An Efficient Construction of a Compression Function for Cryptographic Hash

Rashed Mazumder, Atsuko Miyaji, Chunhua Su

An Efficient Construction of a Compression Function for Cryptographic Hash*

Rashed Mazumder[2], Atsuko Miyaji[1,2,3] and Chunhua Su[1] **

[1] Graduate School of Engineering, Osaka University
{miyaji,su}@comm.eng.osaka-u.ac.jp
[2] Japan Advanced Institute of Science and Technology
{miyaji,s1420213}@jaist.ac.jp
[3] Japan Science and Technology Agency (JST) CREST

Abstract. A cryptographic hash (CH) is an algorithm that invokes an arbitrary domain of the message and returns fixed size of an output. The numbers of application of cryptographic hash are enormous such as message integrity, password verification, and pseudorandom generation. Furthermore, the CH is an efficient primitive of security solution for IoT-end devices, constrained devices, and RfID. The construction of the CH depends on a compression function, where the compression function is constructed through a scratch or blockcipher. Generally, the blockcipher based cryptographic hash is more applicable than the scratch based hash because of direct implementation of blockcipher rather than encryption function. Though there are many $(n, 2n)$ blockcipher based compression functions, but most of the prominent schemes such as MR, Weimar, Hirose, Tandem, Abreast, Nandi, and ISA09 are focused for rigorous security bound rather than efficiency. Therefore, a more efficient construction of blockcipher based compression function is proposed, where it provides higher efficiency-rate including a satisfactory collision security bound. The efficiency-rate (r) of the proposed scheme is $r \approx 1$. Furthermore, the collision security is bounded by $q = 2^{125.84}$ ($q =$ numer of query). Moreover, the proposed construction requires two calls of blockcipher under single iteration of encryption. Additionally, it has double key scheduling and it's operational mode is parallel.

Keywords: cryptographic hash, collision resistance, constrained device

1 Introduction

A cryptographic hash (CH) is defined as to proceed data from an arbitrary domain to a fixed domain [1, 2, 6–8]. The applications of CH are enormous. Generally, the CH is used in message verification, password verification, pseudorandom generation, and message authentication [1–3, 7]. Furthermore, the cryptographic hash is an efficient

* This work is partially supported by the Grant-in-Aid for Scientific Research (C)(15K00183) and (15K00189) and Japan Science and Technology Agency, CREST and Infrastructure Development for Promoting International S&T Cooperation
** JSPS Grant-in-Aid for Young Scientists (15K16005)

primitive of security solution for IoT-end device, RfID, and resource constrained device [35–39, 44]. Usually, the internal construction of CH depends on compression function [16, 17]. The compression function is based on scratch or blockcipher [6, 8, 16, 17, 31]. The blockcipher based compression function is a combination of component functions (Fig. 1). The component functions depend on the 16 modes of PGV construction so far [8, 16, 17]. Additionally, a classical structure of Merkle Damgrad is used for message encryption of the cryptographic hash, if message size is bigger than the blocksize [1–3]. According to Fig. 1, message (M) is multiple of blocklength. Hence, message is partitioned as $M|m_{i=1}||\ldots||m_l$. Thereafter, partitioned message injects as input with initial vector value (IV). The function F_i is called compression function, which is built by blockcipher or scratch. Usually, one of the PGV modes needs to select as a component function of compression function [8, 16, 17]. On the contrary, the generic of blockcipher compression function is more suitable than that of the scratch for encryption of a constrained device, IoT-end device because of implementation of blockcipher rather than the encryption function [6, 13, 14].

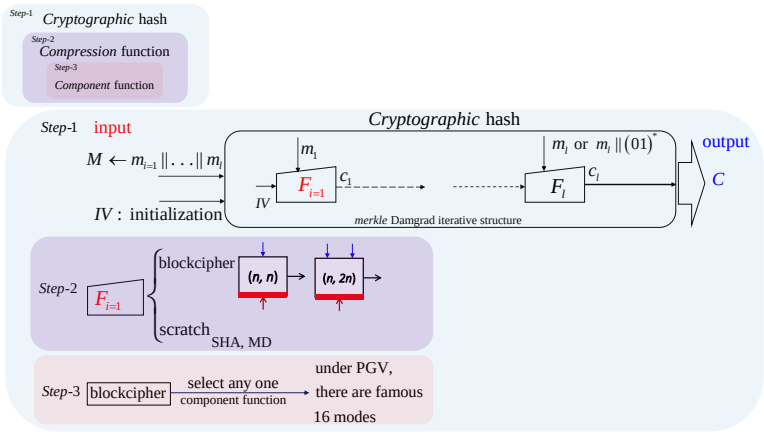

Fig. 1. Basic concept of cryptographic hash [2, 6, 8, 34]

Usually, the blockcipher based compression function is classified as single block-length $(\mathcal{SBL})$ and double block-length $(\mathcal{DBL})$. Due to short size of output, the application of $\mathcal{SBL}$ is limited now [2, 9, 33]. On the other hand, the $\mathcal{DBL}$ is more reliable construction due to its better resistance against birthday attack [2, 13, 16, 18, 21]. Moreover, the $\mathcal{DBL}$ is categorized as (n, n) and $(n, 2n)$ blockcipher (base is key size). The $(n, 2n)$ blockcipher is better due to upper security bound (larger key space) [6, 8, 13, 20, 23]. Generally, there are certain parameters that indicate the strength of blockcipher based compression function such as:

- security bound $(CR$: collision and PR : preimage resistance$)$
- efficiency-rate (r)
- number of calling blockcipher $(\#E)$

– key scheduling (KS)
– operational mode (OM)

The CR is defined as a game, where an adversary tries to find similar output under two different input, but the advantage of adversary is very limited [6, 13, 21]. Under PR, it is infeasible for adversary to find any m (message) such that $y = F(m)$, where y is predefined by the adversary [2, 6, 16]. The number of blockcipher $(\#E)$ depends on number of calling blockcipher per message-block encryption. The KS directs the number of key requirement for single message block encryption [16]. Furthermore, the OM stands for operational mode (parallel or serial) [17, 18]. In addition, the efficiency-rate [6, 15] is defined as:

$$r = \frac{\text{size of message block/per iteration}}{(\text{number of blockcipher call}) \times \text{block-length}}$$

Table 1. Result of existing familiar schemes

Name	CR	KS	r	$\#E$	OM
MR [23, 31]	$O(2^n)$	1	$1/2$	2	Parallel
Weimar [6]	$O(2^n)$	2	$1/2$	2	Parallel
Hirose [13]	$O(2^n)$	1	$1/2$	2	Parallel
Tandem [6, 14]	$O(2^n)$	2	$1/2$	2	Parallel
Abreast [6, 14]	$O(2^n)$	2	$1/2$	2	Parallel
Nandi [20]	$O(2^{\frac{2n}{3}})$	3	$2/3$	3	Serial
ISA09 [21]	$O(2^n)$	3	$2/3$	3	Serial

CR: Collision resistance, KS: Key Scheduling, r: Efficiency rate
$\#E$: Number of blockcipher calls, OM: Operational mode

Motivation. The parameters of CR, PR, r, $\#E$, OM, and KS are vital for any satisfactory scheme of blockcipher based compression function [1, 6–8, 13, 21]. Firstly, certain gaps are identified from the current familiar schemes based on the above parameters. Thus, the importance of the findings are shown in the field of efficient and secure communication. For example, the key scheduling cost is analysed in respect of construction of compression function. Usually, 176 bytes are needed for operating of single key scheduling [27]. Hence, minimization of key scheduling is a common practice. Additionally, the operation mode is very crucial for resource limited devices, where the parallel mode can provide maximum support in respect of memory system [29, 30]. Moreover, the efficiency-rate needs to reach the landmark $(r = 1)$ [6, 13, 15, 21]. There are some well-known schemes of blockcipher compression function such as MR, Weimar, Hirose, Tandem, Abreast, Nandi, and ISA09 (Table 1). For example, the CR of MR scheme is bounded by $q = 2^{126.70}$ but the r is $1/2$ (q : number of queries). The scheme of Weimar-DM provides tight security bound such as $q = 2^{126.23}$ [6]. Moreover, it follows double key scheduling including $1/2$ efficiency-rate. The scheme of Hirose delivers marginal security bound as $q = 2^{124.55}$ but it ensures a single key scheduling. However, the CR and PR bound of the Tandem-DM and Abreast-DM are

not satisfactory as that of the MR, Weimar, and Hirose [23]. Moreover, the efficiency-rate of Tandem-DM and Abreast-DM is $1/2$ like MR, Weimar, and Hirose [6, 11, 12]. Though the scheme of Nandi is bounded by $q = O\left(2^{2n/3}\right)$ but it provides higher efficiency-rate $(r = 2/3)$ [20]. Additionally, the construction of ISA09 provides better efficiency-rate $(r = 2/3)$ [21]. According to the above discussions and Table 1, most of the existing schemes have rigorous security margin. However, the efficiencies are low for the constructions of MR, Weimar, Hirose, Tandem and Abreast. On the other hand, the schemes of Nandi and ISA09 satisfies higher efficiency-rate. Moreover, the constructions of Nandi and ISA09 satisfies $KS = 3$ and $\#E = 3$ [20, 21]. On the contrary, the OM is serial for Nandi and ISA09 schemes. Thus, the overall efficiencies are not adequate for the ISA09 and Nandi schemes.

Now-a-days, the importance of an efficient blockcipher compression function are enormous [6, 8, 13, 33, 34, 40, 41, 44]. The blockcipher is one of the important cryptographic primitive for the security solution of IoT environment according to certain standards such as ISO/IEC29192-1, ISO/IEC29192-2, ISO/IEC29192-3, and ISO/IEC29192-4,[42–44]. Generally, IoT-end device, RfID, and constrained device are used in IoT environment [39–42]. Furthermore, these devices need to operate fast but the major draw-backs are limited memory, power, and processor [37, 38, 42–44]. Therefore, the cryptographic solution scheme should satisfies the property of better efficiency. In summary, the targets for an efficient blockcipher compression function are as follows:

- higher efficiency-rate
- reasonable key scheduling
- less number of calling blockcipher $(\#E)$
- operational mode
- satisfiable security bound

Contribution. In this paper, a blockcipher based compression function is proposed. The component function of the proposed construction follows one of the secure modes of PGV. The contributions of the proposed construction are as follows:

- efficiency rate, $r = 0.996$
- $KS = 2$
- $\#E = 2$
- Parallel mode
- CR security bound, $q = 2^{125.84}|q$: number of query

In addition, a comparative study of the proposed construction and current familiar schemes is given through Table 2.

Outline. The basic preliminaries are provided in Section 2. The technical details of the proposed scheme are given in Section 3. Section 4 is responsible for the analysis of security bound. Furthermore, the result analysis is given including performance analysis in section 5. Finally, the conclusions and future works are provided in Section 6.

Table 2. Comparison: The proposed scheme and existing familiar schemes [6, 14, 15, 20, 21, 23]

	CR	r	KS	$\#E$	OM
MR	$2^{126.70}$	$r = 0.5$	1	2	P
Weimar	$2^{126.23}$	$r = 0.5$	2	2	P
Hirose	$2^{124.55}$	$r = 0.5$	1	2	P
Tandem	$2^{120.87}$	$r = 0.5$	2	2	P
Abreast	$2^{124.42}$	$r = 0.5$	2	2	P
proposed scheme	$2^{125.84}$	$r = 0.996$	2	2	P
Nandi	$O(2^{2n/3})$	$r = 0.66$	3	3	S
ISA09	$O(2^n)$	$r = 0.66$	3	3	S
MDC-2	$O(2^n)$	$r = 0.5$	2	2	P
MDC-4	$O(2^n)$	$r = 0.5$	4	4	SP

P: Parallel, S: Serial, SP: Semi-Parallel

2 Preliminaries

2.1 ideal cipher model (ICM)

In ideal cipher model, a blockcipher is defined as $\mathcal{B}(n,k)$ where n means block-length and k means key-length. The operation of $\mathcal{B}(n,k)$ is $\mathcal{E} = \{0,1\}^n \times \{0,1\}^k \rightarrow \{0,1\}^n$. The reply of forward $(\mathcal{E})$ and backward $\left(\mathcal{E}^{-1}\right)$ query is random and independent permutation of $\mathcal{K} \in \{0,1\}^k$. Let $\mathcal{BLOCK}_n^k$ is the set of all blockciphers $\mathcal{B}(n,k)$. Under ideal cipher model, $\mathcal{E}$ is chosen randomly from $\mathcal{BLOCK}_n^k$. Actually, $\mathcal{E}$ invokes key and plaintext as input and returns ciphertext as output. On the contrary, input of $\mathcal{E}^{-1}$ are key and ciphertext. Then output is plaintext. Usually, the query and response through $\mathcal{E}$ and $\mathcal{E}^{-1}$ are stored as k_i, x_i, y_i. Moreover, the adversary is not allowed to make any duplicate query [17, 22].

2.2 security definition

There are certain properties, which are responsible for analysing the security issue of blockcipher compression function. For example, collision resistance (CR), preimage resistance (PR), padding oracle attack, and initial value (CV) attack are the most familiar properties [6, 13, 23, 24]. In this section, the collision and preimage resistance of the blockcipher compression function are briefly discussed [16–19] .

collision resistance of compression function The adversary $\mathcal{A}$ is allowed for accessing to the blockcipher oracle $\left(\mathcal{E} \in \mathcal{BLOCK}_n^k\right)$. Hence, the output of compression function are (α_1, β_1, m_1) and (α_2, β_2, m_2). Furthermore, an experiment is defined as Exp-coll$_{f_\mathcal{E}}(\mathcal{A})$. The output of the experiment is 1 iff following condition satisfies.

$$f_\mathcal{E}(\alpha_1, \beta_1, m_1) = f_\mathcal{E}(\alpha_2, \beta_2, m_2) \wedge \{(\alpha_1, \beta_1, m_1) \neq (\alpha_2, \beta_2, m_2)\}$$

, where $f_{\mathcal{E}}$ is a blockcipher compression function and α, β are chaining values including $m|$ message. The advantage of adversary for finding a collision under $f_{\mathcal{E}}$ is defined below. Let, $\mathrm{Adv}_{f_{\mathcal{E}}}^{\mathrm{coll}}(\mathcal{A}) = \Pr[\text{Exp-coll}_{f_{\mathcal{E}}}(\mathcal{A}) = 1]$, where coll stands for collision. The advantage of adversary $\mathcal{A}$ is quantified by the number of queries that are allowed to ask blockcipher oracle. Therefore, $\mathrm{Adv}_{f_{\mathcal{E}}}^{\mathrm{coll}}(q) = \max_{\mathcal{A}}\left\{\mathrm{Adv}_{f_{\mathcal{E}}}^{\mathrm{coll}}(\mathcal{A})\right\}$, where the maximum is taken over all adversaries that ask at most q oracle queries [16, 19].

preimage resistance of compression function The adversary $\mathcal{A}$ has access on blockcipher oracle $\left(\mathcal{E} \in \mathcal{BLOCK}_n^k\right)$. Furthermore, $\mathcal{A}$ selects value of α, β randomly before making any query to blockcipher oracle. Let the feedback of oracle are α' and β' in respect of adversarial query. In addition, assume an experiment $\text{Exp-pre}_{f_{\mathcal{E}}}(\mathcal{A})$, where pre stands for preimage. Hence, the output of the defined experiment is 1 iff:

$$f_{\mathcal{E}}(\alpha_1, \beta_1, m_1) = (\alpha, \beta)$$

, where $f_{\mathcal{E}}$ is a blockcipher compression function and α_1, β_1 are chaining values including $m|$ message. The advantage of adversary for finding a preimage under $f_{\mathcal{E}}$ is defined by $\mathrm{Adv}_{f_{\mathcal{E}}}^{\mathrm{pre}}(\mathcal{A}) = \Pr[\text{Exp-pre}_{f_{\mathcal{E}}}(\mathcal{A}) = 1]$. Moreover, the advantage of $\mathcal{A}$ is evaluated through the total number of queries. Therefore, $\mathrm{Adv}_{f_{\mathcal{E}}}^{\mathrm{pre}}(q) = \max_{\mathcal{A}}\left\{\mathrm{Adv}_{f_{\mathcal{E}}}^{\mathrm{pre}}(\mathcal{A})\right\}$, where the maximum is taken over all adversaries that ask q oracle queries [16, 19].

3 Proposed Scheme

Usually, the efficiency-rate can be increased by using three calls of blockcipher. The above method is used in Nandi and ISA09 [20, 21]. Furthermore, a method of using a pair of chaining values including message in the two blockciphers is also useful. Such kind of method is used in MDC-2 and later in MDC-4 [4, 9, 45]. The proposed construction is actually inspired and followed by the construction of MDC-2 and MDC-4 [4, 9, 45]. However, in respect of security there is a drawback for these (MDC-2, 4) kind of construction. In MDC-2, two chaining values are used as input, where message is common for two blockciphers. There is no dependency between two chaining values as input. On the contrary, it can be said that the computations of the two block ciphers used in the compression function are completely isolated. For example, given the input and output $(x_1, y_1 \rightarrow x_2, y_2)$, if the input is swapped then the new output will be swapped values of the old output $(y_1, x_1 \rightarrow y_2, x_2)$. It actually suffers for symmetric property. Therefore, certain changes are occurred in the proposed construction (Fig. 2). For example, one constant bit 0 and 1 is used to each of the block ciphers as part of the key for the proposed scheme (trivial practice in cryptography, [14]). Hence, the attacker can't predict the output of the chaining values which is given under the assumption where the attacker can freely alter the input of chaining values and message. This premise is used for breaking the symmetric property of the proposed scheme, where $x||y$ and $y||x$ will be treated as two different values. Moreover, the scheme is secured under a generic attack because of the ideal cipher model primitive [26]. Additionally, the MDC-2, MDC-4 are (n, n)-bit $\mathcal{DBL}$ hash functions with efficiency-rate $1/2$ and $1/4$ [24],

where the proposed scheme is based on $(n, 2n)$ blockcipher. Furthermore, a different component function is used in respect of the MDC-2 and MDC-4. The proposed scheme can compress $4n$ bits into $2n$ bits, where MDC-2 and MDC-4 can compress $3n$ bits to $2n$ bits. Furthermore, the proposed scheme satisfies type-1 (from Stam's conjecture), where two blockciphers $\mathcal{E}_{\mathrm{l}}$, $\mathcal{E}_{\mathrm{r}}$ are distinct and independent under the ICM [8, 16]. In general, the proposed scheme is defined as variant of the MDC-2 and MDC-4.

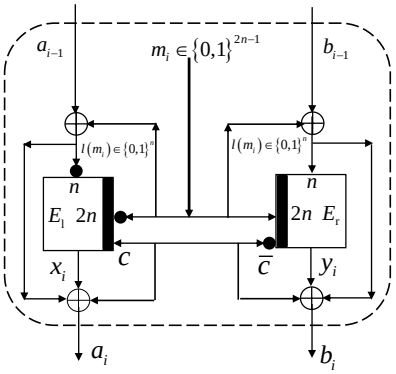

Fig. 2. Proposed Scheme (Variant of MDC-2, 4)

Definition 1. Let $\mathcal{E} \in \mathcal{BLOCK}_n^k$ be a block cipher taking a set of k-bit key and n-bit block-length such that $\mathcal{E}_{\mathrm{l,r}} = \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$. $\mathcal{E}^{\mathrm{dbl}} = \{0,1\}^k \times \{0,1\}^{2n} \rightarrow \{0,1\}^{2n}$ is defined as a double block length (dbl) cipher and parallel calling of two independent blockciphers of $\mathcal{E}_{\mathrm{l,\,r}}$ such that,

$$x_i \leftarrow \mathcal{E}_{\mathrm{l},(\overline{m}_i||c)}\left(\overline{a_{i-1} \oplus l\left(m_i\right)}\right)$$
$$y_i \leftarrow \mathcal{E}_{\mathrm{r},(m_i||\bar{c})}\left(b_{i-1} \oplus l\left(m_i\right)\right)$$

where parameters are defined as $m_i \in \{0,1\}^{2n-1}, (a, b, x, y) \in \{0,1\}^n$ and $l\left(m_i\right) =$ lsb of $m_i \in \{0,1\}^n, c = \{1\}$. Thus, the final output is $f_{\mathcal{E}}\left(a_i, b_i\right)$ where,

$$\begin{cases} a_i \leftarrow x_i \oplus \left(a_{i-1} \oplus l\left(m_i\right)\right) \oplus c \\ b_i \leftarrow y_i \oplus \left(b_{i-1} \oplus l\left(m_i\right)\right) \oplus \bar{c} \end{cases}$$

Definition 2. Let $f_{\mathcal{E}} = \{0,1\}^k \times \{0,1\}^{2n} \rightarrow \{0,1\}^{2n}$ be a blockcipher based compression function such as $\left(a_i, b_i, m_i\right) = f\left(a_i, b_i, m_i\right)$, where, $a_i \in \{0,1\}^n$, $b_i \in \{0,1\}^n$, $m_i \in \{0,1\}^{2n-1}$, and $c = \{0,1\}$. Therefore, $f_{\mathcal{E}}$ consists of ideal blockcipher $(\mathcal{E})$ such as:

$$\left[\begin{array}{l} a_i = f_{\mathrm{l}}\left(\overline{a_{i-1} \oplus l\left(m_i\right)}, \bar{m}_i||c\right) \oplus \left(a_{i-1} \oplus l\left(m_i\right)\right) \oplus c \leftarrow \\ \mathcal{E}_{\mathrm{l}}\left(\overline{a_{i-1} \oplus l\left(m_i\right)}, \bar{m}_i||c\right) \oplus \left(a_{i-1} \oplus l\left(m_i\right)\right) \oplus c \end{array}\right]$$

$$\begin{bmatrix} b_i = f_{\mathrm{r}}\left(b_{i-1} \oplus l\left(m_i\right), m_i || \bar{c}\right) \oplus \left(b_{i-1} \oplus l\left(m_i\right)\right) \oplus \bar{c} \leftarrow \\ \mathcal{E}_{\mathrm{r}}\left(b_{i-1} \oplus l\left(m_i\right), m_i || \bar{c}\right) \oplus \left(b_{i-1} \oplus l\left(m_i\right)\right) \oplus \bar{c} \end{bmatrix}$$

4 Security Analysis

The security proof of the proposed scheme follows an ICM [16, 17], where $\mathcal{A}$ is not allowed to make any duplicate query. For example, the query of $\mathcal{E}\left(k, x\right) = y$ isn't being executed by the adversary, if $\mathcal{E}^{-1}\left(k, y\right) = x$ query is already in the query storage $(\mathcal{Q})$. The adversary $\mathcal{A}$ searches for a collision under a pair of different inputs (query) through the blockcipher oracle. Additionally, $\mathcal{A}$ tries to find an output of compression function for making collision with initial chaining value. Moreover, the preimage attack means: Adversary $\mathcal{A}$ selects α', β' randomly and tries to find $f\left(\alpha, \beta, m\right) = \alpha', \beta'$. In addition, the advantage of $\mathcal{A}$ is very limited to get the above success.

4.1 collision security analysis

An adversary $\mathcal{A}$ has access to a blockcipher oracle for finding a collision. The query is Q_i and corresponding response is triplet as (m : mesage, k : key, c : ciphertext). For any i-th iteration ($i \leq q$), the query process looks either $Q_i \in \{(m, k) = c\}$ or $Q_i \in \{(c, k) = m\}$. The Q_i stores in $\mathcal{Q} \in (Q_1, Q_2, ..., Q_i)$ for each iteration of i where $\mathcal{Q}$: query storage. Under this circumstance, adversary $\mathcal{A}$ has target to find,

$$f_{\mathcal{E}}\left(m_i, k_i, c_i\right) = f_{\mathcal{E}}\left(m_j, k_j, c_j\right) | \because \left(m_i, k_i, c_i\right) \neq \left(m_j, k_j, c_j\right) \wedge \left(i \neq j\right) \quad (1)$$

According to the definition of proposed scheme, 1 is re-defined as:

$$f_{\mathcal{E}}\left(a_i, b_i, m_i\right) = f_{\mathcal{E}}\left(a_j, b_j, m_j\right) | \because \left(a_i, b_i, m_i\right) \neq \left(a_j, b_j, m_j\right) \wedge \left(i \neq j\right) \quad (2)$$

Theorem 1. *Let $f_{\mathcal{E}}$ be a double block-length compression function (Def. 1, 2). An adversary, $\mathcal{A}$ is assigned for finding a collision (coll) under the $f_{\mathcal{E}}$ after q pairs of queries. Hence, the advantage of $\mathcal{A}$ is bounded by,*

$$Adv_{f_{\mathcal{E}}}^{coll}\left(q\right) \leq \frac{6q^2 - 2q}{\left(2^n - q\right)^2}$$

Proof. An adversary $\mathcal{A}$ makes a relevant query to the blockcipher oracle, where the number of query is limited by q queries. For any i-th query, the reply of x_i and y_i randomly selects by the adversary from the blockcipher oracle. The main difficulty is to find out the set size of an oracle from where these fresh value come. There are three possible incidents that are responsible for collision-hit under any i-th iteration. In the beginning, the three incidents are clarified through two targets ($\mathcal{TAR}1$, $\mathcal{TAR}2$). The goal of the first incident is to find a collision for two distinct queries ($j < i$) where $\mathcal{TAR}1$ represents the responsibilities of the first incident. The $\mathcal{TAR}2$ is responsible for second and third incident. Since $\mathcal{A}$ has target to find a collision through single query. Furthermore, $\mathcal{A}$ investigates for a collision against initial chaining values. Finally, three phases of $\mathcal{QUERY}$, $\mathcal{RESPONSE}$, and $\mathcal{CHECK}$ have been defined under

Algorithm 1 $\mathcal{TAR}1$ (for notations follow Def. 1, 2)

$\mathcal{Q}$: Query storage, q: query, $\mathcal{A}$: Adversary, ${}^{\mathcal{TAR}1}\mathcal{C}_i$: event, m : message
for each node $(i < q)$ **do**
  $\mathcal{QUERY}$: $\left(\mathcal{E} \leftarrow \mathcal{BLOCK}_n^k\right) \leftarrow \mathcal{A}^{\mathcal{E},\mathcal{E}^{-1}}$
  $\mathcal{RESPONSE}$:
  $q_{i,1} \leftarrow \left(x_i = \mathcal{E}_{(m_i||c)}\left(\overline{a_{i-1} \oplus l\left(m_i\right)}\right)\right) \wedge q_{i,2} \leftarrow y_i$
  $\mathcal{CHECK}$:
  if $(q_{i,1}, q_{i,2})$=$(q_{j,1}, q_{j,2})$, where $j < i$ **then**
    1. Call: ${}^{\mathcal{TAR}1}\mathcal{C}_i \wedge$ terminate
  else
    write the value of $q_{i,1}, q_{i,2}$ to $\mathcal{Q}$
  end if
end for

$\mathcal{TAR}1$ and $\mathcal{TAR}2$. Let adversary $\mathcal{A}$ is allowed to ask query to blockcipher oracle at $\mathcal{QUERY}$ phase. Moreover, corresponding feedback assign under $\mathcal{RESPONSE}$ phase. In addition, a collision is checked in the phase of $\mathcal{CHECK}$.

collision probability based on the first incident $(\mathcal{TAR}1)$. Under an iteration of i, a pair of query is executed that returns two distinct outputs. According to algorithm 1, there is a chance to make collision through two different query-pairs after any i-th $(j < i < q)$ iteration. For example, a query pair of j-th iteration are:

$$\begin{bmatrix} a_j \leftarrow \mathcal{E}_{\mathrm{l},\bar{m}_j||c}\left(\overline{a_{j-1} \oplus l\left(m_j\right)}\right) \oplus \left(a_{j-1} \oplus l\left(m_j\right)\right) \oplus c, \\ b_j \leftarrow \mathcal{E}_{\mathrm{r},m_j||\bar{c}}\left(a_{j-1} \oplus l\left(m_j\right)\right) \oplus \left(a_{j-1} \oplus l\left(m_j\right)\right) \oplus \bar{c} \end{bmatrix}$$

Moreover, the query responses are $a_i \leftarrow E_{\mathrm{l},\bar{m}||c}\left(\overline{a_{i-1} \oplus l\left(m_i\right)}\right) \oplus \left(a_{i-1} \oplus l\left(m_i\right)\right) \oplus c$ and $b_i \leftarrow E_{\mathrm{r},m||\bar{c}}\left(a_{i-1} \oplus l\left(m_i\right)\right) \oplus \left(a_{i-1} \oplus l\left(m_i\right)\right) \oplus \bar{c}$ on the i-th $(j < i)$ iteration. Let ${}^{\mathcal{TAR}1}\mathcal{C}_i$ be an event, where adversary tries to find a collision through different two iterations $(j < i \leq q)$. Thus, equation 2 is re-defined as:

$$\begin{cases} \{a_i \leftarrow (c \oplus a_{i-1} \oplus l\left(m_i\right) \oplus x_i)\} = \\ \{a_j \leftarrow (c \oplus a_{j-1} \oplus l\left(m_j\right) \oplus x_j)\} \end{cases} \vee \begin{cases} \{a_i \leftarrow (c \oplus a_{i-1} \oplus l\left(m_i\right) \oplus x_i)\} = \\ \{b_j \leftarrow (\bar{c} \oplus b_{j-1} \oplus l\left(m_j\right) \oplus y_j)\} \end{cases} \tag{3}$$

$$\begin{cases} \{b_i \leftarrow (\bar{c} \oplus b_{i-1} \oplus l\left(m_i\right) \oplus y_i)\} = \\ \{a_j \leftarrow (c \oplus a_{j-1} \oplus l\left(m_j\right) \oplus x_j)\} \end{cases} \vee \begin{cases} \{b_i \leftarrow (\bar{c} \oplus b_{i-1} \oplus l\left(m_i\right) \oplus y_i)\} = \\ \{b_j \leftarrow (\bar{c} \oplus b_{j-1} \oplus l\left(m_j\right) \oplus y_j)\} \end{cases} \tag{4}$$

From $3 \wedge 4$, the probability of collision hit under the event of ${}^{\mathcal{TAR}1}\mathcal{C}_i$ is $\frac{2(i-1)}{(2^n-(i-1))^2}$ (when $j < i \leq q$). Therefore, the probability of single event under the $\mathcal{TAR}1$ is:

$$\Pr\left[{}^{\mathcal{TAR}1}\mathcal{C}\right] = 2\left(i-1\right) \Big/ \left(2^n - (i-1)\right)^2$$

If ${}^{\mathcal{TAR}1}\mathcal{C}$ be the events of all colliding pairs under the $f_{\mathcal{E}}$ for q pairs of queries. Hence,

$$\Pr\left[{}^{\mathcal{TAR}1}\mathcal{C}\right] = \Pr\left[{}^{\mathcal{TAR}1}\mathcal{C}_2 \vee .. \vee {}^{\mathcal{TAR}1}\mathcal{C}_q\right]$$

$$\leq \sum_{i=2}^{q} \Pr\left[{}^{\mathcal{TAR}1}\mathcal{C}_i\right] \leq \frac{2 \times 2 \times (i-1)}{(2^n - i)^2} = \frac{2q^2 - 2q}{(2^n - q)^2} \tag{5}$$

collision probability based on the second and third incident ($\mathcal{TAR}2$). Let a_i, b_i be the output of compression function ($i < q$), where

$$\{(a_i \leftarrow x_i \oplus (a_{i-1} \oplus l\,(m_i)) \oplus c)\,, (b_i \leftarrow y_i \oplus (b_{i-1} \oplus l\,(m_i)) \oplus \bar{c})\}$$

Hence, there is a probability to make collision when $a_i = b_i$. Let ${}^{\mathcal{TAR}2}\mathcal{C}_i$ be a collision event for the above condition under the check phase of $i < q$. Furthermore, there is an option to make a collision with initial chaining values. For example, the output pair of the proposed scheme a_i, b_i collides with the initial chaining values (a_0, b_0) at any phase of query process. Therefore, the conditions of collision-hit under the initial key attack are $\{a_i = (a_0)\,, (b_0)\} \vee \{b_i = (a_0)\,, (b_0)\}$.

Algorithm 2 (for $\mathcal{TAR}2$)

1: $\mathcal{TAR}2$: line 2 to 7 is replaced of $\mathcal{TAR}1$ (from line 4)
2: **if** $(q_{i,1} = q_{i,2})$ **then**
3: create the event $\left({}^{\mathcal{TAR}2}\mathcal{C}_i\right) \wedge$ terminate
4: "AND"
5: **if** $(q_{i,1}, q_{i,2}) = (q_{0,1}, q_{0,2})$ **then**
6: create the event $\left({}^{\mathcal{TAR}2}\mathcal{C}_i\right) \wedge$ terminate
7: **end if**
8: **end if**

Hence, the probability of collision under two incidents is at most $1/(2^n - i) \times 2 \times 2/(2^n - i)$. Finally, the probability of these two incidents under the event of ${}^{\mathcal{TAR}2}\mathcal{C}$ for q pairs of queries is:

$$\Pr\left[{}^{\mathcal{TAR}2}\mathcal{C}\right] = \Pr\left[{}^{\mathcal{TAR}2}\mathcal{C}_1 \vee ..{}^{\mathcal{TAR}2}\mathcal{C}_q\right]$$

$$\leq \sum_{i=1}^{q} \Pr\left[{}^{\mathcal{TAR}2}\mathcal{C}_i\right] = \sum_{i=1}^{q} \frac{1}{(2^n - i)} \times \frac{2 \times 2}{(2^n - i)} \leq \frac{q}{(2^n - q)} \times \frac{2 \times 2 \times q}{(2^n - q)} = \frac{4q^2}{(2^n - q)^2} \tag{6}$$

Adding the values of 5 and 6, Theorem 1 satisfies.

4.2 Preimage Security Analysis

A standard proof technique of Armknecht et al. is used for the preimgae security proof of the proposed scheme[14]. The PR security bound of MR, Weimar, Hirose, Tandem and Abreast is also based on [14]. The two important concepts are adopted such as query: super, normal and adjacent query-pair from [6, 14]. Let $\mathcal{A}$ randomly picks the output value of compression function (a', b'). Now $\mathcal{A}$ has target to find a probability for preimage-hit through $f_{\mathcal{E}}^{\mathrm{p}}\,(a_i, b_i, m) = (a', b')$ condition, where a_i, b_i, m : input of compression function and $a_i \neq b_i$.

Theorem 2. *Let $f_{\mathcal{E}}$ be a double block-length compression function. An adversary $\mathcal{A}$ is defined for finding a preimage-hit under the $f_{\mathcal{E}}$ after q pairs of queries. Hence, the advantage of $\mathcal{A}$ is bounded by,*

$$Adv^{pre}_{f_{\mathcal{E}}}(q) \leq 8q \Big/ N^2 + 8q \Big/ (N-q)^2$$

Proof. An adversary $\mathcal{A}$ keeps a query database in the form of,

$$\begin{bmatrix} \left\{ a_i \leftarrow \mathcal{E}_{\mathrm{l},\overline{m}||c}\left(\overline{a_{i-1} \oplus l(m_i)}\right) \oplus (a_{i-1} \oplus l(m_i)) \oplus c \right\} \\ \text{and } \left\{ b_i \leftarrow \mathcal{E}_{\mathrm{r},m||\bar{c}}(b_{i-1} \oplus l(m_i)) \oplus (b_{i-1} \oplus l(m_i)) \oplus \bar{c} \right\} \end{bmatrix}$$

In such a fashion, when the oracle size reaches $N/2$ (N : Oracle size (2^n)), the rest of the queries under the key-set reaches the adversary as free query [6, 14, 25]. This free set of queries exist in the domain which is called the super query database ($\mathcal{SQD}$). On the other hand, the first $N/2$ is defined as a normal query database ($\mathcal{NQD}$) [14]. Additionally, the free queries are asked by the adversary non-adaptively in the super query database ($\mathcal{SQD}$). Therefore the successful conditions of a preimage-hit are:

$$\begin{aligned} &\left\{ a_i \leftarrow \mathcal{E}_{\mathrm{l},\bar{m}_i||c}\left(\overline{a_{i-1} \oplus l(m_i)}\right) \oplus (a_{i-1} \oplus l(m_i)) \oplus c \right\}, \\ &\left\{ a_j \leftarrow \mathcal{E}_{\mathrm{l},\bar{m}_j||c}\left(\overline{a_{j-1} \oplus l(m_j)}\right) \oplus (a_{j-1} \oplus l(m_j)) \oplus c \right\} = \{(a'), (b')\} \end{aligned} \tag{7}$$

$$\begin{aligned} &\left\{ b_i \leftarrow \mathcal{E}_{\mathrm{r},m||\bar{c}_i}(b_{i-1} \oplus l(m_i)) \oplus (b_{i-1} \oplus l(m_i)) \oplus \bar{c} \right\}, \\ &\left\{ b_j \leftarrow \mathcal{E}_{\mathrm{r},m||\bar{c}_j}(b_{j-1} \oplus l(m_j)) \oplus (b_{j-1} \oplus l(m_j)) \oplus \bar{c} \right\} = \{(a'), (b')\} \end{aligned} \tag{8}$$

Equation 7 and 8 can occur in either in the domain of a normal query win ($\mathcal{NQW}$) or super query win ($\mathcal{SQW}$). Therefore, the probability of the preimage-hit is $\Pr[\mathcal{NQW}] + \Pr[\mathcal{SQW}]$.

Algorithm 3

```
1: procedure PREIMAGE TARGET
2:     for i < N/2 do (for normal query)
3:         run QUERY ∧ RESPONSE ∧ CHECK
4:     end for
5:     for N/2 < i < N do for super query
6:         (QUERY ∧ CHECK)
7:     end for
8: end procedure
```

probability of $\mathcal{NQW}$. The adversary $\mathcal{A}$ makes any relevant query independently and receives a_i, b_i. Furthermore, $\mathcal{A}$ executes until the oracle set size reaches to $N/2$ [6, 14]. According to the above mentioned conditions (7, 8), the hitting probability is $2 \times 2 \Big/ (2^n - q)$.

If $\mathcal{A}$ makes a query $\mathcal{E}_{1,\overline{m}_i||c}\left(\overline{a_{i-1} \oplus l\left(m_i\right)}\right)$ (left block) then the answer of a right block provides as free query to $\mathcal{A}$ because of the adjacent query pair [6, 14]. Thereafter, the set size is $(2^n - q)/2$ which outfits the probability as $2\Big/(2^n - q)$. Thus, the probability of the normal query is:

$$\Pr\left[\mathcal{NQW}\right] = q \times 2 \times 2/(2^n - q) \times 2/(2^n - q) = {}^{8q}\Big/_{(2^n - q)^2} \tag{9}$$

probability of $\mathcal{SQW}$. The concept of a super query oracle is very simple [6, 14]. If the query oracle reaches at the point of $N/2$, then the rest of the queries set as free to the adversary [6, 14]. Later these queries are asked by the adversary non-adaptively [14] for finding a preimage-hit (Algorithm 3). Moreover, the preimage-hit is notified either in this domain ($\mathcal{SQD}$) or not. Thus, the probability is either $2/N$ or 0 for any output value of a_i/b_i. Now a pair of conditions under $\mathcal{SQW}$ are:

$$\left\{a_i \leftarrow \left(l\left(m_i\right) \oplus a_{i-1} \oplus x_i\right) \oplus c\right\} = \left(a'\right), \left(b'\right) \tag{10}$$

$$\left\{b_i \leftarrow \left(l\left(m_i\right) \oplus b_{i-1} \oplus y_i\right) \oplus \overline{c}\right\} = \left(a'\right), \left(b'\right) \tag{11}$$

According to 10, the answer of a_i has a possibility to come from the set size of $N/2$. Hence, the probability is $2/N$. Recalling the concept of an adjacent query pair (free query) [6, 14], where the answer of another block (right block) comes from the set size of $N/2$. As a result, the probability of 10 is in total $4/N^2$. In similar way, the probability of 11 is $4/N^2$. Now, the final probability of the $\mathcal{SQW}$ is evaluated based on the the number of points for a $\mathcal{SQW}$, the cost of $\mathcal{SQW}$ and the probability of obtaining preimgae-hit such as:

$$\Pr\left[\mathcal{SQW}\right] = q/(N/2) \times (N/2) \times 2 \times \left(4/N^2\right) = 8q/N^2 \tag{12}$$

Adding the values of 9 and 12, Theorem 2 satisfies.

5 Result Analysis

5.1 collision resistance analysis

Theorem 1 provides a probability of collision hit under the given adversary $\mathcal{A}$. The number of queries (q) is important for finding an upper bound of the collision security. Hence, the value of q is required to investigate when the adversarial advantage is $1/2$ (birthday attack).

Let, $N = 2^n$ and $\mathrm{Adv}^{\mathrm{coll}}_{f_E}(\mathcal{A}) \leq \frac{6q^2 - 2q}{(2^n - q)^2}$ [Theorem 1.], where $n = 128$. According to the birthday attack [1, 6, 13, 20, 21], $\mathrm{Adv}^{\mathrm{coll}}_{f_\mathcal{E}}(\mathcal{A}) = \frac{1}{2}$. Thus, the number of queries are $q = 2^{125.84}$.

5.2 Efficiency-rate

The efficiency-rate of a blockcipher based compression function is defined as $r = \frac{|m|}{(n \times \#E)}$, where $|m|$=length of message, n=blocklength and $\#E$=number of blockcipher calls. According to the definitions (Def. 1, Def.2) of the proposed scheme, the efficiency-rate is $r = 0.996 \Rightarrow r \approx 1$. In Fig. 3, the proposed scheme is compared with the existing schemes in respect of efficiency-rate .

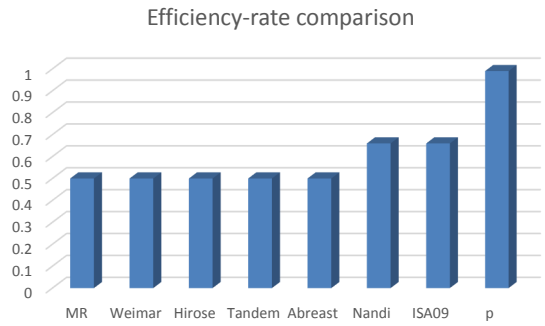

Fig. 3. Comparison of efficiency-rate

Table 3. Required memory for key scheduling [20, 21, 27]

Name	KS	required memory (in byte, B)
Proposed scheme	2	2×176 B
Nandi [20]	3	3×176 B
ISA09 [21]	3	3×176 B

Table 4. Required memory for key scheduling [6, 23, 27]

Name	KS	l	B	$\mathcal{V}$	B $+ \mathcal{V}$
Proposed scheme	2	$l = 1$	$\mathfrak{a} \leftarrow 2 \times 176$ B	γ	$\mathfrak{a}$
MR [23]	1	$l = 2$	$\mathfrak{b} \leftarrow 1 \times 176$ B	γ	$\mathfrak{b} + \gamma$
Weimar [6]	2	$l = 2$	$\mathfrak{c} \leftarrow 2 \times 176$ B	γ	$\mathfrak{c} + \gamma$
Hirose [13]	1	$l = 2$	$\mathfrak{d} \leftarrow 1 \times 176$ B	γ	$\mathfrak{d} + \gamma$
Tandem [12]	2	$l = 2$	$\mathfrak{e} \leftarrow 2 \times 176$ B	γ	$\mathfrak{e} + \gamma$
Abreast [11]	2	$l = 2$	$\mathfrak{f} \leftarrow 2 \times 176$ B	γ	$\mathfrak{f} + \gamma$

l: number of iteration for processing $2n$-bit message
B: required memory for key scheduling in byte
$\mathcal{V}$: memory require for storing output ($\gamma = 2n$ bit)
B $+ \mathcal{V}$: total required memory for key scheduling, when message $= 2n$

5.3 Performance analysis

In this section, a comparison study is given for the proposed scheme in respect of memory resources. It is known that 176 bytes of memory is required for single key scheduling [27]. For example, a $2n$-bit size of message is taken for encryption. Therefore, the following Table 3 and 4 are made based on the characteristics of the current familiar schemes and the proposed scheme. For any $\mathcal{DBL}$ compression function, the output is $2n$-bit. Therefore, assume that the minimum $2n \rightarrow \gamma$ bit is required to store the output value (denoted as $\mathcal{V}$) of i-th iteration. In Table 4, the message size is $2n$-bit for example. Hence, the memory resource doesn't need to store the output for the proposed scheme.

Next, the above cost (Table 4) is generalized including the number of iterations (l) for tn-bit message $(t > 2)$ in Table 5. Additionally, the proposed scheme is faster than that of the MR, Weimar, Tandem, Abreast (if, $m > 2n$) in certain cases.

Table 5. Required memory for key scheduling, when $m = tn$

Name	l	$\mathcal{V}$	B $+ \mathcal{V}$
Proposed scheme	$l = tn/2n$	γ	$\mathfrak{a} + \gamma$
MR	$l = tn/n$	γ	$\mathfrak{b} + \gamma$
Weimar	$l = tn/n$	γ	$\mathfrak{c} + \gamma$
Hirose	$l = tn/n$	γ	$\mathfrak{d} + \gamma$
Tandem	$l = tn/n$	γ	$\mathfrak{e} + \gamma$
Abreast	$l = tn/n$	γ	$\mathfrak{f} + \gamma$

$\mathfrak{a}, \mathfrak{b}, \mathfrak{c}, \mathfrak{d}, \mathfrak{e}, \mathfrak{f}$: these values come from the Table IV (column B)

6 Conclusion

This paper studied the gap between security bound and efficiency of compression function for the cryptographic hash. Additionally, study result introduces that the blockcipher based compression function is more suitable than the scratch based construction for security solution of IoT-end devices, RfID, and constrained devices. Thus, a better efficient compression function (blockcipher based) is proposed in this paper. Additionally, the proposed scheme provides improved efficiency-rate, less call of blockcipher, and reasonable security bound. It satisfies two calls of $2n$-bit key property, where two block ciphers are independent. The proof technique of this scheme depends on the ICM tool. The proposed scheme has a provision of fixed size message encryption property. Therefore, this property opens a window for new applications, where a variable length of the message can be encrypted without padding. Finally, the proposed scheme is secure under one of the modes of PGV which can be extended to make the scheme secure under all modes of the PGV [17–19].

References

1. A. Bogdanov, G. Leander, C. Paar, A. Poschmann, M. J. B. Robshaw, Y. Seurin , *"Hash Functions and RFID Tags: Mind the Gap," LNCS, CHES,* vol. 5154, pp. 283-299, 2008.
2. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography,* 5th ed, CRC Press, 2001.
3. J. P. Kaps, B. Sunar, "Energy Comparison of AES and SHA-1 for Ubiquitous Computing," *LNCS, Emerging in Embedded and Ubiquitous Computing*, vol. 4097, pp. 372-381, 2006.
4. X. Wang, X. Lai, D. Feng, H. Chen, X. Yu, *"Cryptanalysis of the Hash Functions MD4 and RIPEMD," LNCS, EUROCRYPT*, vol. 3494, pp. 1-18, 2005.
5. X. Wang, X. Lai, X. Yu, *"Finding Collisions in the Full SHA-1.," CRYPTO*, vol. 3621, 2005.

6. E. Fleischmann, C. Forler, S. Lucks, J. Wenzel, *"Weimar-DM: A Highly Secure Double-Length Compression Function," LNCS, ACISP*, vol. 7372, pp. 152-165, 2012.
7. J. Lee, K. Kapitanova, S. H. Son *"The price of security in wireless sensor networks," ELSEVIER, Computer Network*, vol. 54, no. 17, pp. 2967-2978, December 2010.
8. O. Ozen, M. Stam, *"Another Glance at Double-Length Hashing," LNCS, Cryptography and Coding*, vol. 5291, pp. 176-201, 2009.
9. J. Lee, M. Stam, *"MJH: A Faster Alternative to MDC-2," CT-RSA*, vol. 6558, 213-236, 2011.
10. X. Lai, X. Massey, L. J., "Hash function based on block ciphers," *LNCS, EUROCRYPT*, vol. 658, pp. 55-70, 1993.
11. J. Lee, D. Kwon, *"The Security of Abreast-DM in the Ideal Cipher Model," IEICE Transactions*, vol. 94-A(1), pp. 104-109, 2011.
12. J. Lee, M. Stam, J. Steinberger, "The Collision Security of Tandem-DM in the Ideal Cipher Model," *LNCS, CRYPTO*, vol. 6841, pp. 561-577, 2011.
13. S. Hirose, *"Some Plausible Constructions of Double-Block-Length Hash Functions," LNCS, FSE*, vol. 4047, pp. 210-225, 2006.
14. F. Armknecht, E. Fleischmann, M. Krause, J. Lee, M. Stam, J. Steinberger, *"The Preimage Security of Double-Block-Length Compression Functions," LNCS, ASIACRYPT*, vol. 7073, pp. 233-251, 2011.
15. B. Mennink, *"Optimal Collision Security in Double Block Length Hashing with Single Length Key," LNCS, ASIACRYPT*, vol. 7658, pp. 526-543, 2012.
16. J. A. Black, P. Rogaway, T. Shrimpton, *"Black-Box Analysis of the Block-Cipher-Based Hash-Function Constructions from PGV," LNCS, CRYPTO*, vol. 2442, pp. 320-335, 2002.
17. J. A. Black, P. Rogaway, T. Shrimpton, M. Stam, *"An Analysis of the Blockcipher-Based Hash Functions from PGV," LNCS, J.CRYPTOL*, vol. 23, pp. 519-545, 2010.
18. S. Hirose, H. Kuwakado., *"Collision Resistance of Hash Functions in a Weak Ideal Cipher Model," IEICE Transactions*, vol. 95 A(1), pp. 251-255, 2012.
19. M. Liscov, *"Constructing an ideal hash function from weak ideal compression function," LNCS, SAC*, vol. 4356, pp. 358-375, 2006.
20. M. Nandi, W. Lee, K. Sakurai, S. Lee, *"Security Analysis of a 2/3-Rate Double Length Compression Function in the Black-Box Model," LNCS, FSE*, vol. 3557, pp. 243-254, 2005.
21. J. Lee, S. Hong, J. Sung, H. Park, *"A New Double-Block-Length Hash Function Using Feistel Structure," LNCS, ISA*, vol. 5576, pp. 11-20, 2009.
22. C. E. Shannon, *"Communication Theory of Secrecy Systems," Bell Systems Technical Journal*, vol. 128-4, pp. 656-715, 1949.
23. A. Miyaji, R. Mazumder, *"A new (n, 2n) Double Block Length Hash Function based on Single Key Scheduling," IEEE explore, AINA*, PP. 564-570, 2015.
24. F. Abed, C. Forler, E. List, S. Lucks, J. Weznel *"Counter-b DM: A Provably Secure Family of Multi-Block-Length Compression Functions," LNCS*, vol. 8469, pp. 440-458, 2014.
25. J. S. Coron, Y. Dodis, E. List, S. Lucks, J. Weznel, *"Merkle-Damgard revisited: How to construct a hash function," LNCS, Crypto*, vol. 3621, pp. 430-448, 2005.
26. D. Yevgeniy, P. Prashant, *"On the Relation Between the Ideal Cipher and the Random Oracle Models," LNCS, Theory of Cryptography*, vol. 3876, pp. 184-206, 2006.
27. D. Joan, R. Vincent, *"The Design of Rijndael, AES-The Advanced Encryption Standard"*, ISBN 978-3-662-04722-4, Springer Press, 2002.
28. H. Kuwakado, S. Hirose, *"Hashing Mode Using a Lightweight Blockcipher*, *LNCS, Cryptography and Coding"*, vol. 8308, pp. 213-231, 2013.
29. D. Burak *"Parallelization of a Block Cipher Based on Chaotic Neural Networks"*, *LNAI, ICAISC*, pp. 192-201, 2015.
30. J. W. Bos, O. Ozen, M. Stam, *"Efficient Hashing Using the AES Instruction Set"*, *LNCS, CHES*, vol. 6917, pp. 507-522, 2011.

31. R. Mazumder, A. Miyaji, *"A New Scheme of Blockcipher Hash"*, *IEICE Transactions*, Vol. 99-D (4), 2016.
32. L. R. Knudsen, F. Mendel, C. Rechberger, S. S. Thomsen, *"Cryptanalysis of MDC-2"*, *LNCS, Eurocrypt*, Vol. 5479, pp. 106-120, 2009.
33. A. Miyaji, R. Mazumder, T. Sawada*"A New (n, n) Blockcipher Hash Function: Apposite for Short Messages"*, *IEEE Explore, AsiaJCIS*, pp. 56-63, 2014.
34. R. Mazumder, A. Miyaji, *"A Single Key Scheduling based Compression Function"*, *LNCS, CRiSIS*, pp. 207-222, vol. 9572, 2015.
35. L. Barreto, A. Celesti, M. Villari, M. Fazio, A. Puliafito, *"An Authentication Model for IoT Clouds"*, *IEEE explore, ASONAM*, pp. 1032-1035, 2015.
36. A. Riahi, E. Natalizio, Y. Challal, N. Mitton, A. Iera, *"A systemic and cognitive approach for IoT security"*, *IEEE explore, ICNC*, pp. 183-188, 2014.
37. J. Y. Lee, Y. H. Huang, *"A lightweight authentication protocol for Internet of Things"*, *IEEE explore, ISNE*, pp. 1-2, 2014.
38. Q. Jing, A. V. Vasilakos, J. Wan, *"Security of the Internet of Things: perspectives and challenges*, *Springer, Wireless Networks*, volume-20, issue 8, pp. 2481-2501, 2014.
39. M. Abomhara, G. M. Kien, *"Security and privacy in the Internet of Things: Current status and open issues*, *IEEE explore, PRIMS*, pp. 1-8, 2014.
40. A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, *"Internet of Things for Smart Cities*, *IEEE Internet of Things Journal*, volume-1, issue 1, pp. 22 - 32, 2014.
41. L. D. Xu, W. He, S. Li, *"Internet of Things in Industries: A Survey*, *IEEE Transactions on Industrial Informatics*, volume-10, issue 4, pp. 2233 - 2243, 2014.
42. S. Hirose, K. Ideguchi, H. Kuwakado, T. Owada, B. Preneel, H. Yoshida, *"A Lightweight 256-Bit Hash Function for Hardware and Low-End Devices: Lesamnta-LW"*, *LNCS, ICISC*, vol. 6829, pp. 151-168, 2010.
43. T. Shirai, K. Shibutani, T. Akishita, S. Moriai, T. Iwata *"The 128-bit Blockcipher CLEFIA"*, *IACR archive, Extended Abstract*, https://www.iacr.org/archive/fse2007/45930182/45930182.pdf
44. H. Yoshida *"On the standardization of cryptographic application techniques for IoT devices in ITU techniques for IoT devices in ITU-T and ISO/IEC JTC 1 T and ISO/IEC JTC1*, https://www.ietf.org/proceedings/94/slides/slides-94-saag-2.pdf, 2015
45. E. Fleischmann, C. Forler, and S. Lucks *"The Collision Security of MDC-4*, *LNCS, Africacrypt*, vol. 7374, pp. 252-269, 2012.