PAPER Special Section on Cryptography and Information Security

PGV-Style Block-Cipher-Based Hash Families and Black-Box Analysis

Wonil LEE[†], **Mridul NANDI**[††], **Palash SARKAR**[††], **Donghoon CHANG**[†††], **Sangjin LEE**[†††a)], ***Nonmembers***, ***and*** **Kouichi SAKURAI**[†], ***Member***

SUMMARY In [1] it was proved that 20 of 64 PGV hash functions [2] based on block cipher are collision-resistant and one-way in the black-box model of the underlying block cipher. Here, we generalize the definition of PGV-hash function into a hash family and we will prove that, aside from the previously reported 20 hash functions, we have 22 more collision-resistant and one-way hash families. As all these 42 families are keyed hash family, these are also target-collision-resistant. All these 42 hash families have tight upper and lower bounds on (target) collision-resistant and one-wayness.

key words: *hash function, block cipher, black-box model, provable security*

1. Introduction

Brief History. Preneel, Govaerts, and Vandewalle [2] considered the 64 basic ways of constructing a (collision-resistant) hash function $H : (\{0,1\}^n)^* \to \{0,1\}^n$ from a block cipher $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$. They regarded 12 of these 64 schemes as secure, though no proofs or formal claims were given. After that Black, Rogaway, and Shrimpton [1] presented a more proof-centric look at the schemes from PGV, providing both upper and lower bounds for each. They proved that, in the black box model of a block cipher, 12 of 64 compression functions are CRHFs (Collision-Resistant Hash Functions) and 20 of 64 extended hash functions are CRHFs.

Motivation for Our Study. Examples of the most commonly used collision-resistant hash functions are MD5 and SHA-1. For such hash functions, one cannot exactly analyze security. However, the security of collision-resistant or one-way PGV hash functions can be analyzed under the assumption that the underlying block cipher is a black box, i.e., random permutation. However, the security of other notions like target collision resistance cannot be analyzed because it needs a family of hash functions instead of a single hash function. Moreover, it seemed that more PGV hash functions will be secure if we change the original definition of the PGV hash function. Thus, we generalize the definition of the PGV hash function to mean a PGV hash family and prove some security notions like target collision resistance, collision resistance and one-way-ness.

General Definition of PGV hash family. Let $0 \le l < n$ and $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. If $l = 0$, let $\{0,1\}^0 = \{\epsilon\}$, where ϵ is the empty string. Using the block cipher E, we want to construct the compression function family $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^l}$, $f^k : \{0,1\}^n \times \{0,1\}^{n-l} \to \{0,1\}^n$.

Let $h_0, v \in \{0,1\}^n$ be fixed values. We define the 64 ways to construct a (*block-cipher-based*) *compression function family* $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^l}$ in the following manner: for each $k \in \{0,1\}^l$,

$$f^k(h, m) = E_a(b) \oplus c,$$

where $a, b, c \in \{h, (m\|k), h \oplus (m\|k), v\}$. Note that $|h| = n$ and $|m| = n - l$. Then we can define the *extended hash family* $\mathcal{H} = \{H^k\}_{k\in\{0,1\}^l}$ from the compression function family $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^l}$ as follows: for each $k \in \{0,1\}^l$, $H^k : (\{0,1\}^{n-l})^* \to \{0,1\}^n$ is defined by

function $H^k(m_1 \cdots m_t)$
for $i \leftarrow 1$ **to** t **do** $h_i \leftarrow f^k(h_{i-1}, m_i)$
return h_t.

Note that the key k of the extended hash family is equal to the key of the compression function family.

Note that if $l = 0$, then $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^0} = \{f^\epsilon\}$ is a singleton set corresponding to the original definition of PGV [2]. In this case, we denote $\mathcal{F}$ as just f without the superscript ϵ. We call this f a (*block-cipher-based*) *compression function*. Similarly, we denote $\mathcal{H}$ as H without the superscript ϵ. We call this H an *extended hash function*.

Results. For $0 < l < n$, the security of the 64 schemes is summarized in Table 1, which also serve to define the different extended hash functions H_t and their compression functions f_t. In this paper, we fix E1 = $\{1, ..., 20\}$, E2 = $\{21, 22, 26, 28\}$, E3= $\{23, 24, 25, 31, 34, 35\}$, E4= $\{27, 29, 30, 32, 33, 36\}$, and E5 = $\{37, ..., 42\}$. Here, the numbers correspond to the numbers in the first column of Table 1. E6 is a set of the remaining extended hash families that are not represented in the first column of Table 1. Thus, |E6| = 22. This classification is based on some property of

Manuscript received March 22, 2004.
Manuscript revised July 7, 2004.
Final manuscript received September 15, 2004.

[†]The authors are with the Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka-shi, 812-8581 Japan.
[††]The authors are with Applied Statistics Unit, Indian Statistical Institute, Kolkata, India.
[†††]The authors are with Center for Information Security Technologies (CIST), Korea University, Seoul, Korea.
a) E-mail: sangjin@korea.ac.kr

Table 1 Summary of results of 64 extended hash families. Column 1 shows our number ι for the function family (We write $\mathcal{F}_\iota$ for the compression function family and $\mathcal{H}_\iota$ for its induced extended hash family). Column 2 shows the number from [2]. Column 3 defines $f_k(h_{i-1}, m_i)$ for some $k \in \{0,1\}^l$. We write x_i for $(m_i \| k)$ and w_i for $x_i \oplus h_{i-1}$. Columns 4 and 5 show our (target) collision resistance bounds. Columns 6 and 7 show our inversion resistance bounds. Note that there is a restriction on q for some cases (See Theorem 2).

ι	$\jmath$	$h_i =$	(T)CR LB	(T)CR UB	IR LB	IR UB
	1	$E_{x_i}(x_i) \oplus v$	1	1	–	–
22	2	$E_{h_{i-1}}(x_i) \oplus v$	$q/2^{l+1}$	$2q/(2^{l+1}-1)$	$q/2^{l+1}$	$q/2^{l-1}$
13	3	$E_{w_i}(x_i) \oplus v$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$q/2^l$	$q/2^{l-1}$
	4	$E_v(x_i) \oplus v$	1	1	–	–
	5	$E_{x_i}(x_i) \oplus x_i$	1	1	–	–
1	6	$E_{h_{i-1}}(x_i) \oplus x_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
9	7	$E_{w_i}(x_i) \oplus x_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
	8	$E_v(x_i) \oplus x_i$	1	1	–	–
	9	$E_{x_i}(x_i) \oplus h_{i-1}$	1	1	–	–
21	10	$E_{h_{i-1}}(x_i) \oplus h_{i-1}$	$q/2^{l+1}$	$2q/(2^{l+1}-1)$	$q/2^{l+1}$	$q/2^{l-1}$
11	11	$E_{w_i}(x_i) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
	12	$E_v(x_i) \oplus h_{i-1}$	1	1	–	–
	13	$E_{x_i}(x_i) \oplus w_i$	1	1	–	–
3	14	$E_{h_{i-1}}(x_i) \oplus w_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
14	15	$E_{w_i}(x_i) \oplus w_i$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$q/2^l$	$q/2^{l-1}$
	16	$E_v(x_i) \oplus w_i$	1	1	–	–
15	17	$E_{x_i}(h_{i-1}) \oplus v$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$.15q^2/2^n$	$9(q+3)^2/2^n$
	18	$E_{h_{i-1}}(h_{i-1}) \oplus v$	1	1	–	–
16	19	$E_{w_i}(h_{i-1}) \oplus v$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$q/2^l$	$q/2^{l-1}$
	20	$E_v(h_{i-1}) \oplus v$	1	1	–	–
17	21	$E_{x_i}(h_{i-1}) \oplus x_i$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$.15q^2/2^n$	$9(q+3)^2/2^n$
23	22	$E_{h_{i-1}}(h_{i-1}) \oplus x_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
12	23	$E_{w_i}(h_{i-1}) \oplus x_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
35	24	$E_v(h_{i-1}) \oplus x_i$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$.15q^2/2^l$	$q^2/2^{l-1}$
5	25	$E_{m_i}(h_{i-1}) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
	26	$E_{h_{i-1}}(h_{i-1}) \oplus h_{i-1}$	1	1	–	–
10	27	$E_{w_i}(h_{i-1}) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
	28	$E_v(h_{i-1}) \oplus h_{i-1}$	1	1	–	–
7	29	$E_{x_i}(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
24	30	$E_{h_{i-1}}(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
18	31	$E_{w_i}(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$q/2^l$	$q/2^{l-1}$
25	32	$E_v(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
19	33	$E_{x_i}(w_i) \oplus v$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$.15q^2/2^n$	$9(q+3)^2/2^n$
26	34	$E_{h_{i-1}}(w_i) \oplus v$	$q/2^{l+1}$	$2q/(2^{l+1}-1)$	$q/2^{l+1}$	$q/2^{l-1}$
38	35	$E_{w_i}(w_i) \oplus v$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^n$	$q/2^{n-1}$
37	36	$E_v(w_i) \oplus v$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$.15q^2/2^l$	$q^2/2^{l-1}$
20	37	$E_{x_i}(w_i) \oplus x_i$	$.3q(q-1)/2^n$	$q^2/2^{n-1}$	$.15q^2/2^n$	$9(q+3)^2/2^n$
4	38	$E_{h_{i-1}}(w_i) \oplus x_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
27	39	$E_{w_i}(w_i) \oplus x_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
36	40	$E_v(w_i) \oplus x_i$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$.15q^2/2^l$	$q^2/2^{l-1}$
8	41	$E_{x_i}(w_i) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
28	42	$E_{h_{i-1}}(w_i) \oplus h_{i-1}$	$q/2^{l+1}$	$2q/(2^{l+1}-1)$	$q/2^{l+1}$	$q/2^{l-1}$
29	43	$E_{w_i}(w_i) \oplus h_{i-1}$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
30	44	$E_v(w_i) \oplus h_{i-1}$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
6	45	$E_{x_i}(w_i) \oplus w_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
2	46	$E_{h_{i-1}}(w_i) \oplus w_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$.4q/2^n$	$2q/2^n$
39	47	$E_{w_i}(w_i) \oplus w_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^n$	$q/2^{n-1}$
40	48	$E_v(w_i) \oplus w_i$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$q/2^n$	$q/2^{n-1}$
	49	$E_{x_i}(v) \oplus v$	1	1	–	–
	50	$E_{h_{i-1}}(v) \oplus v$	1	1	–	–
41	51	$E_{w_i}(v) \oplus v$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^n$	$q/2^{n-1}$
	52	$E_v(v) \oplus v$	1	1	–	–
	53	$E_{x_i}(v) \oplus x_i$	1	1	–	–
31	54	$E_{h_{i-1}}(v) \oplus x_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
32	55	$E_{w_i}(v) \oplus x_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
	56	$E_v(v) \oplus x_i$	1	1	–	–
	57	$E_{x_i}(v) \oplus h_{i-1}$	1	1	–	–
	58	$E_{h_{i-1}}(v) \oplus h_{i-1}$	1	1	–	–
33	59	$E_{w_i}(v) \oplus h_{i-1}$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
	60	$E_v(v) \oplus h_{i-1}$	1	1	–	–
	61	$E_{x_i}(v) \oplus w_i$	1	1	–	–
34	62	$E_{h_{i-1}}(v) \oplus w_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$
42	63	$E_{w_i}(v) \oplus w_i$	$.3q(q-1)/2^l$	$q^2/2^{l-1}$	$q/2^n$	$q/2^{n-1}$
	64	$E_v(v) \oplus w_i$	1	1	–	–

the hash family that is used to prove the security. A high-level summary is given in Tables 2 and 3. The adversarial model (and the definition of q) will be described below. It should be noted that there exists a trade-off between the size of l and efficiency. If l is large, then we can obtain better security but we lose efficiency.

Table 2 $l = 0$. This is analyzed in [1]. EHF = extended hash family, (T)CB= (target) collision bound, and IB= inversion bound.

EHF	(T)CB	IB
E1	$\Theta(q^2/2^n)$	$\Theta(q/2^n)$ or $\Theta(q^2/2^n)$
E2	$\Theta(1)$	–
E3/E4/E5	$\Theta(1)$	–
E6	$\Theta(1)$	–

Table 3 $0 < l < n$. This is analyzed in this paper. Abbreviations are the same as those in Table 2.

EHF	(T)CB	IB
E1	$\Theta(q^2/2^n)$	$\Theta(q/2^l)$ or $\Theta(q/2^n)$ or $\Theta(q^2/2^n)$
E2	$\Theta(q/2^l)$	$\Theta(q/2^l)$
E3/E4/E5	$\Theta(q^2/2^l)$	$\Theta(q/2^l)$ or $\Theta(q^2/2^l)$ or $\Theta(q/2^n)$
E6	$\Theta(1)$	–

Black-Box Model. Our security model is the one dating to Shannon [6] and used for works like [3]–[5]. The adversary $\mathcal{A}$ is given access to oracles E and E^{-1} where E is a random block cipher $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ and E^{-1} is its inverse. That is, each key $a \in \{0,1\}^n$ names a randomly selected permutation $E_a = E(a,\cdot)$ on $\{0,1\}^n$, and the adversary is given oracles E and E^{-1}. The latter, on input (a,y), returns a point x such that $E_a(x) = y$. See [1] for more details and discussions about the black-box model.

In the above PGV hash function families, we do not use any mask keys unlike in [7], [10], [12], and [13]. We prove the target collision resistance of these hash families under the black-box model and it will be more efficient in terms of key size compared with the results in [7], [10], [12], and [13] wherein mask keys are used.

2. Preliminary

Notation. We use the following standard notations in this paper.

1. $[a,b] = \{a, \cdots, b\}$ where $a \le b$ and a, b are integers.
2. If $x \in \{0,1\}^n$ and $0 \le l < n$, $x = x[L]\|x[R]$, where $|x[L]| = n - l$ and $|x[R]| = l$.
3. If $S \subseteq \{0,1\}^n$ and $a \in \{0,1\}^n$, $S \oplus a = a \oplus S = \{a \oplus s | s \in S\}$. Note that $|S \oplus a| = |a \oplus S| = |S|$.
4. A *block cipher* is a map $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ where, for each key $a \in \{0,1\}^n$, the function $E_a(\cdot) = E(a,\cdot)$ is a permutation on $\{0,1\}^n$. If E is a block cipher then E^{-1} is its inverse, where $E_a^{-1}(y)$ is the string x such that $E_a(x) = y$.
5. A *hash function family* is a $\mathcal{H} = \{H^k\}_{k\in\{0,1\}^l}$, where $H^k : D \to \{0,1\}^n$, $D \subseteq \{0,1\}^*$.
6. Hash function family $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^l}$, $f^k : D \to \{0,1\}^n$ is a *compression function family* if $D = \{0,1\}^n \times \{0,1\}^{n-l}$ for some fixed l.
7. Fix $h_0 \in \{0,1\}^n$. The *extended hash family* of compression function family $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^l}$, $f^k : \{0,1\}^n \times \{0,1\}^{n-l} \to \{0,1\}^n$, is the hash function family $\mathcal{H} = \{H^k\}_{k\in\{0,1\}^l}$ such that $H^k : (\{0,1\}^{n-l})^* \to \{0,1\}^n$ defined by $H^k(m_1 \cdots m_t) = h_t$, where $h_i = f^k(h_{i-1}, m_i)$.
8. For the function H, (M, M') is called a *collision pair* of H if $M \neq M'$ and $H(M) = H(M')$.
9. We write $x \xleftarrow{R} S$ for the experiment on choosing a random element from the finite set S and calling it x.

Assumption. From now on, we will always assume $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a random block cipher, i.e., for each $a \in \{0,1\}^n$, $E_a(\cdot)$ is a random permutation. Adversaries are probabilistic algorithms. Thus, every probability in this paper is based on the randomness of the block cipher and random coins. We fix $h_0, v \in \{0,1\}^n$.

Collision resistance and Inversion resistance of hash function ($l = 0$). To quantify the collision resistance of the (block-cipher-based) hash function H, we consider the random block cipher E. An adversary $\mathcal{A}$ is given oracles for $E(\cdot,\cdot)$ and $E^{-1}(\cdot,\cdot)$ and wants to find a collision for H, i.e., M, M' where $M \neq M'$ but $H(M) = H(M')$. We also define the difficulty in inverting hash functions. We use the following measure for the difficulty of $\mathcal{A}$ in inverting a hash function at a random point.

Definition 1: (Collision resistance and inversion resistance of the compression function 'f') Let f be a block-cipher-based compression function, $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$. Then the advantages of $\mathcal{A}$ in finding collisions and inverse elements in f are

$$\begin{aligned}\mathbf{Adv}_f^{Coll}(\mathcal{A}) &= \Pr[((h,m),(h',m')) \leftarrow \mathcal{A}^{E,E^{-1}} : \\ &\quad ((h,m) \neq (h',m') \& f(h,m) = f(h',m')) \\ &\quad or\ f(h,m) = h_0] \\ \mathbf{Adv}_f^{Inv}(\mathcal{A}) &= \Pr[h^* \xleftarrow{R} \{0,1\}^n; (h,m) \leftarrow \mathcal{A}^{E,E^{-1}}(h^*) : \\ &\quad f(h,m) = h^*].\end{aligned}$$

Definition 2: (Collision resistance and inversion resistance of the extended hash function 'H') Let H be a block-cipher-based extended hash function, $H : (\{0,1\}^n)^* \to \{0,1\}^n$. Then the advantages of $\mathcal{A}$ in finding collisions and inverse elements in H are

$$\begin{aligned}\mathbf{Adv}_H^{Coll}(\mathcal{A}) &= \Pr[(M,M') \leftarrow \mathcal{A}^{E,E^{-1}} : \\ &\quad M \neq M' \ \&\ H(M) = H(M')] \\ \mathbf{Adv}_H^{Inv}(\mathcal{A}) &= \Pr[h^* \xleftarrow{R} \{0,1\}^n; M \leftarrow \mathcal{A}^{E,E^{-1}}(h^*) : \\ &\quad H(M) = h^*].\end{aligned}$$

Collision resistance, Target collision resistance and Inversion resistance of hash function family ($0 < l < n$). To quantify the collision resistance and target collision resistance of the (block-cipher-based) hash function family $\{H^k\}_{k\in\{0,1\}^l}$, we consider the random block cipher E. The adversary $\mathcal{A}$ is given oracles for $E(\cdot,\cdot)$ and $E^{-1}(\cdot,\cdot)$. Then, the adversary $\mathcal{A}^{E,E^{-1}}$ for collision resistance plays the following game called *Coll*.

1. $\mathcal{A}^{E,E^{-1}}$ is given the key k which is chosen uniformly at random from $\{0,1\}^l$.
2. $\mathcal{A}^{E,E^{-1}}$ has to find M, M' such that $M \neq M'$ but $H_k(M) = H_k(M')$.

The adversary $\mathcal{A}^{E,E^{-1}} = (\mathcal{A}_{guess}, \mathcal{A}_{find}(\cdot,\cdot))$ for target

collision resistance plays the following game called $TColl$.

1. $\mathcal{A}_{guess}$ commits to M.
2. The key k is chosen uniformly at random from $\{0,1\}^l$.
3. $\mathcal{A}_{find}(M,k)$ has to find M' such that $M \neq M'$ but $H_k(M) = H_k(M')$.

The adversary $\mathcal{A}^{E,E^{-1}}$ for inversion resistance plays the following game called Inv.

1. The key k is chosen uniformly at random from $\{0,1\}^l$.
2. h^* is chosen uniformly at random from the range $\{0,1\}^n$.
3. $\mathcal{A}^{E,E^{-1}}$ try to find M such that $H^k(M) = h^*$.

Definition 3: (Collision resistance, target collision resistance, and inversion resistance of the compression function family '$\mathcal{F}$') Let $\mathcal{F} = \{f^k\}_{k\in\{0,1\}^l}$ be a block-cipher-based compression function family, where $f^k : \{0,1\}^n \times \{0,1\}^{n-l} \to \{0,1\}^n$. Then the advantages of $\mathcal{A}$ with respect to (target) collision resistance and inversion resistance are the following real numbers.

$$\begin{aligned}\mathbf{Adv}^{Coll}_{\mathcal{F}}(\mathcal{A}) &= \Pr[k \xleftarrow{R} \{0,1\}^l; ((h,m),(h',m')) \leftarrow \\ &\quad \mathcal{A}^{E,E^{-1}}(k) : ((h,m) \neq (h',m') \ \&\ f^k(h,m) \\ &\quad = f^k(h',m')) \ or \ f^k(h,m) = h_0]\end{aligned}$$

$$\begin{aligned}\mathbf{Adv}^{TColl}_{\mathcal{F}}(\mathcal{A}) &= \Pr[(h,m) \leftarrow \mathcal{A}^{E,E^{-1}}_{guess}; k \xleftarrow{R} \{0,1\}^l; \\ &\quad (h',m') \leftarrow \mathcal{A}^{E,E^{-1}}_{find}((h\|m),k) : (h,m) \\ &\quad \neq (h',m') \ \&\ f^k(h,m) = f^k(h',m')]\end{aligned}$$

$$\begin{aligned}\mathbf{Adv}^{Inv}_{\mathcal{F}}(\mathcal{A}) &= \Pr[k \xleftarrow{R} \{0,1\}^l; h^* \xleftarrow{R} \{0,1\}^n; \\ &\quad (h,m) \leftarrow \mathcal{A}^{E,E^{-1}}(h^*,k) : f^k(h,m) = h^*].\end{aligned}$$

Definition 4: (Collision resistance, target collision resistance, and inversion resistance of the extended hash family '$\mathcal{H}$') Let $\mathcal{H} = \{H^k\}_{k\in\{0,1\}^l}$ be a block-cipher-based extended hash family, where $H^k : (\{0,1\}^{n-l})^* \to \{0,1\}^n$. Then the advantage of $\mathcal{A}$ with respect to (target) collision resistance and inversion resistance are the following real numbers.

$$\begin{aligned}\mathbf{Adv}^{Coll}_{\mathcal{H}}(\mathcal{A}) &= \Pr[k \xleftarrow{R} \{0,1\}^l; M, M' \leftarrow \mathcal{A}^{E,E^{-1}}(k) : \\ &\quad M \neq M' \ \&\ H^k(M) = H^k(M')]\end{aligned}$$

$$\begin{aligned}\mathbf{Adv}^{TColl}_{\mathcal{H}}(\mathcal{A}) &= \Pr[M \leftarrow \mathcal{A}^{E,E^{-1}}_{guess}; k \xleftarrow{R} \{0,1\}^l; \\ &\quad M' \leftarrow \mathcal{A}^{E,E^{-1}}_{find}(M,k) : M \neq M' \\ &\quad \ \&\ H^k(M) = H^k(M')]\end{aligned}$$

$$\begin{aligned}\mathbf{Adv}^{Inv}_{\mathcal{H}}(\mathcal{A}) &= \Pr[k \xleftarrow{R} \{0,1\}^l; h^* \xleftarrow{R} \{0,1\}^n; \\ &\quad M \leftarrow \mathcal{A}^{E,E^{-1}}(h^*,k) : H^k(M) = h^*].\end{aligned}$$

Maximal Advantage. If $\mathcal{A}$ is an adversary and $\mathbf{Adv}^{XXX}_{Y}(\mathcal{A})$ is a measure of the adversarial advantage already defined, then we write $\mathbf{Adv}^{XXX}_{Y}(q)$ to mean the maximal value of $\mathbf{Adv}^{XXX}_{Y}(\mathcal{A})$ over all adversaries $\mathcal{A}$ that use queries bounded by the number q.

Conventions. We follow conventions similar to those in [1]. Note that these conventions are important for facilitating discussion and proving the following theorems. In the rest of this paper, we assume the following significant conventions.

1. First, an adversary does not ask any oracle query in which the response is already known; namely, if $\mathcal{A}$ asks a query $E_a(x)$ and oracle returns y, then $\mathcal{A}$ does not ask a subsequent query of $E_a(x)$ or $E_a^{-1}(y)$; and if $\mathcal{A}$ asks $E_a^{-1}(y)$ and oracle returns x, then $\mathcal{A}$ does not ask a subsequent query of $E_a^{-1}(y)$ or $E_a(x)$.
2. Second, if M is one of the outputs produced by an adversary, then the adversary should make necessary E/E^{-1} queries to compute $H^k(M)$ during the whole query process.
3. Similarly, we use the same assumption regarding the oracle query process of an adversary $\mathcal{A}$ for the compression function family $\mathcal{F}$.

These assumptions are all without loss of generality in that the adversary $\mathcal{A}$ not obeying these conventions can easily be modified to the adversary $\mathcal{A}'$ having a similar computational complexity that obeys these conventions and has the same advantage as $\mathcal{A}$.

3. (Target) Collision Resistance of Extended Hash Family

In this section, we will analyze the security of $\mathcal{H}_\iota$ for each $\iota \in [1,42]$ defined in Section 1 in the notion of (target) collision resistance. We consider any adversary $\mathcal{A}$ with respect to Coll, i.e., after obtaining random key k, he will try to find a collision pair (M_1, M_2) for H^k_ι, i.e., $M_1 \neq M_2$, $H^k_\iota(M_1) = H^k_\iota(M_2)$. For that, he will make some E/E^{-1} queries. The *transcript* of $\mathcal{A}$ is defined by the sequence of query-response quadruples $\{(s_i, x_i, y_i, \sigma_i)\}_{1\le i\le q}$ where q is the maximum number of queries made by the adversary, $s_i, x_i, y_i \in \{0,1\}^n$ and $\sigma_i = +1$ (in case of E-query) or -1 (in case of E^{-1}-query) and $E_{s_i}(x_i) = y_i$. $(s_i, x_i, y_i, \sigma_i)$ will be called the i^{th} query-response quadruple (or q-r quadruple). In this section, we fix some keys k and v. Note that, if $\sigma_i = +1$ (or -1) then y (or x respectively) is a random string as we assume that the block cipher $E_s(\cdot)$ is a random permutation.

Proposition 1: For fixed $x, y \in \{0,1\}^n$ and $A \subseteq \{0,1\}^n$, $\Pr[y_i = y] \le \frac{1}{2^n - i + 1}$ and $\Pr[y_i \in A] \le \frac{|A|}{2^n - i + 1}$ whenever $\sigma_i = +1$. Similarly, if $\sigma_i = -1$ then $\Pr[x_i = x] \le \frac{1}{2^n - i + 1}$ and $\Pr[x_i \in A] \le \frac{|A|}{2^n - i + 1}$

Proof. Before the i^{th} query, at most $(i-1)$ outputs (or inputs) of a block cipher with same key are known. Thus, output (or input) of the next E will be uniformly distributed to at least $2^n - (i-1)$ elements. □

Here, we fix any arbitrary hash family $\mathcal{H}_\iota$ for $\iota \in [1,42]$. In this section, $V := \{0,1\}^n$ is called the *vertex set* and $L := \{0,1\}^{n-l}$ the *label set*. A triple $(h_1, h_2, m) \in V \times V \times L$ (or a pair $(h_1, h_2) \in V \times V$) is called the *labeled arc* (or an

arc only). We also say that (h_1, h_2, m) is an arc (h_1, h_2) with the label m, or m is a label of the arc (h_1, h_2) and we use the notation $h_1 \rightarrow_m h_2$. Now, given a triple $\tau = (s, x, y)$, where, $s, x, y \in V$, we define a set of labeled arcs $A(\tau)$ by

$$A(\tau) = \{(h_1, h_2, m) \in V \times V \times L : f^k(h_1, m) = h_2 \Leftrightarrow E_s(x) = y\}.$$

For example, in the case of $\mathcal{H}_{21}$, $f^k_{21}(h_1, m) := E_{h_1}(m||k) \oplus h_1$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_s(x) = y) \Longleftrightarrow (E_{h_1}(m||k) \oplus h_1 = h_2 \Leftrightarrow E_s(x) = y) \Longleftrightarrow (h_1 = s, h_2 = y \oplus h_1 = y \oplus s, m||k = x)$. Hence, $A(\tau) = \{(s, s \oplus y, x[L])\}$ if $x[R] = k$, otherwise it is an empty set.

Given a set of labeled arcs A, we define induced arc set $A' = \{(h_1, h_2) : \exists m \in L, (h_1, h_2, m) \in A\}$. For a set of triple(s) $\tau = \{\tau_1 = (s_1, x_1, y_1), \ldots, \tau_a = (s_a, x_a, y_a)\}$, we can define the *labeled arc set* $A(\tau) = \bigcup_{i=1}^{a} A(\tau_i)$. It can be easily checked that $A'(\tau) = \bigcup_{i=1}^{a} A'(\tau_i)$. Every member of $A(\tau)$ (or $A'(\tau)$) will be called the *labeled arc* (or *arc*) *corresponding* to the set of triple(s) τ. Given a transcript $\{(s_i, x_i, y_i, \sigma_i)\}_{1 \le i \le q}$ of an adversary $\mathcal{A}$, let $\tau[i]$ denote the sets of triples $\{\tau_1 = (s_1, x_1, y_1), \ldots, \tau_i = (s_i, x_i, y_i)\}$. For each i, we have a labeled directed graph $T_i = T(\tau[i]) = (V, A(\tau[i]))$ and a directed graph $T'_i = (V, A'(\tau[i]))$. Define $T_0 = (V, \emptyset)$. Given a path $P = (h_1, h_2, \ldots, h_p)$ from h_1 to h_p in T_i, $M = m_1 || \ldots || m_{p-1}$ is called a label of P if m_i is a label of (h_i, h_{i-1}) for each i. So we have a picture like $(h_1 \rightarrow_{m_1} h_2 \rightarrow_{m_2} \ldots \rightarrow_{m_{p-1}} h_p)$ in T_i.

Observation 1 : By our conventions, the adversary can compute $f^k_t(h_1, m) = h_2$ after the i^{th} query iff for some $j \le i$, $E_{s_j}(x_j) = y_j \Rightarrow f^k_t(h_1, m) = h_2$, and hence $(h_1, h_2, m) \in A(\tau[i])$. Similarly, the adversary can compute $H^k_t(m_1 || \cdots || m_a)$ after the i^{th} query iff $h_0 \rightarrow_{m_1} h_1 \rightarrow_{m_2} .. \rightarrow_{m_a} h_a$ is a path in $A(\tau[i])$ and $H^k_t(m_1 || \cdots || m_a) = h_a$.

Definition 5: When $\iota \in$ E1, E2 or E4, h in T_i is **old** if $deg(h) \ge 1$ in T_i or $h = h_0$. When $\iota \in$ E2 or E4, h in T_i is **old** if $h = h_0$ or there exists an h_1 such that $deg(h_1) \ge 1$ in T_i and $h[R] = h_1[R]$. Here, $deg(h) = indeg(h) + outdeg(h)$. The other remaining vertices are known as **new** vertices. Here, we call the set of all *old* vertices in T_i, O_i.

The next proposition will be used for security analysis. It gives an upper bound of $|O_i|$ and indicates the structure of the set of labeled arcs $A(\tau_i)$ and $A'(\tau_i)$.

Proposition 2: If $A(\tau_i)$ is not empty then we have the following.

1. For $\iota \in$ E1 or E2, $A(\tau_i)$ is a singleton and $|O_i| \le 2i + 1$.

2. For $\iota \in$ E3, $A'(\tau_i) = \{(h_1, h_2) : h_2[R] = u\}$, where h_1 and u are fixed depending only on j and τ_i. Thus, the graph of $A'(\tau_i)$ resembles an outward directed star and $|A'(\tau_i)| = 2^{n-l} = |A(\tau_i)|$ and hence $|O_i| \le (2i + 1)2^{n-l}$.

3. For $\iota \in$ E4, $A'(\tau_i) = \{(h, h \oplus a) : h[R] = u\}$, where a and u are fixed depending only on j and τ_i. Thus, the graph of $A'(\tau_i)$ consists of 2^{n-l} parallel arcs and $|A'(\tau_i)| = 2^{n-l} = |A(\tau_i)|$, and hence $|O_i| \le (2i + 1)2^{n-l}$.

4. For $\iota \in$ E5, $A'(\tau_i) = \{(h_1, h_2) : h_1[R] = u\}$, where h_2 and u are fixed depending only on j and τ_i. Thus, the graph of the $A'(\tau_i)$ resembles an inward directed star and $|A'(\tau_i)| = 2^{n-l} = |A(\tau_i)|$ and hence $|O_i| \le (2i + 1)2^{n-l}$.

Moreover, for each $(h_1, h_2) \in A'(\tau_i)$, there exists a unique m such that $h_1 \rightarrow_m h_2$. For the hash families E3, E4 and E5, if $h_1[R] = h_2[R]$, then $h_1 \in O_i \Rightarrow h_2 \in O_i$ for all i.

Proof. Bounds for $|O_i|$'s and the last part of the proposition are straightforward from the structure of $A'(\tau_i)$. We will prove this for one hash function from each class. Other cases will be very similar and one can check analogously. Let $\tau_i = (s_i, x_i, y_i)$.

1. In the case of $\mathcal{H}_1$, $f^k_1(h_1, m) := E_{h_1}(m||k) \oplus (m||k)$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \Longleftrightarrow (E_{h_1}(m||k) \oplus (m||k) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \Longleftrightarrow (h_1 = s_i, h_2 = y_i \oplus (m||k), x_i = m||k)$. Hence, $A(\tau) = \{(s_i, y_i \oplus x_i, x_i[L])\}$ if $x_i[R] = k$, otherwise it is an empty set.

   In the case of $\mathcal{H}_{21}$, after defining $A(\tau)$ in this section, we have shown that $A(\tau) = \{(s_i, s_i \oplus y_i, x_i[L])\}$ if $x_i[R] = k$, otherwise it is an empty set.

2. In the case of $\mathcal{H}_{23}$, $f^k_{23}(h_1, m) := E_{h_1}(h_1) \oplus (m||k)$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \Longleftrightarrow (E_{h_1}(h_1) \oplus (m||k) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \Longleftrightarrow (h_1 = s_i = x_i, h_2 = y_i \oplus (m||k))$. Hence, $A(\tau) = \{(s_i, h_2, m) : h_2[R] = y_i[R] \oplus k, m = h_2[L] \oplus y_i[L]\}$ if $x_i = s_i$, otherwise it is an empty set.

3. In the case of $\mathcal{H}_{27}$, $f^k_{27}(h_1, m) := E_{w_1}(w_1) \oplus (m||k)$ where $w_1 = h_1 \oplus (m||k)$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \Longleftrightarrow (E_{w_1}(w_1) \oplus (m||k) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \Longleftrightarrow (h_1 = s_i \oplus (m||k), h_2 = y_i \oplus (m||k) = h_1 \oplus (y_i \oplus s_i), s_i = x_i)$. Hence, $A(\tau) = \{(h_1, h_1 \oplus (s_i \oplus y_i), x_i[L] \oplus h_1[R])\}$ if $x_i = s_i$, otherwise it is an empty set.

4. In the case of $\mathcal{H}_{37}$, we can similarly prove that $A(\tau_i) = \{(h_1, y_i \oplus v, m) : h_1[R] = s_i[R] \oplus k, m = h_1[L] \oplus s_i\}$ if $x_i = s_i$, otherwise it is an empty set. □

Definition 6: For each $1 \le i \le q$, we define some events.

1. C_i : the adversary gets a collision after i^{th} query.
2. $\mathsf{PathColl}_i$: there exist two paths P_1 and P_2 (not necessarily distinct) from h_0 to some h^* in T_i such that P_1 and P_2 have two different labels.
3. Succ_i : there exists an arc $(h, h') \in A'(\tau_i)$, where both h and h' are old vertices in T_{i-1}.

Proposition 3: The event $\mathsf{PathColl}_i$ is equivalent to C_i.

Proof. $\mathsf{C}_i \Leftrightarrow \mathsf{PathColl}_i$ can be proved using the last part of Observation 1.

Proposition 4: For E1, E2, E3, and E4 hash families, the conditional event $(\mathsf{C}_i \mid \neg\mathsf{C}_{i-1})$ necessarily implies Succ_i. For $E5$, C_i necessarily implies $\mathsf{Succ}_{i'}$ for some $i' \leq i$.

Proof. Let P_1 and P_2 be two distinct paths from h_0 to h^* in T'_i with different labels for some h^*. As $\mathsf{PathColl}_{i-1}$ is not true, there exists at least one arc in $P_1 \cup P_2$ which corresponds to τ_i. If Succ_i is not true, then one of the vertices of an arc corresponding to τ_i should be new in T_{i-1} which implies that there exist two arcs either $(h_1, h_2), (h_2, h_3)$ or $(h_1, h_3), (h_2, h_3)$ corresponding to τ_i. However, this is not possible by the structure of $A'(\tau_i)$ (see Proposition 2) in the cases of E1, E2, E3 and E4 hash families. Similarly we can prove it when $P_1 = P_2$.

In the case of the E5 hash function for $P_1 = P_2$, the proof is similar as $(h_1, h_3), (h_2, h_3)$ case will not arise. Thus, assume that P_1 and P_2 are different and there exist (h_1, h_3) and (h_2, h_3) corresponding to τ_i in the path $P_1 \cup P_2$. By Proposition 2, $h_1[R] = h_2[R]$. If Succ_i is not true but $(\mathsf{PathColl}_i|\neg\mathsf{PathColl}_{i-1})$ is true, then we have two paths P'_1 and P'_2 in T_{i-1} from h_0 to $h_a = h_1$ and $h'_b = h_2$, respectively. Let $P'_1 = (h_0 \rightarrow h_1 \rightarrow \ldots \rightarrow h_a)$ and $P'_2 = (h_0 \rightarrow h'_1 \rightarrow \ldots \rightarrow h'_b)$. Thus, if $\mathsf{Succ}_{i'}$ is not true for all i' such that $1 \leq i' \leq i$, then at least one new vertex from $P'_1 \cup P'_2$ is added to O_j for each j whenever it is added. As there are new $a+b$ vertices for T_0 in $P'_1 \cup P'_2$ and at most one arc can be added to $A_j(\tau_{i'})$ every time (because of the structure of $A_j(\tau_{i'})$) we have to add exactly one new vertex in each i', because $h_1[R] = h_2[R]$. Thus, we will add two new vertices in $P'_1 \cup P'_2$ to a set of old vertices when we add h_1 or h_2 first time and hence contradiction. □

Observation 2: In E5, $\mathsf{C}_q \Rightarrow \bigvee_{i=1}^{q} \mathsf{Succ}_i$ by the above Proposition 4. Thus, we have $\Pr[\mathcal{A}$ gets a collision$] \leq \sum_{i=1}^{q} \Pr[\mathsf{Succ}_i]$. In other hash families, by the above Proposition 4, $\Pr[\mathcal{A}$ gets a collision$] \leq \sum_{i=1}^{q} \Pr[\mathsf{C}_i|\neg\mathsf{C}_{i-1}] \leq \sum_{i=1}^{q} \Pr[\mathsf{Succ}_i]$. Thus, it is sufficient to have an upper bound of $\Pr[\mathsf{Succ}_i]$ in all hash functions.

Theorem 1: For each $1 \leq i \leq q$, we have the following.

1. For the E1 hash family, $\Pr[\mathsf{Succ}_i] \leq (2i-1)/2^{n-1}$.
2. For the E2 hash family, $\Pr[\mathsf{Succ}_i] \leq 2/(2^{l+1}-1)$ if $q \leq 2^{n-l-1}$.
3. For the E3,E4 or E5 hash families, $\Pr[\mathsf{Succ}_i] \leq (2i-1)/2^{l-1}$.

Proof. Let $\mathcal{A}$ be an adversary attacking $\mathcal{H}_\iota$. Assume that $\mathcal{A}$ asks its oracles at most q queries. Assume that the random key k is given. Let $(s_i, x_i, y_i, \sigma_i)$ be the i^{th} q-r quadruple.

Consider H_1^k in the case of the E1 hash family. For the other hash families in E1, the proof is analogous to the proof of H_1^k.

1. Case 1: $\sigma_i = +1$. $\mathsf{Succ}_i \Rightarrow y_i \oplus x_i \in O_{i-1}$ (See Proposition 2). Thus, $\Pr[\mathsf{Succ}_i] \leq \Pr[y_i \in O_{i-1} \oplus x_i] \leq (2i-1)/(2^n - i + 1)$ (by Propositions 1 and 2).
2. Case 2: $\sigma_i = -1$. $\mathsf{Succ}_i \Rightarrow y_i \oplus x_i \in O_{i-1}$ (See Propositions 2). Hence, $\Pr[\mathsf{Succ}_i] \leq \Pr[x_i \in O_{i-1} \oplus y_i] \leq (2i-1)/(2^n - i + 1)$ (by Proposition 1 and 2).

Therefore, $\Pr[\mathsf{Succ}_i] \leq (2i-1)/(2^n - i + 1) \leq (2i-1)/2^{n-1}$. Consider H_{21}^k in the case of the E2 hash family. For the other hash families in E2, the proof is analogous to the proof of 21.

1. Case 1: $\sigma_i = +1$. $\mathsf{Succ}_i \Rightarrow y_i \oplus s_i \in O_{i-1}$ (See Proposition 2). Hence, $\Pr[\mathsf{Succ}_i] \leq \Pr[y_i \in O_{i-1} \oplus s_i] \leq (2i-1)/(2^n - i + 1)$ (by Propositions 1 and 2).
2. Case 2: $\sigma_i = -1$. $\mathsf{Succ}_i \Rightarrow x_i[R] = k$. Let $Q = \{x | x[R] = k\}$ then $|Q| = |2^{n-l}|$. Hence, $\Pr[\mathsf{Succ}_i] \leq \Pr[x_i \in Q] \leq 2^{n-l}/(2^n - i + 1)$ (by Proposition 1).

Therefore, $\Pr[\mathsf{Succ}_i] \leq \max\{(2i-1)/(2^n - i + 1), 2^{n-l}/(2^n - i + 1)\}$. Since $q \leq 2^{n-l-1}$, $\Pr[\mathsf{Succ}_i] \leq 2^{n-l}/(2^n - i + 1) \leq 2/(2^{l+1} - 1)$. Consider H_{23}^k in the case of the E3 hash family. For the other hash families in E3, the proof is analogous to the proof of 21. For E4/E5 hash functions, the proof is analogous to the proof of 23.

1. If $\sigma_i = +1$, then Succ_i implies that there exists an arc $(h, h') \in A(\tau_i)$ such that $h' \in O_{i-1}$. This implies that there exists an m such that $(y_i \oplus (m||k)) \in O_{i-1}$. By Proposition 2, $(y_i \oplus (m||k)) \in O_{i-1} \Leftrightarrow (y_i \oplus (0||k)) \in O_{i-1} \Leftrightarrow y_i \in O_{i-1} \oplus (0||k)$. Therefore, by Propositions 1 and 2, $\Pr[\mathsf{Succ}_i] \leq 2^{n-l}(2i-1)/(2^n - i + 1)$.
2. If $\sigma_i = -1$, then Succ_i implies that $x_i = s_i$. Hence, $\Pr[\mathsf{Succ}_i] \leq \Pr[x_i = s_i]$. Hence, by Proposition 1, $\Pr[\mathsf{Succ}_i] \leq \Pr[x_i = s_i] \leq 1/(2^n - i + 1)$.

Therefore, $\Pr[\mathsf{Succ}_i] \leq \max\{2^{n-l}(2i-1)/(2^n - i + 1), 1/(2^n - i + 1)\} = 2^{n-l}(2i-1)/(2^n - i + 1) \leq (2i-1)/2^{l-1}$. □

Thus, we have the following theorem using Observation 2. Note that we can first prove 1 and 3 of the following theorem with the restriction $q \leq 2^{n-1}$. However, in this case the upper bound is vacuous when $q > 2^{n-1}$. Thus, we do not need to restrict $q \leq 2^{n-1}$ in cases 1 and 3.

Theorem 2:
1. $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \leq q^2/2^{n-1}$ for $\iota \in$ E1
2. $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \leq 2q/(2^{l+1}-1)$ for all $q \leq 2^{n-l-1}$ and $\iota \in$ E2.
3. $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \leq q^2/2^{l-1}$ for $\iota \in$ E3, E4 or E5.

By the following theorem, the upper bound of advantage for the E1 hash family can also be obtained from that of the corresponding hash function presented in [1].

Theorem 3: $\forall \iota \in [1, 42], \mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \leq \mathbf{Adv}^{\mathrm{Coll}}_{H_\iota}(q)$

Proof. Suppose $\mathcal{A}$ is an adversary with respect to Coll for the hash family $\mathcal{H}_\iota$. We can easily construct the adversary $\mathcal{B}$ with respect to Coll for H_ι. Choose k at random from $\{0,1\}^l$. Run $\mathcal{A}$ to get M_1 and M_2 where, $M_1 = m_1^1||\cdots||m_a^1$, $M_1 = m_1^2||\cdots||m_b^2$, $|m_i^j| = n - l$ and $j = 1$ or 2. $\mathcal{B}$ outputs (M'_1, M'_2) where $M'_1 = (m_1^1||k)||\cdots||(m_a^1||k)$, and $M'_2 = (m_1^2||k)||\cdots||(m_b^2||k)$. It is very easy to check that if (M_1, M_2) is a collision pair for H_ι^k, then (M'_1, M'_2) is a collision pair for H_ι. Note that whenever $\mathcal{A}$ asks for an E-query/E^{-1}-query, $\mathcal{B}$ asks the same query and the output of the query is given to $\mathcal{A}$ in response to the query made by $\mathcal{B}$. □

In [1], the followings are known.

1. For $\iota \in [1, 12]$, $\mathbf{Adv}^{\mathrm{Coll}}_{H_\iota}(q) \le q(q+1)/2^n$.
2. For $\iota \in [13, 20]$, $\mathbf{Adv}^{\mathrm{Coll}}_{H_\iota}(q) \le 3q(q+1)/2^n$.

Thus, we can conclude from Theorems 2 and 3 the following.

Corollary 1: For $\iota \in [1, 12]$, $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \le \mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \le q(q+1)/2^n$. For $\iota = [13, 20]$, $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \le \mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \le q^2/2^{n-1}$.

4. Some Attacks in Target Collision Resistant Game

Idea of Attack : Here we will give a generic attack for all $\mathcal{H}_j$'s for the game TColl (See Section 2). Commit $M_1 = (m_1 \| \dots \| m_q)$. We will later describe how these m_i's will be chosen. Then given the random key k, compute $\mathcal{H}^k_j(M_1)$ using q queries. We will obtain $h_1, \dots h_q$ and $\mathcal{H}^k_j(M_1) = h_q$, where $h_0 \to_{m_1} h_1 \to_{m_2} \dots h_{q-1} \to_{m_q} h_q$. If we get one such $i < i'$ such that $h_i = h_{i'}$, then define $M_2 = m_1 \| \dots \| m_i \| m_{i'+1} \| \dots m_q$. Thus, M_1 and M_2 will be a collision pair. Roughly h_i's are random strings and the probability of success will be the probability for the birthday collision of h_i's which is $\mathrm{O}(q^2/2^n)$. We will choose m_i's so that the key for each query (i.e. s_i) is different. We assume that all h_i's are different, otherwise we get a collision.

Choice of m_i's :

1. If the key of the block cipher E is w in the definition of compression function, then choose $m_i = 0$. Thus, each w_i will be different as h_i's are different.
2. If the key is h or m, then choose $m_i = i$; hence, the keys are different.
3. If the key is v then choose m_i's so that the inputs of compression functions are different. In this case, we will study the lower bound separately.

Theorem 4: $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \ge \mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge \frac{0.3q(q-1)}{2^n}$ for each $\iota \in [1, 42]$ whenever the key of E is not v in the definition of the compression function.

Proof. Define D_i by the event that no collision occurs after the i^{th} query and D the event that the above attack fails after all queries, i.e., D is the same as D_q. Define D_0 by a sure event. Now, $Pr[D] = \prod_{i=1}^{q} Pr[\mathsf{D}_i|\mathsf{D}_{i-1}]$. If D_{i-1} is true, then all $h_{i'}$'s are different for $i' < i$. Now, $h_i = y_i \oplus \alpha_j$ (here, α_j depends on h_{i-1}, m_i and v). Now, D_i is true $\Leftrightarrow y_i \notin \{h_0, h_1, \dots, h_{i-1}\} \oplus \alpha_j$. Thus, $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] = (1 - \frac{i}{2^n})$. So $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge 1 - \prod_{i=1}^{q}(1 - \frac{i}{2^n}) \ge \frac{.3q(q-1)}{2^n}$ (the last inequality is in accordance with Proposition 5). □

For the hash family E3/E4/E5, we can have a better lower bound such as $\Omega(\frac{q^2}{2^l})$ if we just check whether $h_i[R] = h_{i'}[R]$ for $i < i'$ and construct M_2 depending on the type of the hash function. Choose m_i's as described earlier. The construction of M_2 is given below, where $h_i[R] = h_{i'}[R]$ for $i < i'$.

1. E3 : In the E3 family, if $h \to_m h'$ then $h \oplus (a\|0) \to_{m \oplus a} h' \oplus (a\|0)$. Thus, define $M_2 = m_1 \| \dots \| m_{i'} \| (m_{i+1} \oplus a) \| \dots \| (m_{i'} \oplus a) \| m_{i+1} \| \dots \| m_q$. Here, $a = h_i[R] \oplus h_{i'}[R]$. This will result in a collision because $\mathcal{H}_j(m_1 \| \dots \| m_{i'} \| (m_{i+1} \oplus a) \| \dots \| (m_{i'} \oplus a)) = h_i$.
2. E4 : By Proposition 2, we obtain some $m'_{i'}$ such that $h_{i'-1} \to_{m'_{i'}} h_{i'}$. Thus, define $M_2 = m_1 \| .. \| m_{i-1} \| m'_{i'} \| .. \| m_q$. This will result in a collision.
3. E5 : This case is very similar to E4, so we skip this.

Theorem 5: Let $\iota \in$ E3 or E4 or E5. If v is not the key of E in the definition for the compression function, then $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \ge \mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge \frac{0.3q(q-1)}{2^l}$. In other cases, $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \ge \mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge \frac{0.3q(q-1)}{2^{l-1}}$.

Proof. We use the same notations as above. If D_{i-1} is true, then all $h_{i'}[R]$'s are different for $i' < i$. Now, $h_i = y_i \oplus \alpha_j$ (here α_j depends on h_{i-1}, m_i and v). Now, D_i is true $\Leftrightarrow (y_i[R] \oplus \alpha_j[R] =) h_i[R] \notin \{h_0[R], h_1[R], \dots, h_{i-1}[R]\}$. Thus, $y_i \notin A - \{y_1, \dots, y_{i-1}\}$ where $A = \{x; x[R] \oplus a = h_{i'}[R], 0 \le i' \le i-1\}$ and $|A| = i.2^{n-l}$. Hence, $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] = (1 - \frac{i}{2^l})$. Thus, $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge 1 - \prod_{i=1}^{q}(1 - \frac{i}{2^l}) \le \frac{.3q(q-1)}{2^l}$ (the last inequality is in accordance with Proposition 5).

When the key is the same as v, then everything is the same as above except $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] = (1 - \frac{i2^{n-l}-i+1}{2^n-(i-1)})$ as y_i cannot take previous $i-1$ outputs. Thus, if $q \le 2^{n-1}$, $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] \ge (1 - \frac{i}{2^{l-1}})$; hence, $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge \frac{.3q(q-1)}{2^{l-1}}$. □

Attack for E2 Hash Family : We will consider the $\mathcal{H}_{21}$ hash family from E2. Other cases are similar to this family. Fix some $a > 0$ integer such that $(a+1)(a+2)/2 + a + 1 \ge q$. Let $m_1, \cdots, m_a$ be randomly chosen from $\{0,1\}^{n-l}$. Commit $M_1 = m_1 \| \dots \| m_a$. Then, given random key k, computes $\mathcal{H}_{21}(M_1)$ using a queries (we have to perform this by our convention). We will obtain $h_0, h_1 \dots, h_a = \mathcal{H}_{21}(M_1)$. If $h_i = h_{i'}$ for some $i < i'$ then $M_2 = m_1 \| \cdots \| m_i \| m_{i'+1} \| \cdots \| m_a$. The output is M_2. Otherwise run the loop below for $q - a$ many times.

For i, $j = 0$ to a ($j \ne i+1$, $i \le j$)
Compute $E^{-1}_{h_i}(h_i \oplus h_j) = x$
If $x[R] = k$ output $M_2 = m_1 \| \cdots \| m_i \| m_{j+1} \| \cdots \| m_a$.

Theorem 6: For each $\iota \in$ E2, $\mathbf{Adv}^{\mathrm{Coll}}_{\mathcal{H}_\iota}(q) \ge \mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\iota}(q) \ge .3a(a+1)/2^n + (q-a)/2^l$

Proof. Here, we have two possibilities to obtain a collision. In the first case, the success probability is at least $.3a(a+1)/2^n$ by an argument similar to that mentioned above. In the second case, $Pr[x[R] = k] \ge 1/2^l$ for each loop. Altogether, the success probability is at least $(q-a)/2^l$. One can write the proof in more detail. □

Proposition 5: $1 - \prod_{i=1}^{q}(1 - \frac{i}{2^a}) \ge \frac{.3q(q-1)}{2^a}$ for any integer

a.

Proof. The proof is given in [1], so we skip it. □

5. Inversion Resistance of Extended Hash Family

5.1 Upper Bound

In the Inv game, a random key k and a random h^* are given, where $h^* \in \{0,1\}^n$. Then the adversary $\mathcal{A}$ will try to compute M in the case of the extended hash function or h, m in the case of compression function such that $H_\iota^k(M) = h^*$ or $f_\iota^k(h,m) = h^*$. If he finds that, then we will say that adversary wins. As we have studied in the black-box model, the adversary can query E/E^{-1} similar to other games like Coll or TColl. Thus, the adversary has a transcript or sequence of query-response quadruples $\{(s_i, x_i, y_i, \sigma_i)\}_{1\le i\le q}$. In this section, we modify the definition of old vertices. In addition to the previous old vertices, we also include h^* as an old vertex in each T_i (See Section 3). By the new definition of the old vertex, the size of O_i is one more than that of the previous O_i. The definition of Succ_i is the same as the previous definition. Note that the definition of Succ_i involves old vertices. In that sense, this definition is changed slightly. Similarly to C_i, we define Inv_i which means that the adversary gets an inverse of h^* (i.e., the adversary wins) after the i^{th} query. It is very easy to check that $(\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1})$ implies Succ_i. Thus, for an extended hash family, we have one upper bound for the probability of winning in the Inv game which is the same as that in the Coll game (See Section 2 for the upper bound). However, we can have a better bound for the extended hash family using the theorem below.

Theorem 7: $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \le \mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{F}_\iota}(q)$ for each $\iota \in [1,42]$.

Proof. The proof for the single hash function and single compression function is given in [1]. The same proof will carry forward for the hash family and compression family. Intuitively, finding an inverse for an extended hash family is stronger than finding that for a compression function. □

Now, we will study the security analysis of the inversion resistance of compression functions. It can be easily observed that, for $\iota \in \{15,17,19,20,35,36,37\}$, compression functions are not inversion-resistant. All other compression functions are inversion-resistant.

Theorem 8: $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{F}_\iota}(q) \le q/2^{l-1}$ for $\iota \in [21,34]$ or $\iota \in \{13,14,16,18\}$.

Proof. Here we consider the hash family $\mathcal{H}_{23}$. Other cases will be very similar. A random key k and h^* are given to the adversary. The conditional event $(\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1})$ implies that the arc (h,h^*) corresponds to τ_i for some h (See Section 3). Thus, $E_{s_i}(x_i) = y_i \Leftrightarrow h \to_m h^*$ for some h and m. Thus, $h^* = y_i \oplus (m\|k)$ and $s_i = x_i$. If $\sigma_i = +1$, then $\Pr[\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1}] \le \Pr[y_i[R] = h^*[R] \oplus k] \le 2^{n-l}/(2^n - i + 1) \le 1/2^{l-1}$ (assume $q \le 2^{n-l}$, otherwise the bound is trivial). If $\sigma_i = -1$, $\Pr[\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1}] \le 1/(2^n - i + 1) \le 1/2^{n-1}$. Thus, $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{F}_\iota}(q) \le \sum_{i=1}^{q} \Pr[\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1}] \le q/2^{l-1}$. □

Theorem 9: $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{F}_\iota}(q) \le q/2^{n-1}$ for $\iota \in [38,42]$ or $[1,12]$.

Proof. Consider $\iota = 38$. Other cases will be similar. In fact, the idea of the proof is the same as the previous proof. $\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1}$ implies $y_i = h^* \oplus v$ and $x_i = s_i$. Thus, whenever $i \le 2^{n-1}$, $\Pr[\mathsf{Inv}_i|\neg\mathsf{Inv}_{i-1}] \le 1/2^{n-1}$ (check for $\sigma_i = +1$ and -1). □

For other cases $\iota \in \{35,36,37\}$, we can use the same technique used in proving the upper bound for the Coll game. By the discussion made in the beginning of this section, we can have the following theorem.

Theorem 10: $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \le q^2/2^{l-1}$ for $\iota \in [35,37]$ and $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \le \mathbf{Adv}^{\mathrm{Inv}}_{H_\iota}(q) \le 9(q+3)^2/2^n$ for $\iota \in \{15,17,19,20\}$.

Proof. The last part of the theorem is similar to Theorem 3 and from [1] we know $\mathbf{Adv}^{\mathrm{Inv}}_{H_\iota}(q) \le 9(q+3)^2/2^n$ for $\iota \in \{15,17,19,20\}$. □

5.2 Some Attacks in Inv Game for Lower Bound

Attack 1: When $\iota \in \{15,17,19,20,35,36,37\}$, i.e., when the corresponding compression is not inversion-resistant, we can perform meet-in-the-middle-attack. The idea of the attack is presented in [1]. Given h_0 and h^*, we compute two sets F and B such that $h_0 \to h_1$ for every $h_1 \in F$ and $h_2 \to h^*$ for every $h_2 \in B$. Note that we can construct B as the compression functions are not inversion-resistant. If we get an element in $F \cap B$, e.g., say h, then we have an inverse element of h^*. More precisely, if $h_0 \to_{m_1} h \to_{m_2} h^*$ for some m_1 and m_2 then $m_1\|m_2$ will be an inverse element of h^*. Thus, we have the following lower bound which is similar to the bound given in [1]; hence, we skip the proof.

Theorem 11: $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \ge (0.15)q^2/2^n$ for $\iota \in \{15,17,19,20\}$ and $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \ge (0.15)q^2/2^l$ for $\iota \in [35,37]$.

Attack 2: The attacking algorithm is the same as the generic attack for the target collision resistance described in Section 4. We choose $m_1, .., m_q$ and then compute $h_1, .., h_q$. Finally we look for some h_i such that $h_i = h^*$ (for $\iota \in [38,42]$ or $[1,12]$) or $h_i[R] = h^*[R]$ (for $\iota \in [21,34]$). One can prove this accurately but this will be the same as the proof of the collision attack; hence, we skip the details.

Theorem 12: $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \ge q/2^{l+1}$ for $\iota \in [21,34]$ and $\mathbf{Adv}^{\mathrm{Inv}}_{\mathcal{H}_\iota}(q) \ge q/2^n$ for $\iota \in [38,42]$ or $[1,12]$.

6. Conclusion

In this paper, we first generalized the definition of PGV-hash functions into PGV-hash families. In the new definition, we have more secure hash families (42 hash families) with respect to collision resistance and one-way-ness. Unlike previous definitions, it is a keyed family so that we can

study other security notions such as the target collision resistance. In fact, all these 42 hash families become target-collision-resistant. As AES is treated as a good candidate for a block cipher, we can implement these hash families using AES. From our results, only the attack for these hash families should explore some internal weakness of AES. That is, these hash families can be practically constructed using AES until we obtain some weakness of AES. The proof techniques used here are natural and direct for security notions. Thus, one can also study these proof techniques to obtain good ideas about using the black-box model.

Acknowledgements

This work was supported (in part) by the Ministry of Information & Communications, Korea, under the Information Technology Research Center (ITRC) Support Program, and also partly supported by the Grant-in-Aid for Creative Scientific Research No.14GS0218 of the Ministry of Education, Science, Sports and Culture (MEXT). The first author was supported by the 21st Century COE Program 'Reconstruction of Social Infrastructure Related to Information Science and Electrical Engineering' of Kyushu University.

References

[1] J. Black, P. Rogaway, and T. Shrimpton, "Black-box analysis of the block-cipher-based hash function constructions from PGV," Advances in Cryptology-Crypto'02, Lecture Notes in Computer Science, vol.2442, pp.320–335, Springer-Verlag, 2002.

[2] B. Preneel, R. Govaerts, and J. Vandewalle, "Hash functions based on block ciphers: A synthetic approach," Advances in Cryptology-CRYPTO'93, LNCS, pp.368–378, Springer-Verlag, 1994.

[3] S. Even and Y. Mansour, "A construction of a cipher from a single pseudorandom permutation," Advances in Cryptology-ASIACRYPT'91, LNCS 739, pp.210–224, Springer-Verlag, 1992.

[4] J. Kilian and P. Rogaway, "How to protect DES against exhaustive key search," J. Cryptology, vol.14, no.1, pp.17–35, 2001. Earlier version in CRYPTO' 96.

[5] R. Winternitz, "A secure one-way hash function built from DES," Proc. IEEE Symposium on Information Security and Privacy, pp.88–90, IEEE Press, 1984.

[6] C. Shannon, "Communication theory of secrecy systems," Bell Syst. Tech. J., vol.28, no.4, pp.656–715, 1949.

[7] M. Bellare and P. Rogaway, "Collision-resistant hashing: Towards making UOWHFs practical," Advances in Cryptology-Crypto'97, Lecture Notes in Computer Science, vol.1294, pp.470–484, Springer-Verlag, 1997.

[8] I.B. Damgard, "A design principle for hash functions," Advances in Cryptology—Crypto'89, Lecture Notes in Computer Sciences, vol.435, pp.416–427, Springer-Verlag, 1989.

[9] R. Merkle, "One way hash functions and DES," Advances in Cryptology—Crypto'89, Lecture Notes in Computer Sciences, vol.435, pp.428–446, Springer-Verlag, 1989.

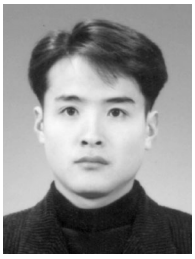
[10] I. Mironov, "Hash functions: From Merkle-Damgard to shoup," Advances in Cryptology—Eurocrypt'01, Lecture Notes in Computer Science, vol.2045, pp.166–181, Springer-Verlag, 2001.

[11] M. Naor and M. Yung, "Universal one-way hash functions and their cryptographic applications," Proc. Twenty First Annual ACM Symposium on Theory of Computing, pp.33–43, ACM Press, 1989.

[12] P. Sarkar, "Construction of UOWHF: Tree Hashing Revisited," Cryptology ePrint Archive, http://eprint.iacr.org/2002/058.

[13] V. Shoup, "A composition theorem for universal one-way hash functions," Advances in Cryptology—Eurocrypt'00, Lecture Notes in Computer Science, vol.1807, pp.445–452, Springer-Verlag, 2000.

[14] D. Simon, "Finding collisions on a one-way street: Can secure hash functions be based on general assumptions?," Advances in Cryptology—Eurocrypt'98, Lecture Notes in Computer Science, vol.1403, pp.334–345, Springer-Verlag, 1998.



Wonil Lee received the B.S., M.S., and D.S. degrees from Korea University, Seoul, Korea, in 1998, 2000, and 2003, respectively. He is a researcher of Center for Information Security Technologies (CIST) in Korea University and Faculty of Information Science and Electrical Engineering in Kyushu University. His current research interests include hash function and theory of cryptography.

Mridul Nandi received B.Stat., and M.Stat. from Indian Statistical Institute, Kolkata, India in 1999 and 2001, respectively. Now he is a Ph.D. student in Applied Statistics Unit, Indian Statistical Institute, Kolkata. He is also a member of Cryptology Research Group (CRG), India. His current research interests include hash function and theory of cryptography.

Palash Sarkar received his Bachelor of Electronics and Telecommunication Engineering degree in the year 1991 from Jadavpur University, Kolkata and Master of Technology in Computer Science in the year 1993 from Indian Statistical Institute, Kolkata. He completed his Ph.D. from Indian Statistical Institute in 1999. Currently he is an associate professor at Indian Statistical Instute. His research interests include cryptology, discrete mathematics and computer science.

Donghoon Chang received the B.S., and M.S. degrees from Korea University, Seoul, Korea, in 2001 and 2003, respectively. Now he is a Ph.D. student in Korea University. He is also a researcher of Center for Information Security Technologies (CIST) in Korea University. His current research interests include hash function and theory of cryptography.

Sangjin Lee received the B.S., M.S., and D.S. degrees from Korea University, Seoul, Korea, in 1987, 1989, and 1994, respectively. He had been engaged in the research and development on cryptography and information security at Electronics and Telecommunication Research Institute from 1989 to 1999. Currently he is a professor of Graduate School of Information Security in Korea University. His current research interests include cryptanalysis and theory of cryptography.

Kouichi Sakurai received the B.S. degree in Mathematics from Faculty of Science, and the M.S. degree in Applied Science from Faculty of Engineering, Kyushu University in 1986 and 1988, respectively. He had been engaged in the research and development on cryptography and information security at Computer & Information Systems Laboratory at Mitsubishi Electric Corporation from 1988 to 1994. He received the Dr. degree in engineering from Faculty of Engineering, Kyushu University in 1993. Currently he is a full professor of Department of Computer Science of Kyushu University. His current research interests include cryptography and information security.