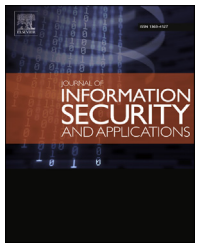
Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/jisa



ASH-512: Design and implementation of cryptographic hash algorithm using co-ordinate geometry concepts

Pallipamu Venkateswara Rao [a,*], *K. Thammi Reddy* [b], *P. Suresh Varma* [a]

[a] *Department of Computer Science and Engineering, Adikavi Nannaya University, Rajahmundry 533105, Andhra Pradesh, India*

[b] *Department of Computer Science and Engineering, GITAM University, Visakhapatnam 530045, Andhra Pradesh, India*

ARTICLE INFO

Article history:
Available online 15 November 2014

Keywords:
Cryptography
Hash function
Message digest
Message authentication
Co-ordinate geometry concepts

ABSTRACT

Sending and receiving information over the internet is easy, fast and cost effective due to development of information technology. These days most of the information either personal or official is communicated through internet only. Most of the paper based documents are replaced by e-documents. So, there is a need to protect confidential or sensitive information in the computer as well as during transmission. Message digest is a mechanism, which is used to ensure message authentication and integrity. 'Algorithm for Secure Hashing-512' (ASH-512) is a novel algorithm proposed in this paper, which is designed using co-ordinate geometric concepts. The algorithm accepts 1024-bits as input and produces 512-bits as output. The algorithm is more secure and easy to construct and thus made this algorithm special. The algorithm is implemented in Java and the results are compared with standard cryptographic hash algorithm-SHA2 (512) and Whirlpool-512.

1. Introduction

Cryptographic hash functions are the tools of cryptography to generate message digest or hash code, which is used in variety of information security applications and protocols. Message digest is the essential element in digital signature schemes and is also used in message integrity, password protection, random number generation, challenge-response protocol etc. Hash algorithm takes arbitrary length input (or empty) and produces a fixed length hash code or message digest as output. The need for novel secure hash algorithms increases proportionately with the increased demand of secure communications. The existing hash functions are actively used in various information security applications and protocols, but currently not able to prevent attacks effectively because attacks are increased due to enhanced computing power of the computers.

The MD2, MD4, RIPEMD family (Mironov, 2005), MD5 (Gauravaram et al., 2006; Wang and Yu, 2005), SHA0 (Kahate, 2006), GOST, Whirlpool, Tiger, SHA1 (Biham et al., 2005; Wang et al., 2005), SHA2 (Federal Information Processing Standards Publication, 2012; Gilbert and Handschuh, 2004) are some of the popular algorithms (Tiwari and Asawa, 2010; Brown et al., 2008) with message digest of different sizes, but their security power is reducing gradually (Ristenpart et al.,

* *Corresponding author*. Tel.: +91 9441447037.
E-mail address: venkat.aknu@gmail.com (P. Venkateswara Rao).
http://dx.doi.org/10.1016/j.jisa.2014.10.006

2011; Knudsenl et al., 2007). The time complexity for finding a collision for MD4 is about 2^{23} MD4 operations without the multi-message modification, and is about 2^{8} MD4 operations with the multi-message modification (Wang and Yu, 2005). The time complexity for finding a collision for HAVAL-128 is about 2^{13} MD4 operations without the multi-message modification, and is 2^{7} HAVAL-128 operations with the multi-message modification (Wang and Yu, 2005). The time complexity for finding a collision for RIPEMD is about 2^{30} RIPEMD operations without the multi-message modification, and is 2^{18} RIPEMD operations with the multi-message modification (Wang and Yu, 2005). The time complexity for finding a collision for SHA-0 is about 2^{61} SHA-0 operations without the multi-message modification, and is 2^{45} SHA-0 operations with the multi-message modification (Wang and Yu, 2005). Stevens et al. have shown how, at an estimated cost of 2^{39} calls to the MD5 algorithm compression function, Chosen prefixes (any two) are P_x and P_y and suffixes are S_x and S_y to be constructed such that the results of concatenation of $P_x||S_x$ and $P_y||S_y$ collide under MD5 (Stevens et al., 2012). Xiaoyun et al. have published a paper on HAVAl-180, which has shown that any message of 1024 bits (m), they have modified on m and the modified message can collide with another message m_1 with the probability of $1/2^7$, wherein $m_1 = m + \Delta m$, (Δm is a fixed difference selected in advance) (Xiaoyun et al., 2005). An attack was announced by Xiaoyun et al. on SHA-1. The attack can find collisions in the SHA-1, requiring less than 2^{69} operations and a brute-force search would require 2^{80} operations (Schneier on security: SHA-1 broken). In 2005, Vincent Rijmen and Elisabeth Oswald published a paper entitled 'Update on SHA-1'. In this paper, they show that an attack on a reduced version of SHA-1, with 53 rounds out of 80 rounds, there is a possibility to find collisions less than 2^{80} operations (Rijmen and Oswald, 2005).

The National Institute of Standards and Technology (NIST) announced as a public request for new secure hash algorithm (Third-round report of the SHA-3 cryptographic hash algorithm competition, 2012; Status report on the second round of the SHA-3 cryptographic hash algorithm competition, 2011; Brown et al., 2008), which is named as SHA-3. In this contest Keccak algorithm (Status report on the first round of the SHA-3 cryptographic hash algorithm competition, 2009) has been selected as SHA-3 family (Martin, 2008) on 2012 and it is yet to be standardized. Taking NIST initiative as inspiration, we tried to design more secure hash algorithm. Section 2 of this paper denotes an overview of popular cryptographic hash functions.

Section 3 proposes a novel secure hash function (ASH-512), which is developed using co-ordinate geometry concepts. Section 4 deals with results and discussion. The security and efficiency analysis of proposed algorithm is discussed in Section 5. Finally Section 6 concludes with security strength of ASH-512.

2. Review of cryptographic hash functions design

Hash functions are currently a hot topic of research in cryptography. The area of information security welcomes new approaches to the design of secure hash functions. Innumerable hash functions (Sheena Mathew and Jacob, 2010; Knudsenl et al., 2007) have been developed and used in various security applications. The designs of some of the existing algorithms are briefly discussed below (Ferguson, 2010).

a. *Iterated hash functions*

The cryptographic hash function takes an input of arbitrary length of message and produces a fixed length of message digest or hash code as output. It is difficult to design an algorithm that accepts messages of variable length and/or gives message digest of variable length. The hash functions are designed so far based on compression function, which accepts fixed length input and produces a fixed length output. The design principle of iterated hash functions involves dividing the input into fixed length blocks and each block is passed into the compression function. The resultant algorithm is named as "Iterated hash function" (Stallings, 2003).

b. *Hash functions based on block ciphers*

In the iterated hash function, designer is concentrated on compression function. The design of compression function is not an easy task. Block cipher is an another approach to construct a compression function on available cryptographic primitive. The advantage of this design is the reusability of existing implementations in software and hardware. The hashing and encryption are required for every application and complexity of the implementation is minimized by using a block cipher. The disadvantage is that the block cipher based hash functions are less efficient than dedicated hash functions.

c. *Hash functions using modular arithmetic*

In this process modular arithmetic is used as a basic building block for design of a compression function in cryptographic hash function. The reusability of existing implementation is allowed similar to asymmetric key cryptosystem. The advantage of this mechanism is that to change the security level by changing the value of modulus (M). However, this hash function exhibits very less performance compared to block cipher based hash function.

d. *Dedicated Hash functions*

The dedicated hash functions are special functions which are designed for explicit purpose of hashing. These are designed by keeping two things in mind, which are reusability and good performance.

The hash functions of this type which have received much attention in practice are based on the MD4 algorithm. MD4 (Rivest, 1992a) was originally designed towards software implementation on 32-bit platforms. Another hash function MD5 (Wang and Yu, 2005; Rivest, 1992b) is an improved variant of MD4. Based on the principle of MDx