HASH MECHANISMS SUITABLE FOR TRANSMISSION OF SAFETY- RELATED MESSAGES

Mária Franeková

Department of Control and Information Systems, Faculty of Electrical Engineering, University of Žilina, Veľký diel, 010 26 Žilina, Slovak Republic, Phone: ++(0)41-5133346, E-mai:maria.franekova@fel.utc.sk

Summary The paper deals with problems of data security in the safety – related transmission systems defined within railway process. In details is analysed possibility of using hash -coding techniques, which are generally defined in norm EN 50159-2. According to specification of transmission in railway applications the type of secure hash keying and non-keying techniques are recommended. There are discussed the results of experimental part, which are related to the comparison of time relation of hash code determination.

1. INTRODUCTION

During communications across open transmission systems must be an adequate defence against the identified treats of safety system guaranteed. Requirements on communications between the parts of signalling across the open transmission system are defined in the norm EN 50159-2 [1]. This norm defines the necessary conditions for communications only, without their detailed specification of safety mechanisms. The library of defences can be take over from the types of defences, which are used in the non – safety data transmission systems and expanded about the special safety mechanisms (depend on types of threats in open transmission systems and specification of transmission).

In the case if non – authorised access cannot be eliminate in the norm [1] the used of strong cryptographic mechanisms is recommended. In this case the two solution of creating the safety protocol are recommended:

Model of message representation - the type B0 - to user data are added safety related data (e. g. time stamp), redundant data from safety code and all contents of telegram is ciphering,

Model of message representation - the type B1 - created telegram is non-ciphering, but from the message the cryptographic redundancy is produced (on the base of hash function or with the used of digital signature).

On the present, many cryptographic algorithms exist. With the used them safety functions of integrity, confidentiality and authentication on of messages can realised [2].

The paper deals with the analysis of hash mechanisms, which can be used in the part of model representation the type B1, sign as cryptographic redundancy (see Fig. 1). For the selection of hash mechanisms is necessary to issue from following specification of transmission within railway communication subsystem:

1. Processes, that are control in railway applications are depended of time.
2. The group communication processes need the tools for authentication of all sources.
3. Messages are time valid and they are transmitted in cycle. If delay between messages is over defined value then it must predict failure state.
4. During safety-related transmission (transmission of very sensitive information which corruption can cause material damage or damage of human living) is necessary to avoid repetition, deleting, re-sequences, delaying, corruption and masquerading of messages and to guarantee the value of safety integrity level defined in the norm [3].

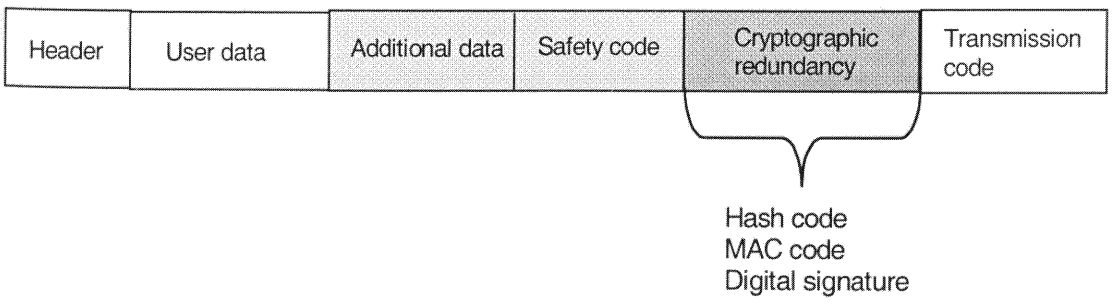

Fig. 1 Model of message representation within the open transmission system - type B1

2. IMPORTANT PARAMETERS FOR SELECTION OF HASH MECHANISMS WITHIN RAILWAY APPLICATIONS

Hash mechanisms are important building blocks of different cryptographic applications [4]. The purpose of hash mechanisms is to produce a "fingerprint" of a file, message or other block data. There are basic elements of digital signature schemes and authentication protocols. A hash function H accepts a variable size message M as an input and produces a fixed-sizes output, referred as a hash code or message digest $h = H(M)$, which is function of all bits of the message and provides an error-detection capability. An alternative authentication technique MAC (Message Authentication Code) involves the use of secret key K to generate a small fixed-sizes block of data, known as cryptography checksum MAC = $C_K(M)$. Unlike MAC code a hash code does not use a key but is function only of the input message.

Hash functions can be used within safety-related transmission of messages mainly for integrity check of messages along with sources and as part of digital signature schemes. Model representation of message type B1, shown on the Fig. 1, recommends the use of key hash function or non - key hash function. Today it is known variety of ways, in which a hash code can be used provide message authentication (see in the Tab. 1).

Tab. 1. Key and non-key hash functions.

Key hash function	Non-key hash function
MAC (ISO/IEC 9797-1) **CBC-MAC** (ANSI X9.19) **HMAC - MD5** (RFC2043) **HMAC-SHA1**(RFC 2401) **HMAC-RIPEMD160**(RFC2857)	**MDC-2** (ISO/IEC 10118-2) **MD2** (RFC 1319) **MD4** (RFC 1320) **MD5** (RFC 1321) **SHA1** (FIPS 180-1) **SHA256,384,512** (FIPS 180-2) **RIPEMD128,160** (ISO/IEC10118-3) **MASH 1, 2** (ISO/IEC 10118-4)

The key an non-key hash function, which are standardised can divided into following groups: Message Digest Code MDC, Message Digest MD (older types of hash functions), Secure Hash Algorithms SHA, message digest algorithm developed under the European RACE Integrity Primitives Evaluation RIPEMD and MASH. Other not so known hash functions are used, too e.g. types of Haval, Tiger, Whirpool, Sapphire, Snefru and so on [6].

In the present we cannot consider some types of hash algorithm as computationally safety. For transmission of safety related-messages the hash algorithms have too fulfil the following properties:

a) one-way property

For any given value h it is computationally infeasible to find x such that $h = H(x)$.

b) weak collision resistance

For any given block x, it is computationally infeasible to find $y \neq x$ with $H(y) = H(x)$.

b) strong collision resistance

It is computationally infeasible to find any pair (x, y) such that $H(x) = H(y)$.

3. SECURITY OF HASH FUNCTIONS

Just as with symmetric and asymmetric encryption, we can group the attacks on hash mechanisms into two categories: brute force attacks and cryptoanalytic attacks. We have to regard differently security of non-key hash mechanisms and MAC. The brute force attack on a MAC is a more difficult because depend on the size of key too. In recent years, there has been considerable effort, and some successes, in developing cryptoanalytic attack on hash function. To understand these, we need to lock at the overall structure of a typical secure hash function (see Fig. 2). This structure referred to as an iterated hash function and is the structure of most hash function in use today, including MD5, SHA-1 and RIPEMD-160. Message M is completed to undivided bits number of blocks M_i for i=1,2, ...,L (frequently to 512 bits). The hash algorithm involves repeated use a compression function f, that takes two inputs (an n - bit input from the previous step, called the chaining variable and a b-bit blocks) and produces an n - bit output. At the start of hashing, the chaining variable has an initial value IV that is specified as a part of algorithm. The final value of the chaining variable is the hash value. Usually $b>n$; hence the term compression. The hash function can be summarised as

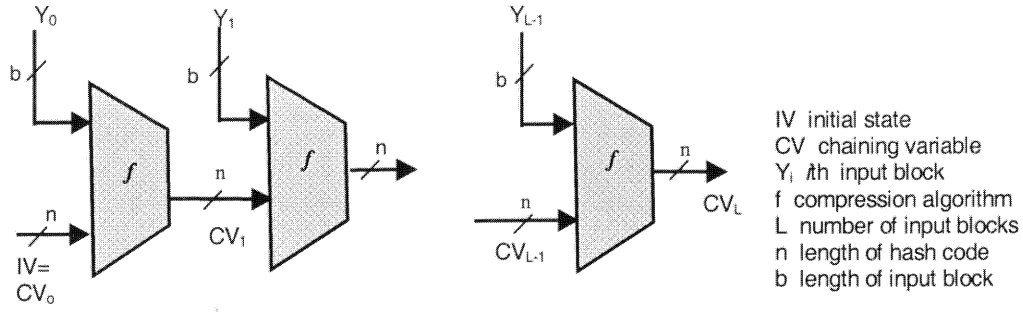

Fig. 2 General Structure of Secure Hash Code

follows:

$$H_0 = IV,$$
$$H_i = f(H_{i-1}, M_i), i = 1,2,....,L \quad (1)$$
$$H(M) = H_L.$$

Security of hash function depend on the following parameters:

3.1. Length of hash code

Length of hash code n belongs to basic security parameter of hash function. Similarly as length of key within symmetric ciphers, the hash code must be resistant again brute force attack (i. e. computing of all combination of hash code). Length of hash code must guarantee, so that two different messages do not have the same code, i. e. hash function is collision resistance. With the use of hash code of length n we can generate 2^n of different messages. In praxis, from the reason of birthday attack [4], the numbers of these messages are $2^{n/2}$ only. The birthday attack is the mathematical justification what determines the minimum value of k such that the probability is greater than 0,5 that at least two different messages have the same hash code. For a code of length n, the level of effort required is proportional to the following number of combinations (see Tab. 2).

Tab. 2. required length of hash code .

Required property	Number of combinations
One-way	2^n
Weak collision resistance	2^n
Strong collision resistance	$2^{n/2}$

In present length of hash code over 160 bits is considered to computationally security.

The brute force attack on a MAC depend on the relatively size of the key K and of the length of MAC code n. There are two lines of attack possible: attack on the key space and attack on the MAC value.

The level of effort for brute force attack on a MAC algorithm can be expressed as $\min(2^K, 2^n)$. It would appear reasonable to require that the key length and MAC length satisfy a relationship such as $\min(K,n) \geq N$, where N is in the range of 128 bits.

3.2. Design of compression function *f*

Compression function is the most important element of hash algorithm. If the compression function is collision resistance, then so is the resultant iterated hash function. Definition of compression function f is different for any types of hash functions. Secure compression function must be collision resistant. Mathematically is defined as impossibility to find the messages Mi and M_i' and suitable hash function H_{i-1}, for which is valid:

$$f(H_{i-1}, M_i) = f(H_{i-1}, M_i') \quad (2)$$

Cryptoanalysis on internal structure of f is based on attempts to find efficient techniques for producing collisions for a single execution of f. Once that is done, the attack must take account the fixed value of IV. The attack on f depends on exploiting its internal structure. Otherwise, the finding collision on a compression function belongs to weak collision, but it consider as safety defection. This is reason, mainly for transmission of very sensitive information, these types of hash function do not recommend. Collision on compression function was found e. g. on hash function of type MD4, MD5 [5].

3. 3. Design of hash function H

Secure hash functions are considered one-way function (it is invisible to determine from hash code h message M) and collision resistance (it is invisible assign for two different messages M and M' the same hash code h). The finding collisions again hash function is considered as minority safety defection and these types of hash functions are stopped for the use in praxis. Today we can regard as secure hash functions SHAx (Secure Hash Algorithm), developed by the National Institute of Standard and Technology NIST (for x = 256, 384 or 512). The candidate of secure hash function in Europe is hash function of type RIPEMDx, in which algorithm is based on function MD, which is safety support. Length of hash code over x=160 bits considers the level of high security. This was valuated within solution of European project NESSIE [6] too. Hash functions RIPEMD 160, 256, 320 were registered as fire ware of the association TeletrusT and they can use fee for commerce purposes.

A comparison of MD5, SHA-1and RIPEMD-160 is shown in Tab. 3. The algorithms are compared on the following parameters:

1. Length of digest message.
2. Basic unit of processing.
3. Number of steps.
4. Maximum message size.

Tab. 3. Comparison of MD5, SHA-1 and RIPEMD-160.

Parameter	MD5	SHA-1	RIPEMD-160
1	128 bits	160bits	160 bits
2	512 bits	512 bits	512 bits
3	4 rounds of 16	4 rounds of 80	5 paired rounds of 16
4	∞	2^{64}-1 bits	2^{64}-1 bits

3. 4. Speed of hash algorithms

For safety - related messages within railway applications the speed of hash mechanisms are very important parameter. This is follows from specification of transmission within railway applications. In transmission (typically cyclic) the receiver can check if the delay between two messages exceeds a predefined allowed maximum time. If this is the case, an error shell be assumed. In the tab. 4 results of non - key hash codes determination via programme simulations can shown. The value of the hash codes were produced from the message of size 344MB with the used PC Pentium AMD Athlon Processor, 256 MB RAM. The test was aimed on determination of speed of hashing for six type of non- key hash algorithm as it is shown in Tab. 4.

Tab. 4. Result of programme simulation.

Hash function	**Hash code**[bits]	**Time of hashing** [s]	**Speed of hashing** [Mb/s]
CRC-16	16	34,92	9,86
CRC-32	32	35,35	9,73
MD4	128	35,44	9,72
MD5	128	35,38	9,73
SHA1	160	36,78	9,36
RIPEMD128	128	38,15	9,03
RIPEMD160	160	38,26	9,00
RIPEMD256	256	37,99	9,07
RIMEMD320	320	39,78	8,66
Haval128	128	36,91	9,33
Haval160	160	37,10	8,97
Haval224	224	38,40	8,97
Haval256	256	39,63	8,69
Tiger192	192	42,90	8,03

Note: CRC-r does not considered as a hash function. In the table it is illustrated for comparison only.

4. CONCLUSION

In the paper was present the specifications of hash mechanisms, which can be used as secure tools for keeping integrity and authentication of safety - related messages defined in norm STN EN 50159-2.

For selection of hash algorithms into model representation of message - type B1 is necessary take into account the specification of transmission within railway communications subsystem and evolution of cryptography mechanisms. Choosing of hash mechanisms must be aimed at modern, computationally secure algorithms, today.

In the paper parameters of hash function are analysed, which are very important for security of algorithms (lengths of has code, compression function and properties of hash function as one - way and collisions resistant). For application with time valid of message the speed of hash algorithms is important parameter too. On the tab. 4 are illustrated results of time parameters evaluation via programme simulation for six types of non - key hash algorithms. The results are depend on internal structure of hash algorithms (number of steps, lengths of digest, type of compression function as so on) and on parameter of PC.

Today it is regard as secure hash functions SHA256, SHA384, SHA512 or RIPEMDx (for x over 160 bits), which are standardised. The lengths of MAC code must be minimal 128 bits. One of most widely used MAC for data authentication is based on AES (Advance Encryption Standard) in CBC (Cipher Block Chaining) mode. In the resent years, there has been increased interest in developing a MAC derived from a cryptographic hash function.

The work has partially been supported by the grant Agency of Slovak republic VEGA, grant No. 1/1044/04 „Theoretical Foundations for Implementations e-Safety Principles into Intelligent Transportation Systems"

REFERENCES

[1] EN 50159-2 Railway applications, signalling and processing systems. Part 2: Safety related communications in open transmission , CENELEC, 2001

[2] MENEZES, A., van OORSCHOT-VASTONE, S.: Handbook for Applied Cryptography, CRC Press, 1996

[3] EN 501129: Railway applications Safety related electronic systems, CENECEC, 2002

[4] STALLING, W.: Cryptography and Network Security, Prentise Hall. New Jersey, 2003

[5] DOBBERTIN, H.: The status of MD5 after a resent attack, Cryptobytes, summer, 1996

[6] http://www.cosic.esat.kuleuven.ac.be./nessie/