# Topological methods for hash algorithms efficiency increase

**Dr. N.G.Bardis**
Department of Computer Sciences
Military Institutes of University Education
Hellenic Army Academy
Vari, 16673 Attiki, Greece

**Dr. A.Polymenopoulos,**
Senior Adviser
Hellenic Ministry of Public Order
Office of the Deputy Minister
P. Kanellopoulou 4, Athens, Greece

**Dr. A.P.Leros**
Department of Computer Sciences
Military Institutes of University Education
Hellenic Army Academy
Vari, 16673 Attiki, Greece

**N.Karadimas, MSc**
Department of Computer Sciences
Military Institutes of University Education
Hellenic Army Academy
Vari, 16673 Attiki, Greece

**Dr. N.E.Mastorakis**
Military Institutes of University Education, Hellenic Naval Academy,
Terma Hatzikyriakou, 18539, Piraeus, Greece

**Abstract**:- The efficiency increase of hash-algorithms based on cipher-block chains is considered. Scripts for breaking known hash-algorithms have been analyzed. Directions on designing hash-algorithm's topology with increased crypto-resistance and parallel processing ability of cipher-blocks at hash-signature formation are put forward. A hash-algorithm structure based on cipher-blocks with bidirectional communication topology between them is suggested. It is shown that the structure suggested has an increased resistance to breaks and has the ability of parallel calculation of hash-signatures. The results of comparative analysis of the major characteristics of the hash-algorithm suggested structure are presented.
*Keywords:* hash signature, cipher block, parallel processing

## 1 Introduction

The significance of ensuring the integrity and authenticity of information, transmitted over open networks, grows as the process of informational integration becomes widespread. Currently, the integrity of digital messages is assured by applying the digital signature and the Message Authentication Codes (MAC). The key element in both the digital signature mechanism and MAC is the algorithm of hash-signature formation, i.e. the hash-algorithm.

The efficiency of ensuring the integrity and authenticity of information is determined on one hand by the level of information security from any unauthorized alterations and on the other hand by the speed the digital signature or MAC is formed [2]. In so doing, the level of security is estimated through the expenditure of computational resources necessary to form a false message that cannot be detected by the security system. The security level required is determined by the special features used for protecting the message's integrity. These features are based on the fact that the cost of the resource expenditure for break should always exceed the cost of destroying the information message integrity. In practice, achievement of a high security level is always related to the increase of computational resources spent on security realization.

This dictates a necessity for the existence of either several hash-algorithms that differ on the ratio of the two efficiency criteria mentioned above or of a multipurpose algorithm with the flexibility to alter the above criteria values depending on the specific application.

From a structural aspect, the hash-algorithms utilized in practice may be divided into two groups: specific algorithms (SHA, MD-5, RIPEMD-160) and hash-algorithms designed on standardized cipher-block chains.

The hash-algorithms designed on standardized cipher blocks chains and applied in the algorithms of symmetrical cryptographic encoding (cipher-blocks DES, GOST 28147-89, IDEA, MARS, RIJENDAEL) have a number of advantages when compared to recursive hash-algorithms. It should be pointed out that this class of hash-algorithms can be used efficiently on commercial hardware for the realization of standardized cipher-blocks either in the form of specialized VLSI (only for the DES, the industry of developed countries commercialized more than 20 types of different chips [5,6]) or on the cryptographic co-processor IBM 4758. In this case the cryptographic processing speed, when compared to the programmed realization, is 2-3 times higher. The fact that standardized algorithms have been fully approved and thoroughly investigated is a warrant to their high cryptoresistance.

Using hash-algorithms that are based on the same cipher-blocks applied in the algorithms of symmetrical data encoding allows the unification and simplification of software used for cryptographic security in computational systems and networks, and in the authentication of the message sender. That is why, in practice, along with special hash-algorithms, the use of hash-algorithms based on standard cipher-blocks is widespread as well.
These factors require detailed analysis for the possibility of increasing the efficiency of the hash-algorithms based on cipher-block chains both from a security level aspect and in computation process arrangement utilizing software and hardware

## 2 Analysis of hash-signature break schemes and problem statement

To raise a hash-algorithms resistance to breaks, an analysis of methods for finding colliding messages should be carried out. In practice, to find a colliding message, the primary message (M) and its hash-signature H(M) are given as the initial information [1]. Since the hash-algorithm in most cases is not a secrete one, the party carrying out the cryptanalysis knows all the values of the intermediate hash-signatures. The problem of break consists of finding one or several imitative messages M' such, that the following condition is held:

$$M' \neq M, H(M') = H(M) \qquad (1)$$

As it was mentioned above, in a one-way hash-function that transforms an informational message M of any length into a fixed length code H(M) of the hash-signature, it must be computationally difficult to find a colliding message M'. In this case the notion of difficulty depends on special requirements that a situation dictates. However for most practical applications the computational difficulty supposes executing about $2^n$ operations [3], where n is the digit capacity of the hash-signature H(M) being formed by the hash-algorithm.

The cipher-block generates the output of n-bit informational code C with respect to the given input code M of the same capacity and to the k-bit code of the K-key (for DES n=64, k=56): C=F(M,K). The main problem in ensuring the hash-signature's cryptoresistance when using cipher-blocks is in the fact that the latter have the property of reversibility: the input code M may be easily calculated if the K-key code and the output information code C: M=F(C,K) are given.

Since a single cipher-block performs the one-to-one transform of the input code into the output one, if the key is given, i.e. there is not such an M'≠M, at M given, that F(M',K)=F(M,K), therefore, to break hash-signatures built on the basis of a cipher-blocks chain, its necessary to insert alterations into at least two informational blocks. In this case two basic schemes of the indicated blocks are possible.

In the first one, the data channel is used for generating and transferring the signature code, while the key input is used for setting the code of the k-bit block of the informational message i.e. generation of the hash-signature H(M) is carried out according to the following formula:

$$H_j = F(H_{j-1}, M_j), j = 1,...,t, H_o = IV \qquad (2)$$

where IV is the inducing vector, t is the number of m–bit blocks in the total informational message. Inserting or substituting two or more adjacent blocks when the codes Hj-1, Hj+1 are known performs the break according to the following scheme:
An arbitrary code $M'_j \neq M_j$ is set and a distorted version of the intermediate code of the hash-signature is calculated $H'_j$: $H'_j = F(H_{j-1}, M'_j)$.
The break of the (j+1)-th cipher-block is performed, that is, $M'_{j+1}$ is found by means of total searching with the use of linear or differential cryptanalysis that the following condition is held: $H_{j+1} = F(H'_j, M'_{j+1})$.

In this way, the hash-signature break is reduced to the classic break of a cipher-block that requires not more than $2^k$ attempts. It should be pointed out that the blocks do not need to be adjacent - there may be any number of blocks between them. In the most extreme case the whole message is replaced and the initiating vector IV acts as the code $H_{j-1}$ and the code $H_t$ of the message resulting hash-signature acts as the code $H_{j+1}$. In so doing, only the last block is subject to the break.

In the second basic scheme, the information input of the cipher-block is used to set the blocks of the informational message and the subset bits of the hash-signature obtained, before is used as the key. That is, the hash-transformation is carried out according to the following formula:

$$H_j = F(M_j, H_{j-1}), j = 1,...,t, H_o = IV \qquad (3)$$

The break of this cipher-block chain scheme, under the condition that the codes are given, is performed in the following order:
An arbitrary $M'_j$ is chosen and the relation $H'_j = F(M'_j, H_{j-1})$ is calculated.
The calculation of $M'_{j+1}$ is carried out using the known codes $H'_j$ and $H_{j+1}$: $M'_{j+1} = F^{-1}(H_{j+1}, H'_j)$ by means of the reversed transform for the (j+1)-th cipher-block.

Thus, the break of the second basic layout – topology of the hash-algorithm may be carried out in real time and rather simply by the reversibility property of the cipher-blocks.

The simplest way to provide for cipher-block irreversibility is to XOR the input code with the output one: $C = M \oplus F(M, K)$. It should be taken into account that practically all the cipher-blocks used nowadays perform, in cryptographic transforms, XOR of the binary sequence being encoded with a pseudo-random bit string that depends on the K-key code and the string itself, i.e. $F(M, K) = M \oplus \phi(M, K)$, where function $\phi$(M,K) is irreversible by definition. In this case the result of the XOR considered is, in fact, recovery of the primary function $\phi$(M,K) of irreversible pseudo-random code generation: $C = M \oplus F(M, K) = \phi(M, K)$.

Break of this scheme is performed in the following way:
An arbitrary meaning of the code of the j–th information block $M'_j$ is chosen. Knowing $H_{j-1}$, the code $F(M'_j, H_{j-1})$ on the output of the j-th cipher-block is calculated, then XORed with the code $M'_j$: $H'_{j+1} = M'_j \oplus F(M'_j, H_{j-1})$ and used as the key for the (j+1)-th cipher-block.
The break of the (j+1)-th cipher-block is carried out by searching the codes $M'_{j+1}$ that –check-- the equality:

$$H'_{j+1} = M'_{j+1} \oplus F(M'_{j+1}, H'_j) = H_{j+1} = M_{j+1} \oplus F(M_{j+1}, H_j) \qquad (4)$$

Thus, it has been shown that the expenditure of resources necessary for the break of a cipher-block determine the computational resources spent on breaking a hash-signature in a hash-algorithm with the considered layout.

The analysis of different hash-algorithms revealed that at breaks, when the primary informational message M as well as all the intermediate results are known, it is impossible to create a situation in which the party performing the cryptanalysis has no knowledge about both the key and the output information block (it would take $2^{m+k}$ attempts of break). The only possible situation is where an unknown element is either the key or the pair of the input and output data at the unknown key. This refers in full measure to the structures of hash-algorithms based of the standardized cipher-blocks with transverse communications.

The common shortcoming of all the known hash-algorithms based both on cipher-blocks and in recursive ones, that significantly

decreases the efficiency of their practical application in the systems of information security protection, is their strictly sequential character of hash-signature calculation since at hash-signature calculation on the j–th cipher-block, the $H_{j-1}$ hash-signature is used that was obtained on the preceding (j-1)-th step. This makes it impossible to arrange the parallel calculation of hash-signatures and restricts the speed at which the information message in checked for integrity. Besides, with a limited number of cipher-block inputs, utilization of the structures with the end-to-end communication of the intermediate hash-signatures does not seem possible to effectively provide for setting the inputs for the private key in the MAC systems. Finally, employment of the end-to-end schemes of hash-signature generation strictly limits the capacity of the hash-signature being formed by the capacity of the cipher-block outputs. That is why two rows of cipher-blocks are used in the algorithms with increased capacity such as MDC-2 and MDC-4. Thus it has been shown that making use of hash-algorithm on the basis of the standardized cipher-blocks of communication layout with the end-to-end carry of the codes of the intermediate hash-signatures does not allow the problem of information integrity to be efficiently solved in open systems and networks

## 3 Development of hash algorithms layout with increased cryptoresistance and possibility of parallel processing

The analysis carried out has revealed that, from the aspect of both their cryptoresistance and speed, increasing the efficiency of hash-algorithms based on standardized cipher-block chains, improvement should come by:

Organization, in the structures of interblock connections, of the standardized cipher-blocks of more wide binding of cipher-block work rather than one-way communication that allows for localizing the break by searching one block. The suggested arrangement requires instead much more computational resources for break realization. In the simplest example, such an arrangement may be realized in the form of the bilateral communication of the cipher-blocks.

Alteration of the cipher-blocks communications topology with the departure from the end-to-end structure to the parallel one means each cipher-block generates the code of partial hash-signature irrespective of the results of any other cipher-block operating. In so doing the mutual dependence in cipher-blocks operation is to be realized only on the level of the primary information i.e. the data supplied to the informational inputs of one cipher-block is to be realized only on the level of the primary information. In other words, the data supplied on the informational inputs of one cipher-block may be fed on the key inputs of another cipher-block;

Developing the hash-signature resulting code is to be carried out on terms of binding the partial signatures. Furthermore, the binding functions must be non symmetrical and irreversible.

In implementing of stated concept of cipher-block hash-algorithms efficiency at the cost of transition from the layout - topology of the end-to-end communication to the layout of communication on the level of the primary data, the following hash-algorithm structure is suggested and presented in Fig. 1.

The code of block $M_j$ of informational message is supplied on the information inputs of each j–th cipher-block while the key inputs of the j–th cipher-block are supplied with the concatenation of the subsets of bits of the adjacent (j-1)-th and (j+1)-th informational blocks $M_{j-1}=\{m_{j-1}^0,...m_{j-1}^n\}$, $M_{j+1}=\{m_{j+1}^0,...,m_{j+1}^n\}$:

$K_j = m_{j-1}^0,...,m_{j-1}^{k/2-1} \| m_{j+1}^0,...,m_{j+1}^{k/2-1}$ (k is the bit capacity of the key), so the resulting hash-signature H(M) code is calculated in the form:

$$h_1 = M_1 \oplus F(M_1, m_t^0,...,m_t^{k/2-1} \| m_2^0,...,m_2^{k/2-1})$$

$$h_t = M_t \oplus F(M_t, m_{t-1}^0,...,m_{t-1}^{k/2-1} \| m_1^0,...,m_1^{k/2-1})$$

$$h_j = M_j \oplus F(M_j, m_{j-1}^0,...,m_{j-1}^{k/2-1} \| m_{j+1}^0,...,m_{j+1}^{k/2-1}),$$

$$j = 2,...,t-1$$

$$H(M) = \bigoplus_{i=1}^{t} h_i \qquad (5)$$

Using the suggested layout – topology of the cipher-blocks connections, the break is fulfilled by simultaneous fitting of all the informational blocks. The average number of trials necessary for hash-signature break by fitting (i.e. by finding a colliding one relative to the given hash-signature), is determined by its bit capacity m and is equal to $2^m$ . However, each trial requires t calculations of the resulting cipher-block code, so that the total number of

calculations performed by the cipher-block at a break is $2^m \cdot t$. If the length of an information message is large, the number is far larger compared against the known algorithms of the indicated class.

Also, the layout - topology allows for implementing the parallel calculation of partial hash-signatures.
The comparative characteristics of the suggested hash-algorithm structure on the basis of cipher-blocks and of the known hash-algorithms are presented in Table 1. As it can be seen from the data presented in Table 1, the hash-algorithm suggested based on cipher-blocks at the bit capacity of the cipher-block field above the key bit capacity ( $m \geq k$), does not rank below the known hash-algorithms in the calculation speed of the hash-signature. If the number of blocks in the information message being processed exceeds 4, then the algorithm suggested is not second by the level of cryptoresistance to any one of the known hash-algorithms based on cipher-block chains.

If we determine criterion E of an algorithm as the ratio of break time estimation to the hash-signature formation time, then the value of the criterion for the suggested hash-algorithm equals $2^m$ , while for most known algorithms the mentioned criterion value is equal to $2^m \cdot m/l$. This can be seen for the algorithms presented in Table 1 such as MDC-2, MDC-4 and for the linear algorithm with calculation of an intermediate hash-signature in the form of $H_i = M_i \oplus F(M_i, H_{i-1})$ . So, by the adopted efficiency criterion, the suggested hash-algorithm exceeds the known algorithms presented in Table 1 by q times. In this case the numeral value of q is determined by the ratio of the length l of the informational message in bytes to the length m of the informational block of the cipher-block q=l/m.

Thus, the efficiency of the hash-algorithm suggested in practice is always higher than that of the known hash-algorithms of the class under consideration and with growth of the message length the difference in the efficiency increases. There are versions of the structure suggested in which the concatenation operation is replaced by operation of XOR between code fragments of adjacent informational blocks and supplying the result of the XOR on the cipher-block key input. The other version replaces XOR by the arithmetic summation.

The significant advantage of the suggested structure is that it enables the hash-signature's bit capacity to be changed; correspondingly with the capacity increase the complexity of break also increases.

The structure suggested is resistant to break by means of block manipulation. The resistance is achieved at the cost of interblock communication layout -topology. Correspondingly, to generate resulting hash-signatures from partial ones, extremely simple-including linear- transformations may be used, which provides for high speed of interblock processing at parallel calculation of cipher-blocks

## Conclusions

Hash-algorithms on the basis of cipher-block chains posses a number of advantages compared to single-purpose hash-algorithms. However their cryptoresistance is lower and they do not allow for paralleling hash-signature calculation.

Analysis carried out of different scripts of known algorithm breaks has revealed that, by virtue of one-way communications in them the hash-signature break, it may always be reduced to the break of one cipher-block. Consequently, hash-algorithms cryptoresistance may be increased essentially at the expense of complication of cipher-blocks communication layout -topology. In so doing, structures of hash-algorithms that allow parallel processing may be obtained. The corresponding instructions – recommendations for hash-algorithms design on the cipher-block basis are worked out. A hash-algorithm structure has been suggested whose cryptoresistance is much higher compared to the known hash-algorithms schemes on the cipher-block basis. Employing simple functions for collating partial hash-signatures makes it possible to arrange parallel processing, that does not impose strict limits on hash-signature generation speed. The structure is oriented on application of cipher-blocks of the Rijendael type.
The recommendations formulated for designing interblock connections may be applied not only for hash-algorithms developed on the cipher-blocks, but also for schemes of overlapped generating hash-signatures of an informational message and its cipher signature with the sender's private key. In this case introduction of a complex non-unidirectional layout makes it possible to increase the resistance against break

of the sender private key and against the imitation of informational messages. Unlike the known sequential structures of hash-algorithms, the structures suggested provide for cipher-blocks parallel processing, i.e., possess potentially higher performance

## References


[1] Khudsen L., Lai X., Preneel B. “Attacks on fast double block length hash functions”. Journal of Cryptology, Vol.11. No.1, 1998, pp.59-72.

[2] Merkle R.C.,”One-Way Hash Functions and DES,” Advances in Cryptology-CRYPTO ’89 Proceeding, Berlin: Springer-Verlag, 1990, pp.428-446.

[3] Pieprzyk J., Sadeghiyan B. Design of Hashing Algorithms. LNCS 756. Berlin: Springer-Verlag, 1993, p. 194.

[4] Preneel B. “ Cryptographic hash functions”. European Transactions on Telecommunications, Vol. 5, 1994, pp.431-448.

[5] Alfred J. Menezes, Paul C. van Oorschot andScott A. Vanstone *Handbook of Applied Cryptography*by

[6] Schneier B. “Applied Cryptography. Protocols. Algorithms and Source codes in C. Ed.John Wiley, 1996 - 758 pp.


| Hash- Algorithm | Hash Signature Formation Time | Rank of Attacking Time | Possibility for Parallel Processing |
|---|---|---|---|
| $H_i = F(H_i, M_i)$ | $l \cdot T_b / k$ | $2^k \cdot T_b$ | - |
| $H_i = M_i \oplus$ $\oplus F(M_i, H_{i-1})$ | $l \cdot T_b / m$ | $2^m \cdot T_b$ | - |
| MDC-2 | $\frac{2 \cdot l \cdot T_b}{m}$ | $2^{m+1} \cdot T_b$ | - |
| MDC-4 | $\frac{4 \cdot l \cdot T_b}{m}$ | $2^{m+2} \cdot T_b$ | - |
| Suggested Hash Algorithm | $\frac{l \cdot T_b}{m}$ | $2^m \cdot T_b \cdot l / m$ | + |

**Table 1.**

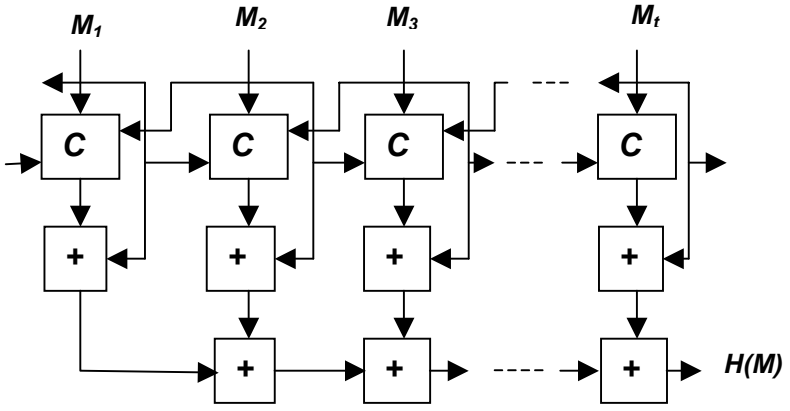


**Fig.1**