Role of Hash Function in Cryptography

Edem Swathi[1], G. Vivek[2], G. Sandhya Rani[3]

[1]Dept. of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad
Email: swathiedem@cbit.ac.in
[2]Dept. of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad
Email: vivek@cbit.ac.in
[3]Dept. of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad
Email: sandhya_g@cbit.ac.in

Abstract— *Hash functions were introduced in cryptology in the late seventies. These are used as a tool to protect the data integrity. Soon it has become clear that they are very useful to solve other security issues in computer networks and in telecommunications. Cryptographic hash function is a mathematical concept, mainly used as a part of the digital signature schemes. In this paper, we describe hashing, cryptography, cryptographic hash function and their properties. It gives an overview of hash functions, compression functions which are designed from scratch and made out of block ciphers.*

Keywords*— *Cryptography, compression functions, data integrity, Message Digest(MD's), Secure hash algorithm(SHA), whirlpool.

I. INTRODUCTION TO HASH FUNCTION AND CRYPTOGRAPHY

Cryptography is study of the techniques used to communicate and to store data privately and securely, without being intercepted by third parties. It includes processes such as encryption, hashing and steganography.

1.1 Cryptography

Now-a-days people use cryptography everyday without realizing (such as bank transactions, website logins, etc) to secure and protect their privacy. Earlier cryptography was concerned solely with message confidentiality i.e. encryption. Encryption is the process of converting plain text (readable form) to cipher text (unreadable form). Decryption is the process of converting cipher text back to the plain text. There are several cryptographic algorithms. But based on the numbers of keys that are used for encryption and decryption, the cryptographic algorithms are categorized as follows:

i. Secret key cryptography: Uses single key for both encryption and decryption; also called symmetric encryption. Primarily used for privacy and confidentiality.

ii. Public key cryptography: Uses one key for encryption and another key for decryption. It is called as asymmetric encryption. Primarily used for authentication and key exchange.

iii. Hash functions: Uses a mathematical transformation to irreversibly "encrypt" information, providing a digital fingerprint.

1.2 Hash Function

Hash function is a function which maps data of arbitrary size to data of fixed size. The values returned by hash function are called as hash values, hash codes, hash sums or simply hashes. Hash functions are primarily used in hash tables.

A hash table is a collection of data items which are stored in such a way as to make it easy to find them later. Each position of the hash table is called as slot. A slot can hold a data item and is assigned with a integer value, starting at 0. A slot is identified by key. The data item stored in slot is referred as value.

0	1	2	3	4	5	6	7	8
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Fig. 1: A hash table with 8 empty slots

For instance, we have 9 slots named with 0,1,2,…,8 respectively. Initially, the hash table contains no data items, so every slot is empty as shown in Fig. 1. Assume that we have set of integer items 67, 24, 11, 98 and 48. The first hash function, known as "remainder method", simply takes an item and divides it by the table size, returning the remainder as the hash value (h(item)=item %9). After applying remainder method on the above mentioned set, the resultant hash table is as follows:

0	1	2	3	4	5	6	7	8
NULL	NULL	11	48	67	NULL	24	NULL	98

Fig. 2: A hash table after applying remainder method after applying on given set of data

Now, 5 out of the 9 slots have been occupied. This is referred to as load factor. Load factor is denoted by λ:

$$\lambda = \frac{\text{number of items}}{\text{table size}} \qquad (1)$$

For the above hash table, the load factor is λ=5/9. For example, if 44 is the next item in our collection, then it would have a hash value of eight (44%9=8). Since 98 has hash value of 8, we would have a problem. This scenario is referred as collision or clash.

From a given collection of items, if a hash function maps each item to a unique location/slot is called as perfect hash function or universal hash function. In practice, it is difficult to assign unique slot for each data item. A good hash function should have minimum collisions, should be easy and quick to compute. A hash function works based on following assumptions:

i. It always returns a number to each data item.
ii. Two equal objects will have same
iii. Two unequal objects will always not have different numbers.

Some common hash functions are:

i. Division Remainder: Computes hash value from key using the % operator.
ii. Truncation or Character/Digit Extraction: It works based on the distribution of digits or characters in the key. More evenly distributed digit positions are extracted and used for hashing purposes.
iii. Folding: It involves splitting keys into two or more parts and then combining the parts to form the hash addresses.
iv. Radix Conversion: Transforms a key into another number base to obtain the hash value.
v. Mid-Square: The key squared and the middle part of the result taken as the hash value.
vi. Use of a random number generator: Given a seed as parameter, the method generates a random number.

II. VARIOUS CHARACTERISTICS OF CRYPTOGRAPHIC HASH FUNCTION

One way to preserve the integrity of a document, a fingerprint of a person can be used. While preparing the document, the authors fingerprint is given on the bottom of the page. To ensure that the document has not been changed, the authors fingerprint and fingerprint on the document can be compared. Another way to preserve the integrity of a document is use of cryptographic hash functions. In this, the message is passed through cryptographic hash function algorithm; it creates a compressed image of the message that can be used like a fingerprint [2]. To check the integrity of the message, we run the cryptographic hash function algorithm again and compare the new message digest with the previous one.

Hash functions are versatile cryptographic building blocks, with applications such as the protection of the authenticity of information and digital signatures. Cryptographic hash function is a mathematical algorithm that takes an arbitrary size of data and encrypts that to a fixed size of data, typically somewhere between 128 and 512 bits. The input data is often called the message, and the output (the hash value or hash) is often called the message digest or data fingerprint. This hashing process can be denoted as:

$$H = H(M) \qquad (2)$$

where M is the input message and h is the digest generate by the hash algorithm H.

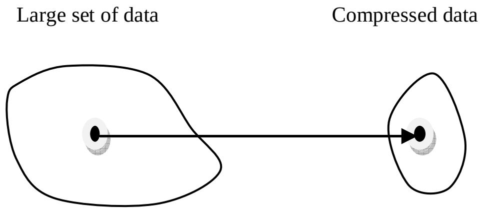

Fig. 3: A cryptographic hash function encrypts a large set of data to a smaller set of data.

Different applications expect different properties of hash function, but some properties are always expected. A cryptographic hash function(H:a->b) has to fulfill different requirements ([1]): It has to be a so-called one-way function that provides the property of irreversibility, which describes the computational impossibility to determine any input data M from a hash value H(M). This property is also called as pre-image resistance. Second requirement is second pre-image resistance. In this, given input message M1, it should be difficult to find different input message M2 such that hash(M1) = hash(M2). Further, the reproducibility property of a hash function has to ensure that if any input data M and M’ are equal, then also the output data H(M) and H(M’) are equal. Contrariwise, in case M and M’ are not equal, the corresponding hashes H(M) and H(M’) have to be unequal. This requirement is called collision resistance [14]. A fifth requirement of cryptographic hashes is the bit sensitivity. It states that small changes in the input data M (e.g. by alternating one bit) should lead to a big change in the output data H(M).

One of our goal of this paper to point out the problems with cryptographic hash functions. Designing a fast and secure cryptographic hash function was believed as a simple task for some years. In early/mid 90s, two hash functions namely MD5 and SHA-1 were developed then used universally. Cryptographic hash functions must not only have good statistical properties. They must also withstand serious attack by malicious and powerful attackers who are trying to invade our privacy. Only two families of hash functions came to be used widely (namely the MD and SHA families). Many algorithms have been proposed, but most of them soon turned out to be too weak to resist attacks.

A cryptographic hash function takes a message of arbitrary length and creates a message digest of fixed length. Creating such a function is accomplished using iteration. In this process, instead of using a hash function with variable size input, a function with fixed-size input is created and is used a necessary number of times. The fixed-size input function is referred to as a compression function [3]. It compresses an n- bit string to create an m-bit string where n is normally greater than m. This scheme is referred as an iterated cryptographic hash function [4].

Merkle_Damgard Scheme: The Merkle-Damgard scheme is an iterated hash function that is collision resistant if the compression function/hash function is collision resistant. This scheme is basis for many cryptographic hash functions today.

III. COMPRESSION FUNCTIONS

Based on Merkle-Damgard scheme, a cryptographic hash function is usually designed from scratch or made out a block cipher.

The hash functions which are designed from scratch are Message Digest(MD) algorithms and Secure Hash Algorithms(SHA) [2]. And the hash function which is designed from block cipher is Whirlpool.

1.1 Message Digest (MD)

All MD hash algorithms are designed by Ron Rivest. We have MD2, MD4 and MD5 algorithms in this family. MD2 is designed in 1989, divides the message into blocks of 128 bits and creates 128-bit digest. It is computationally slightly expensive in terms of memory. For MD2, a pre-image resistance attack found [5]. Then MD4 algorithm is designed in 1990. It divides the message into block of 512 bits and creates 128-bit message digest. It is not strong collision resistant. In this, finding collision is done fast [6]. The MD5 algorithm is a common hash function [1], takes a message and divides it into blocks of 512 bits and creates 128-bit digest. It is derived from MD4 algorithm. It has vulnerabilities and still it is used as a checksum to verify data integrity, but only against unintentional corruption. Collision attacks[7] and pre-image attacks[8] are found for MD5 algorithm.

1.2 Secure Hash Algorithms (SHA)

It is sometimes referred to as Secure Hash Standard (SHS). It is developed by National Institute of Standards and Technology (NIST) and published as Federal Information Processing Standard (FIP 180). The SHA-0 algorithm takes a message and divides it to block of 160 bits. It is not a strong collision resistant and a collision is found [9]. SHA-1 is 160-bit hash function, looks like a MD5 algorithm. In SHA-2 group, we have two hash functions, with different block sizes. They are SHA-256 and SHA-512. The SHA-256 algorithm divides the given message into block of 512 bits and creates 256-bit message digest. In SHA-512 algorithm, the message is divided into block of 1,024 bits and creates 256-bit message digest. Attacks on these versions break only a reduced version of the hash, not on the final version of hash [10][11][12].

MD5 and SHA algorithms are descended from MD4. They can be compared by using some key parameters like block size, key length, cryptanalysis, iterations and total steps. SHA algorithm is considered more highly secure than MD5 as it needs 2^160 bit operations to find out the original message where as MD5 needs 2^128 bit operations. The length of message digest in SHA is 160 bits where as in MD5 it is 128 bits. MD5 is faster compared to SHA algorithm as it requires only 64 iterations whereas SHA needs 80 iterations. Both the algorithms need padding, finger print and almost same numbers of resources are utilized. When security is main concern SHA is a good choice and when time is main concern MD5 is a good choice.

1.3 Whirlpool

It is based on the use of a block cipher for the compression function. A block cipher is a method of encrypting the data to produce cipher text, in which a cryptographic key and algorithm are applied to a block of data at once as a group rather than to one bit at a time. Whirlpool is designed by Vincent Rijmen and Paulo Barreto. It takes a message and divides it into block of 512-bits and creates message digest of 512-bits [13]. As its message digest bit are comparatively more than other algorithm discussed in this paper, difficult to occur collision. It is more resistant to the usual attacks. So it provides more security.

IV. CONCLUSION

Many security systems are using cryptographic hash functions. Though the research on cryptographic hash functions is going on and more progress is yet to be achieved. These functions provide certain security properties and play a key role in building various security applications related to digital signatures, authentication and message integrity. This paper emphasizes on cryptographic hash function families, compression functions and formal terminology. In the last section, we analyzed the various algorithms' block sizes, message digest sizes and the properties on which attacks have been made theoretically and practically.

REFERENCES

[1] Zhijie Shi, Chujiao Ma, Jordan Cote, and Bing Wang: Hardware Implementation of Hash Functions

[2] Prof. Rakesh Mohanty, Niharjyoti Sarangi and Sukant Kumar Bishi: A Secured Cryptographic Hashing Algorithm

[3] Bart Preneel, Katholieke Universiteit Leuven, K. Mercierlaan: Cryptographic Hash Functions

[4] Behrouz A Forouzan, Debdeep Mukhopadhyay: Cryptography and Network and Security, Third edition.

[5] Soren S. Thomsen: An improved preimage attack on MD2

[6] Yu Sasaki, Yusuke Naito, Noboru Kunihiro, Kazuo Ohta: Improved Collision Attacks on MD4 and MD5

[7] Tao Xie, Fanbao Liu, Dengguo Feng: Fast Collision Attack on MD5

[8] Yu Sasaki, Kazumaro Aoki: Finding Preimages in Full MD5 Faster Than Exhaustive Search

[9] Stephane Manuel; Thomas Peyrin (2008-02-11). Collisions on SHA-0 in One Hour. FSE 2008

[10] Florian Mendel, Tomislav Nad, Martin Schlaffer (2013-05-28). Improving Local Collisions: New Attacks on Reduced SHA-256

[11] Somitra Kumar Sanadhya; Palash Sarkar (2008-11-25). New Collision Attacks against Up to 24-Step SHA-2. Indocrypt 2008

[12] Kazumaro Aoki; Jian Guo; Krystian Matusiewicz; Yu Sasaki; Lei Wang "Preimages for Step-Reduced SHA-2", Asiacrypt 2009

[13] William Stallings,The Whirlpool Secure Hash Function, Taylor and Francis Group, 2006

[14] An Overview of Cryptographic Hash Functions and Their Uses, John Edward Silva, 2003,GIAC Security Essentials Practical Version