One-Way Hash Algorithms in Cloud Computing Security- A Systematic Review

Meena Kumari
PhD Research Scholar, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra, Haryana
Email: sanger.meena@gmail.com
Dr. Rajender Nath
Professor, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra, Haryana
Email: rnath@kuk.ac.in

Abstract
Cloud Computing (CC) has been emerged as the next generation of computing in IT Enterprise.CC migrates data and applications to huge data centers, where the management of data is totally under control of service providers. However this feature brings about various security concerns. Information authentication and its Integrity verification are such challenges in CC environment. Hash functions are versatile building blocks that are used in this context. This paper is an attempt to review and critically analyze existing hashing algorithms in CC environment. A comparison is also made among existing algorithms and possible future research directions are also given in this paper.

Keywords - integrity; hash function; digest; Non-invertible matrix; padding.

1. Introduction

The importance of cloud computing is increasing day by day and it is getting an emergent attention in the scientific and industrial communities. According to definition provided by NIST, CC is a model which enables an on demand broad network access to a shared pool of configurable resources, those can be provisioned with minimal management efforts or service provider interaction. In this demanding world the basis to embrace CC over standard IT deployments is flexibility, stability, rapid provisioning, reliability and scalability. The utilities can be leased and released by user through Internet on demand basis. However this new computing paradigm brings about certain security risks also. In CC users data and services does not resides in local premises, instead it is being processed in cloud environment or in a remote server, which introduces potential security threats. Data confidentiality and integrity verification and user authentication are examples of such threats.

Data integrity is a fundamental part of any secure system. By using the message digests generated by a cryptographic hash function a system administrator can identify illegitimate changes in documents. Cryptographic hash functions play a vital role in data integrity verification and message authentication in cloud data storage. This paper analyzes and discusses the hashing algorithms used in cloud computing environment and provides a comparative analysis of existing algorithms.

The rest of the paper is organized as follows: Section 2 introduces the hash functions specifically one way hash functions. Section 3 gives the overview of the existing one way hash algorithms in literature and Section 4 provides the comparative analysis. Section 5

describes the possible research directions and concluding remarks.

2. Hash Function

A hash function is a function which transforms strings of arbitrary finite bit length to strings of *n* bits for some fixed integer *n*. The inputs to a hash function are typically called messages, and the outputs are often referred to as message digests. The properties of hash functions can be used to greatly increase the security of a system administrator's network; when implemented appropriately they can attest the integrity and source of a file.

2.1 One way hash functions

One way hash functions are those functions which converts a variable length string into a fixed length binary sequence that cannot be reversed. Given only a digest, it should be computationally infeasible to find a piece of data that produces the digest back. Following characteristics must be present in a hash function to be a one way hash:

i. *One-way*: Given a hash H(M), it is complex to find the message M.
ii. *Preimage resistance*: Given *H*(M1) find *M2* such that *H*(M2) = *H*(M1).
iii. *Second preimage resistant*: Given a message M1, it is difficult to find another message M2 such that H(M1) = H(M2).
iv. *Collision resistant*: It is difficult to find two messages M1 and M2 such that H(M1) = H(M2).

3. Literature Review

Kavuri Satheesh et al. [1] has proposed an improved hash based message integrity verification process. Proposed message integrity algorithm was tested on attributed based encryption process. Proposed cloud based hash algorithm generates hash value size of 512 bits for each file in the third party cloud servers. Only authorized users can access the requisite files via his/her identity along with the message integrity value. It has also been proved experimentally that proposed cloud based hash algorithm performed well as compared to existing models in terms of file size, time and attacks.

Abutaha Mohammed et al.[2] has proposed a technique to design a practical one way hash algorithm by using singular matrix that cannot be reversed to produce hash value. They designed an algorithm that convert any invertible matrix into noninvertible one. They also made a comparison between their proposed algorithm with MD5, SHA1, and SHA-512 by using two factors which are data size, matrix size.

Zheng Yuliang [5] et al. had proposed a one-way hash algorithm known as HAVAL. HAVAL compresses a message of arbitrary length into a fingerprint of 128, 160, 192, 224 or 256 bits. In addition, HAVAL has a parameter that controls the number of passes a message block (1024 bits) is processed. A message block could be processed in 3, 4 or 5 passes. By combining output length with pass, they provided fifteen (15) choices for practical applications where different levels of security are required. Experiments conducted by authors showed that HAVAL is 60% faster than MD5 when 3 passes are required, 15% faster than MD5 where 4 passes are required, and as fast as MD5 when 5 passes are required.

Berisha Artan et al. [3] had discussed non invertible matrix in GF (2) which could be utilized as multiplication matrix in Hill Cipher technique for one way hash algorithm. Authors proposed a class of matrices in GF(2) which are non-invertible and easy to generate.

Acharya Bibhudendra et al. [4] had suggested some efficient methods for generating self-invertible matrix for Hill Cipher algorithm. These methods include less computational complexity because inverse of the matrix is not required while decrypting in Hill Cipher.

Abutaha Mohammed et al. [6] has proposed a model for hash algorithm by using non-invertible matrix and proved four essential

requirements needed for a practical hash algorithm.

S.G. Srikantaswamy [7] had proposed a method for developing hash function with traditional XOR operation with matrix multiplication, which makes the hash calculation process irreversible. The Proposed scheme also includes the generation of checksum along with padding of the plaintext data.

Hamamreh A. Rushdi [8] had suggested a new technique in Hill Cipher algorithm to overcome its major problem of noninvertible key matrix. The paper has also suggested an enhancement to the security of Hill Cipher against known plaintext attack because all steps in Hill Cipher are dependent on linear algebra calculation this was possible by using public key. key generation depends on various options and without using linear algebra steps. Therefore, it would be difficult for the attacker to get the key. Finally, the authors presented a proper solution for the problem of sending secret key () for the first time.

Hamamreh A. Rushdi [9] had Proposed a hash algorithm based on linear combination of matrices to generate a non-invertible matrix, which takes advantage about of the compact representation of a set of numbers in a matrix and is strong collision resistance.

4. Comparative Analysis

This section provides a comparative analysis of different one way hashing algorithms available in literature. The Table 1 illustrates the details of the papers referenced along with the problem identified.

Table 1: Comparative Analysis

S.No	Author	Proposed work	Merits	Demerits
1.	Abutaha Mohammed et al.(2011,2013)	Used Non-invertible matrix for one way hash function.	1. Better performance in terms of time for hash calculation when matrix size is less. 2. Resistant to collision by using hamming distance algorithm.	1. Hash calculation time is dependent on file size. 2. Algorithm for conversion of an invertible matrix into non invertible is not specified. 3. Works on data integrity verification but not on data confidentiality.
2.	Berisha Artan et al. (2012)	Proposed a technique for generating non invertible matrices in GF (2) and one-way hash algorithm to generate hash value.	1. Proposed model is practical for small amount of data, which implies that can be used for authentication (password verification), MAC. 2. Automation of the algorithm given by Abutaha Mohammed et al.	1. Efficient for files of size less than 6.2 KB. 2. Hash calculation time increases exponentially as the file size increases.

3.	Kavuri Satheesh et al. (2014)	Extension of Whirlpool hash algorithm	1. Data partitioning concept is used for effective storage 2. Access time is less than Whirlpool. 3. Works on both data integrity verification as well as data confidentiality.	1. TPA is vulnerable to attack. How one can trust TPA. There is shifting of trust from service provider to TPA. 2. Hash calculation time is dependent on file size.
4.	S.G.Srikantaswamy et al. (2013)	Proposed a model that uses a non-invertible key matrix and XOR function for hash calculation.	1. Easy and fast method. 2. Increased Avalanche effect due to use of XOR function.	1. Truncation of final hash results in loss of data or precision. 2. Practical implementation is missing.
5.	Rushdi A. Hamamreh et al. (2013)	Proposed a hash algorithm based on linear combination of matrices to find noninvertible matrix, that takes advantage of the compact representation of a set of numbers in a matrix and are strong collision resistance.	1. The proposed algorithm didn't find any collision after 72 bit hash value length.	1. Incomplete steps of hash generation. ie how to generate final hash value is not specified. Intermediate hash values are only generated. 2. Hash calculation time is dependent on file size.

5. Conclusion and Future Research Directions

This paper has presented a critical analysis of the existing one way hash functions. It has been found that most of the algorithm are efficient for small data size. For a large data size their performance degrades exponentially.

With reference to future research direction, a program for conversion of an invertible matrix into non-invertible one and vice versa could be developed. Experimental evaluation of the algorithms could be performed to ascertain the collision resistant property of hash functions. Certain improvements could be incorporated in existing algorithms to enhance the speedup of digest calculation.

References

[1] Kavuri Satheesh et al.," *Data Authentication and Integrity Verification Techniques for Trusted/Untrusted Cloud Servers*", In Proc. Of International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp 2590 - 2596, 24-27 Sept. 2014.

[2] Abutaha Mohammed et al.," *New One Way Hash algorithm Using Non-Invertible Matrix*", In: Proc. Of International Conference on Computer Medical Applications (ICCMA), pp 1-5, Jan. 2013.

[3] Berisha Artan et al.,"*A Class of Non Invertible Matrices in GF (2) for Practical One Way Hash Algorithm*", International Journal of Computer Applications, Volume 54– No.18, pp 15-20, September 2012.

[4] Acharya Bibhudendra et al.,"*Novel Methods of Generating Self-Invertible Matrix for Hill Cipher Algorithm*", International Journal of Security, Volume 1, Issue 1, pp 14-21.

[5] Zheng Yuliang et al., "*HAVAL - A One-Way Hashing Algorithm with Variable Length of Output*", Springer, pp 83-104, 1993.

[6] Abutaha Mohammed et al.," *A Practical One Way Hash Algorithm based on Matrix Multiplication*", International Journal of Computer Applications, Volume 23– No.2, pp 34-38, June 2011.

[7] S.G.Srikantaswamy et al., "*Hash Function Design Using Matrix Multiplication, Ex-or, Checksum Generation and Compression technique Approach*", International Journal of Computer Science, Information Technology, & Security, Vol 3,Number 1, 2013.

[8] Hamamreh A. Rushdi et al.,"*Design of a Robust Cryptosystem Algorithm for Non-Invertible Matrices Based on Hill Cipher*", IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.5, May 2009.

[9] Rushdi A. Hamamreh et al.," *Hash Algorithm for Data Integrity based on Matrix Combination*", In: Proc. Of The International Arab Conference on Information Technology, 2013.