A New (*n*, *n*) Blockcipher Hash Function using Feistel Network: Apposite for RFID Security

Atsuko Miyaji and Mazumder Rashed

Japan Advanced Institute of Science and Technology, 1-1, Asahidai, Nomi-Shi, Ishikawa, Japan
{miyaji, rashed}@jaist.ac.jp

Abstract. In this paper, we proposed a new (n, n) double block length hash function using Feistel network which is suitable for providing security to the WSN (wireless sensor network) device or RFID tags. We use three calls of AES-128 (E_1, E_2, E_3) in a single blockcipher E' so that the efficiency rate is 0.33. Surprisingly we found that the security bound of this scheme is better than other famous (n, n) based blockcipher schemes such as MDC-2, MDC-4, MJH. The collision resistance (CR) and preimage resistance (PR) security bound are respectively by $O(2^n)$ and $O(2^{2n})$. We define our new scheme as JAIST according to our institute name.

Keywords: AES, deal cipher model, collision security, preimage security.

1 Introduction

A cryptographic hash function is a function which maps an input of arbitrary length to an output of fixed length. For any cryptographic hash at least collision resistance and preimage resistance properties should be satisfied.

Cryptographic hash function can be constructed by blockcipher or scratch. Due to adversarial successful attacks on MD4/5 and SHA-family [4, 5] blockcipher based hash functions gained popularity. Another issue is low cost requirement in respect of power and chip size. The AES module requires only a third of the chip area and half of the mean power in compare of SHA. Smaller hash functions like SHA-1, MD5 and MD4 are also less suitable for RFID tags than AES. Inclusively it can be said that the total power consumption of SHA-1 is about 10% higher than the AES [3]. Day by day the uses of RFID tags or WSN's devices has been increased rapidly. So it is a great challenge for the scientists to provide a scheme which can make balance between cost and security issues.

Blockcipher based hash functions are classified into single-block-length (SBL) and double-block-length (DBL). The output length of SBL hash function is equal to the block length and DBL hash function is the twice of block length. Due to birthday attack the collision resistance of hash function can be occurred with the time complexity $O(2^{l/2})$ (l: output length of hash function). So SBL hash function is no longer secure in terms of CR. From the following table 1. current research status of DBL has been found which categorizes into (n, n) and (n, $2n$) blockcipher. In CT-RSA-09, it is found that (n, n) based blockcipher hash function is 40% faster than (n, $2n$) blockcipher hash function [11]. For implementation purpose critical issue is to measure the cost of security under RFID tags or WSN's devices. According to [9], AES-128 is more user friendly because of less power consumption, less encryption and decryption time.

2 Related Work

At first we mention some famous (n, n) blockcipher hash function which are based on AES-128. MDC-2 and MDC-4 has been introduced in the late eighties by Bracht et al but their CR and PR security bound prove have been achieved in Africacrypt-2012 [23]. The CR and PR security bound result of MDC-2 and MDC-4 are respectively by

$\left(O\left(2^{3n/5}\right),O\left(2^{n}\right)\right)$ and $\left(O\left(2^{5n/8}\right),O\left(2^{5n/4}\right)\right)$. Another famous MJH hash function introduced in CT-RSA (2011). It's CR and PR bound is as $O\left(2^{n/2}\right)$ and $O\left(2^{n}\right)$. Two things are remarkable here such as no (*n*, *n*) based blockcipher hash function's CR and PR security bound are equal to $O\left(2^{n}\right)$ and $O\left(2^{2n}\right)$. In other hand if we follow the (*n*, 2*n*) based blockcipher then we can find that the security bound of AES-256 based hash function is better than AES-128 based hash function. Weimar-DM [6] double block length hash functions has been proposed in ACISP- 2012 where CR and PR bound is respectively $O\left(2^{n}\right)$ and $O\left(2^{2n}\right)$. Other famous two schemes of Abreast and Tandem-DM which were proposed Lai and Messy [12]. The CR and PR of Abreast-DM and Tandem-DM was being proved by Lee, Stam and Steinberger [14]. The CR and PR bound of these two schemes are respectively $O\left(2^{n}\right)$ and $O\left(2^{2n}\right)$. In FSE 2006, Hirose [15] proposed another famous construction and shoed that it was bound in $O\left(2^{n}\right)$ for the CR and $O\left(2^{n}\right)$ for the PR. Later this PR security bound has been improved by Lee, Stam and Steinberger [16].

Table 1. Different $(n,\ 2n)$ and $(n,\ n)$ based blockcipher result analysis [6–8]

Hash Type	Comp. Function	Eff. Rate	No. of *E* calls	Key Sch.	Coll. resistance	Pre. resistance
Weimar	$3n \rightarrow 2n$	1/2	2	2	$O\left(2^{n}\right)$	$O\left(2^{2n}\right)$
Hirose	$3n \rightarrow 2n$	1/2	2	1	$O\left(2^{n}\right)$	$O\left(2^{2n}\right)$
ISA-09	$4n \rightarrow 2n$	2/3	3	3	$O\left(2^{n}\right)$	-
MDC-2	$3n \rightarrow 2n$	1/2	2	2	$O\left(2^{n/2}\right)$	$O\left(2^{n}\right)$
MDC-4	$3n \rightarrow 2n$	1/4	4	1	$O\left(2^{5n/8}\right)$	$O\left(2^{5n/4}\right)$
MJH	$3n+c \rightarrow 2n$	1/2	2	1	$O\left(2^{n/2}\right)$	$O\left(2^{n}\right)$
JAIST (Proposed)	$3n \rightarrow 2n$	1/3	3	3	$O\left(2^{n}\right)$	$O\left(2^{2n}\right)$

3 Preliminaries

3.1 Ideal Cipher Model

A blockcipher is a keyed function $E:\{0,1\}^{k}\times\{0,1\}^{n}\rightarrow\{0,1\}^{n}$. For each $k\in\{0,1\}^{k}$ the function $E_{k}\left(\cdot\right)=E\left(k,\cdot\right)$ is a permutation on $\{0,1\}^{n}$. If *E* is a block cipher the E^{-1} denotes it's inverse, where $E_{k}\left(x\right)=y$ and $E_{k}\left(x\right)=y$ is called forward and backward query respectively. Assume that block (*k*, *n*) is the family of all blockciphers $E:\{0,1\}^{k}\times\{0,1\}^{n}\rightarrow\{0,1\}^{n}$. A blockcipher based hash function $H:\{0,1\}^{*}\rightarrow\{0,1\}^{l}$ where $E\in Block\left(k,n\right)$ used as round function. An adversary is given access to oracle E/E^{-1} and for the i^{th} query response q_{i}, adversary keeps the record. In the ICM model the complexity of an attack is measured by the total number of the optimal adversary's queries to the two oracles *E* and E^{-1}.

3.2 Security Definition

An adversary is a computationally unbounded but always-halting collision-finding algorithm *A* with resource-bounded access to an oracle $E \in Block(k,n)$ that means in the collision resistance experiment, a computationally unbounded adversary *A* is given oracle access to a blockcipher *E*. It is allowed that, *A* can make query to a both *E* and E^{-1}.

Definition 1. Collision resistance of a compression function: The adversary *A* is given oracle access and *f* be a blockcipher based hash function, then the advantage of *A* to find collisions in *f* is:

$$\Pr\left[\begin{array}{l} E \leftarrow B(k,n);(h,g,m),(h',g',m') \leftarrow A^{E,E^{-1}} : (h,g,m) \neq (h',g',m') \\ \wedge f^{E}(h,g,m) = f^{E}(h',g',m') \vee f^{E}(h,g,m) = (h_0,g_0) \end{array}\right]$$

For $q \geq 1$, it can expressed that, $Adv_f^{COMP}(q) = \max_A \left\{Adv_f^{COMP}\right\}$ maximum is taken over all adversaries which can query at best *q* oracle queries.

Definition 2. Preimage resistance of a compression function: The adversary *A* is given oracle access to a block cipher $E \in Block(k,x)$ and *f* be blockcipher based hash function. Adversary *A* arbitrary selects a value of (h',g') before making any query to oracle either *E* or E^{-1}. Then the advantage of *A* is to find preimage in *f* such as $Adv_f^{pre}(A) = \Pr\left[H(g,h,m) = (h',g')\right]$.

For $q \geq 1$, $Adv_f^{pre}(q) = \max_A \left\{Adv_f^{pre}\right\}$ where the maximum is taken over all adversaries which can query at best *q* oracle queries.

4 A New (*n*, *n*) Double Block Length Hash Function

In this section, a new (*n*, *n*) double block length hash function has been discussed with diagram which is defined as JAIST scheme according to our institute name. This scheme is created based on Feistel network which is shown in Fig. 1(b). In ISA-2009, there was another scheme based on Feistel network which actually inspires us. We should mention here the similarities and dissimilarities of these two schemes. Our scheme is based on (*n*, *n*) blockcipher where ISA-09 scheme is based on (*n*, 2*n*) blockcipher [23]. In our CR proof technique we used the idea of external and internal collision. Also we provide PR security proof according to Armknecht [16].

Definition 3. Let $E(k,n)$ be a blockcipher taking k: n bit key and n-bit block size such as $E:\{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n \quad (k=n)$. We define,
$E':\{0,1\}^k \times \{0,1\}^{2n} \rightarrow \{0,1\}^{2n}$ be a DBL cipher using Feistel network created by calling three independent blockcipher of *E*.
So $E':\{(a_0,a_1),m_i\} = (y \oplus a_1) \,||\, (x \oplus z \oplus a_0)$ where *x*, *y*, *z* is defines as:
$x : E_{a_1}(m_i),\ y : E_{a_0 \oplus x}(m_i),\ \mathrm{z} : E_{a_1 \oplus y}(m_i)$.

Definition 4. Let $F:\{0,1\}^n \times \{0,1\}^{2n} \rightarrow \{0,1\}^{2n}$ be a compression function. Then we replace the compression function by our defined blockcipher *E*' from definition 3 and Fig. 1(a). Such as, $f\left(h_{i-1}, m_i\right) = E'_a\left(m\right) \oplus c.$

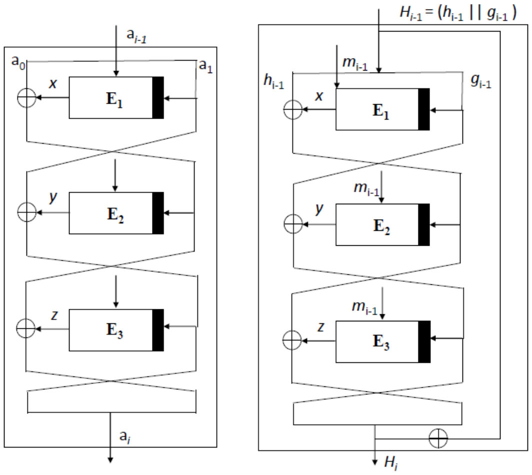

Fig. 1 (a) Compression function (Source: LNCS 5576, p.16), 1 (b) JAIST Scheme

5 Security Analysis of the JAIST Scheme

5.1 Collision Security Analysis

For any collision resistance finding experiment, a computationally unbounded adversary *A* is given oracle access to a blockcipher *E* uniformly sampled among all blockciphers of key length *n* and message length *n*. *A* is allowed to query both *E* and E^{-1}. After *q* queries to *E*, the query history of *A* is the set of triples *Q*= (X_i, K_i, Y_i) such that *E*(K_i, X_i) = Y_i and *i*th query of adversary *A* is either *E*(K_i, X_i) or E^{-1}(K_i, Y_i) for $1 \le i \le q.$ Assume that $Q = \left\{\left(X_i, Y_i, K_i\right)\right\}_{j=1}^{i}$ be the first *i* elements of the query history. Then it can be said that *A* succeeds or finds a collision after its first *i* queries if there exists distinct $\left(H_{i-1}, m_i\right), \left(H'_{i-1}, m'_i\right)$ such that,

$$H^{JAIST}\left(H_{i-1}, m_i\right) = H'^{JAIST}\left(H'_{i-1}, m'_i\right).$$

Theorem 1. Let H^{JAIST} be a double-length hash function composed of compression Function *F* specified in Def. 1. Then the advantage of an adversary in finding a collision in H^{JAIST} after *q* queries can be upper bounded by:

$$\frac{3q}{2\left(N-1\right)^2}$$

Proof

It is assumed that the adversary has made any relevant query to E or E^{-1} which can occur collision in the ideal cipher model. Another issue is the adversary never makes a query which is already available at his query database. In formal meaning, one can assume that the adversary never makes a query $E(K, X) = Y$ obtaining an answer *Y* and then makes the query $E^{-1}(K, Y) = X$ (which will necessarily be answered by *X*). At first consider, adversary *A* which is able to make an arbitrary *q*-query collision. Let *A* be a collision-finding algorithm of H^{JAIST} with oracles E, E^{-1}. *A* asks *q* pairs of queries to E, E^{-1} in total. Since, *h*' and *g*' depends both on the plaintext and ciphertext of E/E^{-1}. One of them is fixed by a query and other is determined by randomly from the query-database *Q*. As a result *h*', *g*' selected randomly from the query and query-oracle-database. It can be describe as like tabular form in the Table 2.

At first one of the important issue should be raised here, in our scheme we construct E' from three calls of E and then combine the result of $h_i \,||\, g_i = H_i$. So at first we find out the security of desire (E_1, E_2, E_3) of three calls of blockcipher under some conditions and assumptions. Under the following three cases we mentioned these.

Table 2. Sub case of $H_{i-1} \neq H'_{i-1} \wedge m_{i-1} = m'_{i-1}$

$h_{i-1} \neq h'_{i-1}, g_{i-1} = g'_{i-1}, m_{i-1} = m'_{i-1}$
$h_{i-1} = h'_{i-1}, g_{i-1} \neq g'_{i-1}, m_{i-1} = m'_{i-1}$
$h_{i-1} \neq h'_{i-1}, g_{i-1} \neq g'_{i-1}, m_{i-1} = m'_{i-1}$

External Collision

For every *j*, $\left[where,\ j \leq q\right]$, let C_j be the event that a colliding pair found for *F* with the j^{th} pair of queries. The event is as like $j' < j$:

$$y \oplus h_{i-1} \oplus g_{i-1} = y' \oplus h'_{i-1} \oplus g'_{i-1}$$
$$x \oplus z \oplus h_{i-1} \oplus g_{i-1} = x' \oplus z' \oplus h'_{i-1} \oplus g'_{i-1}$$

It implies that, $H_{i-1} \neq H'_{i-1} \wedge m_{i-1} = m'_{i-1}$. According to Table 2. described three cases can be occurred for collision resistance. So at first we need to find out this probability of these three cases. This is trivial and also it can be said that from the famous PGV [22] paper for any case the probability is as like $\left(1/2^n - 1\right)^2$. So if C_j be the event that *A* finds a collision pair of the compression function for *f* with the j^{th} pair of queries. Then three queries to the oracle E/E^{-1} are required to compute the output of the compression function for above conditions. It implies that $\Pr\left[C_j\right] \leq \left(j-3\right)\left(\dfrac{1}{2^n - 1}\right); \left(j \geq 3\right)$. Let *C* be the event that a pair is found for *F* with *q* pairs of queries then,

$$\Pr\left[C\right] = \Pr\left[C_3 \vee C_4 \vee C_5 \vee \ldots.. \vee C_q\right] = \left(\frac{1}{2^n - 1}\right)^2 . \sum_{j=3}^{q}\left(j-3\right)$$
$$= \left(\frac{1}{2^n - 1}\right)^2 . \frac{\left(q-2\right)\left(q-3\right)}{2} \qquad \text{(i)}$$

Internal Collision

This is actually for internal collision that means there is a probability to collide such as $\left(h_i = g_i\right) \Rightarrow y \oplus h_{i-1} \oplus g_{i-1} = x \oplus z \oplus h_{i-1} \oplus g_{i-1}$. Let *C* be the event that a pair are found or *F* with *q* pairs of queries, then,

$$\Pr\left[C\right] = \Pr\left[C_3 \vee C_4 \vee C_5 \vee \ldots.. \vee C_q\right] \;\; = \frac{\left(q-2\right)\left(q-3\right)}{2\left(2^n - 1\right)^2} \qquad \text{(ii)}$$

IV Collision

For $3 \leq j \leq q$, it is the event for collision occur with the IV value such as $\left(h_i = h_0, g_i = g_0\right) \Rightarrow \left(y \oplus h_{i-1} \oplus g_{i-1} = h_0, x \oplus z \oplus h_{i-1} \oplus g_{i-1} = g_0\right)$. Let *C* be the event that a pair are found for *F* with *q* pairs of queries, then

$$\Pr\left[C\right]=\Pr\left[C_3\vee C_4\vee C_5\vee .....\vee C_q\right]=\frac{\left(q-2\right)\left(q-3\right)}{2\left(2^n-1\right)^2} \quad \text{(iii)}$$

Take the result from equation (i), (ii) and (iii). Then finally it is shown that,

$$\left(\frac{1}{2^n-1}\right)^2.\frac{\left(q-2\right)\left(q-3\right)}{2}+\frac{\left(q-2\right)\left(q-3\right)}{2\left(2^n-1\right)^2}+\frac{\left(q-2\right)\left(q-3\right)}{2\left(2^n-1\right)^2}\le\frac{3q^2}{2\left(N-1\right)}$$

5.2 Preimage Security Analysis

Let *A* be an adversary that tries to find a preimage for its input σ which is randomly chosen by *A* before making any query to oracle. We follow a similar proof strategy of Armknecht and implementation strategy of Armknecht [16], when *A* selects its queries. The adversary A asks the conjugate queries in pair. Now adversary needs to bind the probability that i^{th} query pair leads to a preimage for σ where σ is defined as $\left(h' \| g'\right)\in\sigma$. So findings is that to calculate the probability that in *q* queries the adversary finds a point $\left(\sigma\right)$, such that $H^{JAIST}\left(h,g,m\right)=\left\{\left(h' \| g'\right)\right\}$.

Theorem 2. Let H^{JAIST} be a double-length compression Function $\left(E\in block\left(n,n\right)\right)$. Then the advantage of an adversary in finding a preimage in H^{JAIST} after *q* queries can be upper bounded by

$$=\frac{16q}{2^{2n}}$$

Proof

According to definition of adjacent query pair [16], the adversary *B* maintains an adversary query database *Q* in the form of $y\oplus h_{i-1}\oplus g_{i-1}, x\oplus z\oplus h_{i-1}\oplus g_{i-1}$ which has been run by adversary *A*. This is called adjacent query pair. Now need to make and implement super query. It implies that the query contains exactly $N/2$ queries with the same key, all remaining queries under this key are given for free to the adversary. Now an adjacent query pair $y\oplus h_{i-1}\oplus g_{i-1}, x\oplus z\oplus h_{i-1}\oplus g_{i-1}$ can be succeed iff,

$$y\oplus h_{i-1}\oplus g_{i-1}=y'\oplus h'_{i-1}\oplus g'_{i-1}$$
$$x\oplus z\oplus h_{i-1}\oplus g_{i-1}=x'\oplus z'\oplus h'_{i-1}\oplus g'_{i-1}$$

Thus the adversary obtains a preimage of $\left\{\left(h' \| g'\right)\in\left\{0,1\right\}^{2n}\right\}\in\sigma$ in particularly if it attains a winning query pair. It can be occurred by any of the following way such as NormalQueryWin(Q) and SuperQueryWin(Q).

Case 1: *Probability* of NormalQueryWin(Q)

Adversary *B* which has been called by adversary *A*, can make forward or backward query. Under this section, the goal is to find out the NormalQueryWin(Q). According to super query and adjacent query pair [16] the fresh value of h_i / g_i could be found in the following two ways.

– *Sub-Case* 1.1 The adversary *B* can make forward or backward query. Assume adversary makes a forward query, where at most $\left(2^n\!/_2-1\right)$ queries could be answered previously and earlier for adjacent query it could be answered at most $\left({2^n}\big/{2}-1\right)$ queries. Otherwise super query can be occurred. So the value of h_i and g_i comes uniformly and independently from the set size ${2^n}\big/{2}$. So probability forms as $\left({2}\big/{2^n}\big/2\right)$.

– *Sub-Case* 1.2 If $y\oplus h_{i-1}\oplus g_{i-1}=h_i$ then there is a probability for the free query (part of adjacent query pair) to return from the set size $\left({2^n}\big/{2}+1\right)$. So probability could be $\left(1\big/2^n\big/2\right)=2\big/2^n$.

So desired probability of NormalQueryWin(Q) is $8\big/2^{2n}$ $\quad$ (iv) .

Case 2: Probability of SuperQueryWin(Q)

In this section the target is to find out the probability of Super query. As for example under the keys, the value of h'/g' already have been known on exactly ${2^n}\big/{2}$ points. So from the definition of super query and adjacent query pair [16] if any pair of the query is the part of super query then the corresponding others query must be the member of the super query domain. From the above discussed points, it can be said that, probability of any query of any blockcipher is either 0 or $2\big/2^n$. Now the question how it can be found. The probability will be 0 if the h' is not in the range of super query that means it is available in the domain of normal query. Conversely it is assumed that due to super query the result comes from the set size $2^n\big/2$, so probability is $2\big/2^n$. For the adjacent query pair following cases can be happened:

$$y\oplus h_{i-1}\oplus g_{i-1}=y'\oplus h'_{i-1}\oplus g'_{i-1},\ x\oplus z\oplus h_{i-1}\oplus g_{i-1}=x'\oplus z'\oplus h'_{i-1}\oplus g'_{i-1}$$
$$y\oplus h_{i-1}\oplus g_{i-1}=x'\oplus z'\oplus h'_{i-1}\oplus g'_{i-1},\ x\oplus z\oplus h_{i-1}\oplus g_{i-1}=y'\oplus h'_{i-1}\oplus g'_{i-1}$$

– Sub-Case 2.1 For the above first condition, the answer will come from the set size $2^n\big/2.$ So the probability would be $2\big/2^n.$ As well as the probability for the second equation is equal to $2\big/2^n.$ So total probability of sub-Case 2.1 looks like $\left(2\big/2^n\right)^2$.

– Sub-Case 2.2 As like same explanation of sub-Case 2.1, the total probability of sub-Case 2.2 is $\left(2\big/2^n\right)^2$.

Now, analysis the probability of case-1 and case-2 and point that the cost of super query occurs is $2^n\big/2$. Another important factor is that the probability of super query occurs, which is at most $q\big/2^n\big/2.$ It implies that, Pr[SuperQueryWin (Q)]:

$$\leq {q}\Big/{\left(2^n/2\right)}\times\left(2^n/2\right)\times 2\times\left(\frac{2}{2^n}\right)^2=\frac{8}{2^{2n}} \quad \text{(v)}$$

Taking the value of equation (iv) and (v), we got the final probability of PR security bound is $16q\big/2^{2n}$.

6 Conclusion

In this article, a new scheme of DBL hash function has been proposed which is based on (n, n) blockcipher. The CR and PR security bound are respectively $O(2^n)$ and $O(2^{2n})$. The result of this construction is better than existing other (n, n) based blockcipher hash function and also this scheme is suitable for providing security to RFID tags/ WSN's because of faster operation of AES-128. In our proof technique we used ICM method which is widely known. But in real life AES does not behave ideally. So there is an open problem to introduce weak cipher model for the security proof. Our scheme's key schedule is more than one which is not cost effective. So there is another challenge to propose a new scheme which obtains single KS and as well as better security bound.

References

1. Bogdanov A., Leander G., Paar C., Poschmann A., Robshaw M. J. B., Seurin Y. : Hash Functions and RFID Tags: Mind the Gap LNCS, CHES, vol. 5154 (2008) 283-299.

2. Menezes A. J., Oorschot P. C., Vanstone S. A. : Handbook of Applied Cryptography, 5th ed, CRC Press, (2001).

3. Kaps J. P., Sunar B. : Energy Comparison of AES and SHA-1 for Ubiquitous Computing. LNCS, Emerging Directions in Embedded and Ubiquitous Computing, vol. 4097 (2006) 372-381.

4. Wang X., Lai X., Feng D., Chen H., Yu X. : Cryptanalysis of the Hash Functions MD4 and RIPEMD," LNCS, EUROCRYPT, vol. 3494 (2005) 1-18.

5. Wang X., Lai X., Yu X. : Finding Collisions in the Full SHA-1. LNCS. CRYPTO. vol. 3621 (2005) 17-36.

6. Fleischmann E., Forler C., Lucks S., Wenzel J. : Weimar-DM: A Highly Secure Double Length Compression Function, LNCS, ACISP, vol. 7372 (2012) 152-165.

7. Mennink B. : Optimal Collision Security in Double Block Length Hashing with Single Length Key," LNCS, ASIACRYPT, vol. 7658 (2012) 526-543.

8. Knudsen L., Preneel B. : Fast and Secure Hashing Based on Codes. LNCS, CRYPTO, vol. 1294 (1997) 485-498.

9. Lee J., Kapitanova K., Son S. H.: The price of security in wireless sensor networks. ELSEVIER, Computer Network, vol. 54, no. 17 (December 2010) 2967-2978.

10. Ozen O., Stam M. : Another Glance at Double-Length Hashing. LNCS. Cryptography and Coding, vol. 5291 (2009) 176-201.

11. Lee J., Stam M.,: MJH: A Faster Alternative to MDC-2. LNCS, CT-RSA, vol. 6558 (2011) 213-236.

12. Lai X., Massey X. : Hash function based on block ciphers. LNCS, EUROCRYPT, vol. 658 (1993) 55-70.

13. Lee J., Kwon D. : The Security of Abreast-DM in the Ideal Cipher Model," IEICE Transactions, vol. 94-A(1) (2011) 104-109.

14. Lee J., Stam M., Steinberger J. : The Collision Security of Tandem-DM in the Ideal Cipher Model," LNCS, CRYPTO, vol. 6841 (2011) 561-577.

15. Hirose S., "Some Plausible Constructions of Double-Block-Length Hash Functions," LNCS, FSE, vol. 4047, pp. 210-225, 2006.

16. Armknecht F., Fleischmann E., Krause M., Lee J., Stam M., Steinberger J. : The Preimage Security of Double-Block-Length Compression Functions. LNCS. ASIACRYPT, vol. 7073 (2011) 233-251.

17. Fleischmann E., Forler C., Gorski M., Lucks S. : Collision Resistant Double-Length Hashing. LNCS, PROVSEC, vol. 6402 (2010) 102-118.

18. Mennink B. : Optimal Collision Security in Double Block Length Hashing with Single Length Key. LNCS, ASIACRYPT, vol. 7658 (2012) 526-543.

19. Knudsen L., Preneel B. : Fast and secure hashing based on codes. LNCS. CRYPTO. vol. 1294 (1997) 485-498.

20. Black J. A., Rogaway P., Shrimpton T. : Black-Box Analysis of the Block-Cipher-Based Hash-Function Constructions from PGV. LNCS. CRYPTO, vol. 2442 (2002) 320-335.

21. Shannon C. E. : Communication Theory of Secrecy Systems. Bell Systems Technical Journal, vol. 128-4 (1949) 656-715.

22. Black J. A., Rogaway P., Shrimpton T., Stam M. : An Analysis of the Blockcipher-Based Hash Functions from PGV," LNCS, J.CRYPTOL, vol. 23 (2010) 519-545.

23. Jesang L., Seokhie H., Jaechul S., Haeryong P. : A New Double-Block-Length Hash Function Using Feistel Structure," LNCS, ISA, vol. 5576 (2009) 11-20.