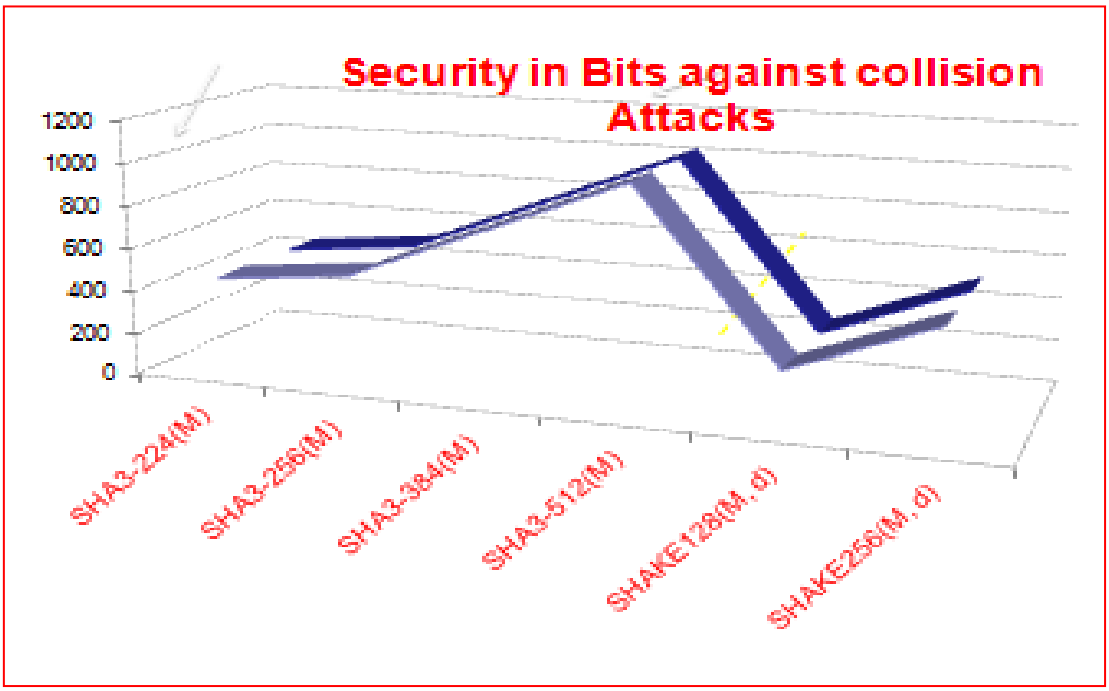

Figure-2: This chart expresses analyses for security constraint (SHA-3 variants)

Overall the precision of security is directly associated with the time a hacker is given to attack and generate keys online. At the start of SHA era, it was the most popular algorithm considered as predecessor of MD5 but even before the launch of SHA-1 in 2005 announcement of successful collision attack defamed it. Figure-2 confirms almost five variable segments of chart where a high value of 1000 bits is at SHA 3 with message digest of 512 bits. Later it fells down slightly to a level in between 400 to 600 bits. Figures resumed in 2009 after launch of SHA-3 extension.

B). Length Extension Attacks

This is basically the hash extender, a type of attack where an attacker can use Hash-message a and the length of message a to calculate Hash for an attacker-controlled message. An application is susceptible to a hash length extension attack if it affixes a secret value to a string, hashes it with a vulnerable algorithm, and commends the attacker with both the string and the hash, but not the secret key[9]. Then, the server pass on the secret key to decide whether or not the data returned later is the same as the original data.

The vulnerable hashing functions like MD-5 or SHA-0 work by taking the input message, and using it to transform an internal state. After all of the input has been processed, the hash digest is generated by outputting the internal state of the function [10]. It is possible to reconstruct the internal state from the hash digest, which can then be used to process the new data. In this way, one may extend the message and compute the hash that is a valid signature for the new message.[9]

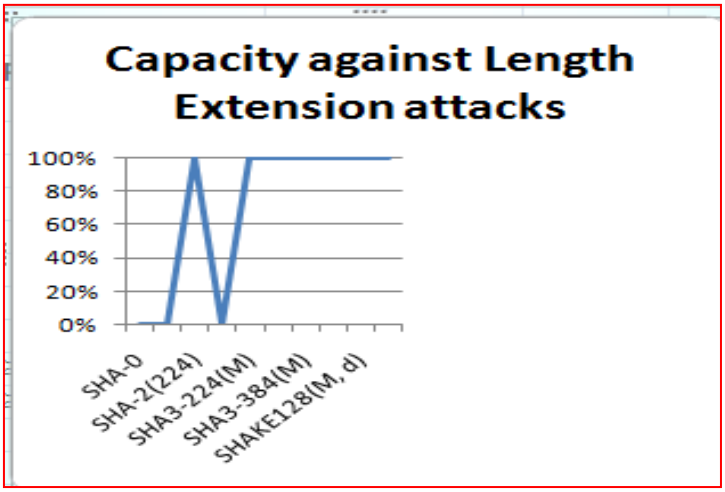

Figure-3: Resistance to Length Extension attacks

The line graph illustrates the three spikes; one bottom line level is presenting the point where algorithms exhibit zero safety against length extension attacks. The culmination point is marking to 100 % security for the above mentioned attack, which is then consistent for advanced series of SHA starting from SHA 2, units are measured in percentage .With regard to the length extension attack which begins when attackers attempt at least billions of tries, safety is peaked at 100% before falling dramatically to 0 % for previous SHA versions.

C). No. Of Rounds

Catering total number of rounds that an algorithm carries out to provide a hash value is somehow crucial though apparently it reflects as rolling a dice thousand times and reading only the last outcome has the same chance as rolling it once [6]. The hashing function may have a short cycle or fixed point [5]. A fixed point is a value for which hash (value) == value. This greatly reduces the security of an iterative hash, because any iteration done after reaching the fixed point will not change anything anymore.

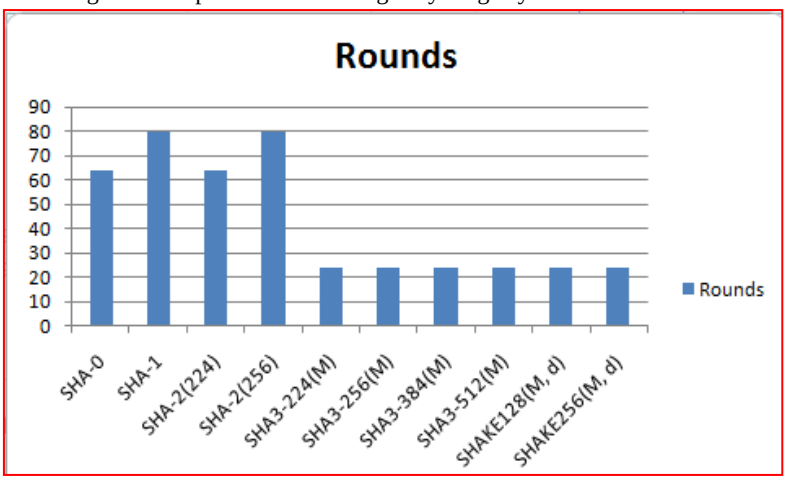

Figure:4: Illustration of number of rounds for each updating SHA

The graph shows changes in the number of rounds along with invention of new versions of algorithms for cyber hashing since 2000, and estimates trends until 2019. Between 2000 and the present day, the round count has been consistently declined after getting a peak at SHA-1 & SHA-2. Round tot up stood at 60 for SHA-0 at the starting years of SHA security and increased to a peak of 80 till the time SHA-2 was in place in 2005. In contrast to this, the sum has decreased steadily until the present time, this rate probably has leveled off at around minimum rounds.

D). Preimage Resistance

If a source is pumping out data that varies by only a few bytes, it is possible that some could collide leaving rest of the phenomena in the circle of their wish and employ that inner state of a message digest to achieve some authentic signature. That is why resistance to preimage is evolutionary for cyber security systems.

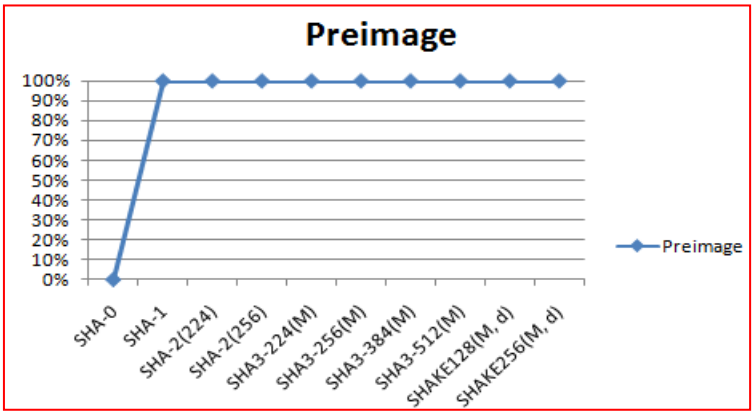

Figure-5: Preimage Resistance of SHA Algorithms

Overall, widely opposing trends for different algorithms in reversing a hash ability reflects that chance of acquiring preimage definitely goes down in SHA latest versions. The chart is presenting two levels broadly gaped, one bottom line states 0 % resistance for reimaging. The other peak tends to pose a reversed behaviour in the later versions of SHA algorithms.

Algorithm	Internal state size (bits)	Rounds
SHA-0/SHA-1	***160:(5 × 32)***	**80**
SHA-2	***256:(8 × 32)***	**64**
SHA-2(SHA-512/256)	***512:(8 × 64)***	**80**
SHA-3	***1600:(5 × 5 × 64)***	**24**

Table-2: A comparison of internal states of SHA algorithms [4].

V. CONCLUSION & SUMMARY

To summarize it is evitable to mention that SHA algorithms still are based on strong foundations. The recent official version of algorithms is based on the extensive aspects for example no mirroring, no collision possibility, fixed output hash string to variable input size. This analysis exhibits-maintained capacity of SHA advanced versions of algorithms to wear out length extension attacks, that even if attacker is familiar with message length and complexity along with in hand hashed version of a message, cannot predict key for that controlled string. Further development in the study while comprising preimage resistance heads to consistent pattern in the algorithm's behaviour. Which reveals that it is roughly unmanageable so far to find the invert of hashed value as well as some fractional mapping of hash values with some inputs. Contrary to fragile approach of producing many hash values of one input string, resilient feature of SHA algorithms that one hash value for one string is still considered a hub of security paradigms.

REFERENCES

[1]. Prashant P. Pittalia,"A Comparative Study of Hash Algorithms in Cryptography**",** International *Journal* of Computer Science and Mobile Computing, Vol.8 Issue.6, June- 2019.

[2].M. A. Al-Shabi,"A Survey on Symmetric and Asymmetric Cryptography Algorithms in information Security", International Journal of Scientific and Research Publications, Volume 9, Issue 3, March 2019 Edition [ISSN 2250-3153].

[3]. Safeeullah Soomro, Mohammad Riyaz Belgaum, Ruchin Jain, Zainab Alansari, "Review and Open issues of Cryptographic Algorithms in Cyber Security", Conference: IEEE International Conference on Computing, Electronics & Communications Engineering 2019 (IEEE iCCECE '19) London Metropolitan University, UK.
[4]. Aradhana Sahu, Samarendra Mohan Ghosh,"Review Paper on Secure Hash Algorithm With Its Variants", International Journal of Technical Innovation in Modern Engineering & Science (IJTIMES),2018.
[5]. A. Gowthaman, Sumathi Manickam, "Performance study of enhanced SHA-256 algorithm", International Journal of Applied Engineering Research,2017.
[6].B. Durdi, P. T. Kulkarni, and K. L. Sudha, "Selective encryption framework for secure multimedia transmission over wireless multimedia sensor networks," in Proceedings of the International Conference on Data Engineering and Communication Technology, 2017.
[7]. Priyanka Vadhera, Bhumika Lall, "Review Paper on Secure Hashing Algorithm and Its Variants", International Journal of Science and Research (IJSR), Volume 3 Issue 6, June 2014.
[8]. Dudhatra Nilesh, Malti Nagle, "The new cryptography algorithm with high throughput", 2014 International Conference on Computer Communication and Informatics.
[9]. Nithin R. Chandran, Ebin M. Manuel, "Performance Analysis of Modified SHA-3", International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST- 2015).
[10]. Elena Andreeva, Bart Mennink, Bart Preneel, **"**Security Reductions of the Second Round SHA-3 Candidates"**,** International Conference on Information Security, ISC 2010.
[11]. Ricardo Chaves, Georgi Kuzmanov, **"**Improving SHA-2 Hardware Implementations", International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2006.

