A Generalization of PGV-Hash Functions and Security Analysis in Black-Box Model

Wonil Lee[1], Mridul Nandi[2], Palash Sarkar[2], Donghoon Chang[1], Sangjin Lee[1], and Kouichi Sakurai[3]

[1] Center for Information and Security Technologies
Korea University, Seoul, Korea
{wonil, dhchang, sangjin}@cist.korea.ac.kr
[2] Applied Statistics Unit, Indian Statistical Institute, Kolkata, India
{mridul_r,palash}@isical.ac.in
[3] Dept. of Computer Science and Communication Engineering,
Kyushu University, Fukuoka, Japan
sakurai@csce.kyushu-u.ac.jp

Abstract. In [1] it was proved that 20 out of 64 PGV-hash functions [2] based on block cipher are collision resistant and one-way-secure in 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 besides the previous 20 hash functions we have 22 more collision resistant and one-way secure hash families. As all these 42 families are keyed hash family, these become target collision resistant also. All these 42 hash families have tight upper and lower bounds on (target) collision resistant and one-way-ness.

1 Introduction

Brief History. Preneel, Govaerts, and Vandewalle [2] considered the 64 basic ways to construct 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 block cipher, 12 of 64 compression functions are CRHFs (Collision Resistant Hash Function) and 20 of 64 extended hash functions are CRHFs.

Motivation of Our paper. The examples of most popular collision resistant hash functions are MD5 and SHA-1. For those hash function one can not exactly analyze the security. But the security of collision resistant or one-way for PGV hash functions can be analyzed under the assumption that the underlying block cipher is black-box i.e. random permutation. But the security of other notions like target collision resistant can not be analyzed as it needs a family of hash functions instead of single hash function. Beside that it seemed that more PGV hash function would become secure if we change the original definition of PGV hash function. So, we generalize the definition of PGV hash function into a PGV hash family and will prove some security notions like target collision resistant, collision resistant and one-way.

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 a 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 extended hash family is equal to the key of compression function family.

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

Our Results. For $0 < l < n$, the security of the 64 schemes is summarized in Figures 1 and 2, which also serve to define the different extended hash functions H_i and their compression functions f_i. 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 are corresponding to the numbers in the first column of Figures 1 and 2 in Appendix. And E6 is the set of remaining extended hash families which are not represented in the first column of Figures 1 and 2 in Appendix. So $|$E6$| = 22$. This classification is based on some property of hash family which is used to prove the security. A high-level summary of our findings is given by Table 1 and 2 . The adversarial model (and the meaning of q) will be described momentarily.

Table 1. $l = 0$. This is analyzed in [1].

Extended Hash Families	(Target) Collision Bound	Inversion Bound
E1 (20 schemes)	$\Theta(q^2/2^n)$	$\Theta(q/2^n)$ or $\Theta(q^2/2^n)$
E2 (4 schemes)	$\Theta(1)$	–
E3/E4/E5 (18 schemes)	$\Theta(1)$	–
E6 (22 schemes)	$\Theta(1)$	–

Table 2. $0 < l < n$. This is analyzed in this paper.

Extended Hash Families	(Target) Collision Bound	Inversion Bound
E1 (20 schemes)	$\Theta(q^2/2^n)$	$\Theta(q/2^l)$ or $\Theta(q/2^n)$ or $\Theta(q^2/2^n)$
E2 (4 schemes)	$\Theta(q/2^l)$	$\Theta(q/2^l)$
E3/E4/E5 (18 schemes)	$\Theta(q^2/2^l)$	$\Theta(q/2^l)$ or $\Theta(q^2/2^l)$ or $\Theta(q/2^n)$
E6 (22 schemes)	$\Theta(1)$	–

Black Box Model. Our security model is the one dating to Shannon [6] and used for works like [3–5]. An 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 the point x such that $E_a(x) = y$. See [1] for more details and discussions about black-box model.

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

2 Preliminary

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

1. $[a, b] = \{a, \cdots, b\}$ where $a \leq b$ are some integers.
2. If $x \in \{0,1\}^n$ and $0 \leq l < n$, $x = x[L]||x[R]$, 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 a 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 of choosing a random element from the finite set S and calling it x.

Assumption. From now on, we 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. We fix some $h_0, v \in \{0,1\}^n$.

Collision Resistance and Inversion Resistance of Hash function ($l = 0$). To quantify the collision resistance of a (block-cipher-based) hash function H, we consider 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 $H(M) = H(M')$. And we also define the difficulty of 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 a 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*

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

Definition 2. *(Collision resistance and inversion resistance of an 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*

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

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 a (block-cipher-based) hash function family $\{H^k\}_{k \in \{0,1\}^l}$, we consider random block cipher E. An 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 a 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 an M.
2. A 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. A 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. $\mathcal{A}^{E,E^{-1}}$ has to find M such that $H^k(M) = h^*$.

Definition 3. *(Collision resistance, target collision resistance, and inversion resistance of a 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.*

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

Definition 4. *(Collision resistance, target collision resistance, and inversion resistance of an 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 the following real numbers.*

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

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

Conventions. We follow the similar conventions of [1]. Note that this convention is important to make the discussion easy and prove the following theorems. For the remainder 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 this 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 this 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 output(s) 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 will use the same assumption about the oracle query procedure of an adversary $\mathcal{A}$ for the compression function family $\mathcal{F}$.

These assumptions are all without loss of generality in that an adversary $\mathcal{A}$ not obeying these conventions can easily be modified to given an adversary $\mathcal{A}'$ having 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 $\mathcal{H}_\imath$ for each $\imath \in [1,42]$ defined in Section 1 in the notion of (target) collision resistant. We consider any adversary $\mathcal{A}$ with respect to Coll. i.e. after having random key k he will try to find a collision pair (M_1, M_2) for $H_\imath^k$ i.e. $M_1 \neq M_2$, $H_\imath^k(M_1) = H_\imath^k(M_2)$. For that he will make some E/E^{-1} queries. *Transcript* of $\mathcal{A}$ is defined by the sequence of query-response quadruples $\{(s_i, x_i, y_i, \sigma_i)\}_{1 \leq i \leq q}$ where q is the maximum number of queries made by 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 $\forall\ i$, $E_{s_i}(x_i) = y_i$. $(s_i, x_i, y_i, \sigma_i)$ will be called by i^{th} query-response quadruple (or q-r quadruple). In this section we fix some key 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 $A \subseteq \{0,1\}^n$, $Pr[y_i = y] \leq \frac{1}{2^n - i + 1}$ and $Pr[y_i \in A] \leq \frac{|A|}{2^n - i + 1}$ whenever $\sigma_i = +1$. Similarly, if $\sigma_i = -1$ then $Pr[x_i = x] \leq \frac{1}{2^n - i + 1}$ and $Pr[x_i \in A] \leq \frac{|A|}{2^n - i + 1}$*

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

Here we fix any arbitrary hash family $\mathcal{H}_\imath$ for $\imath \in [1, 42]$. In this section $V := \{0,1\}^n$ called *vertex set* and $L := \{0,1\}^{n-l}$ called *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 a *labeled arc* (or an *arc* only). We also say (h_1, h_2, m) is an arc (h_1, h_2) with label m or m is a label of the arc (h_1, h_2) and we use the notation $h_1 \to_m h_2$. Now given a triple $\tau = (s, x, y)$ where, $s, x, y \in V$ define a set of labeled arcs $A(\tau)$ by the following set :

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

For example, in case of $\mathcal{H}_{21}$, $f_{21}^k(h_1, m) := E_{h_1}(m||k) \oplus h_1$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_s(x) = y) \iff (E_{h_1}(m||k) \oplus h_1 = h_2 \Leftrightarrow E_s(x) = y) \iff (h_1 = s, h_2 = y \oplus h_1 = y \oplus s, m||k = x)$. 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 *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 an *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 $\tau[i]$ denotes 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 $(h_1 \to_{m_1} h_2 \to_{m_2} \cdots \to_{m_{p-1}} h_p)$ in T_i.

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

Definition 5. *For each hash function and* $0 \le i \le q$

1. *When* $\imath \in$ *E1, E2 or E4,* h *in* T_i *is* **old** *if deg*$(h) \ge 1$ *in* T_i *or* $h = h_0$.
2. *When* $\imath \in$ *E2 or E4,* h *in* T_i *is* **old** *if* $h = h_0$ *or* $\exists\ h_1$*, deg*$(h_1) \ge 1$ *in* T_i *and* $h[R] = h_1[R]$.

Remaining all other vertices are known as **new** *vertices. Call the set of all* old *vertices in* T_i *by* O_i.

The next Proposition will be used to have security analysis. It gives an upper bound of $|O_i|$ and says about 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,*

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

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

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

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

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

Proof. Bounds for $|O_i|$'s and last part of the proposition are straightforward from the structure of $A'(\tau_i)$. We will prove that 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 case of $\mathcal{H}_1$, $f_1^k(h_1, m) := E_{h_1}(m||k) \oplus (m||k)$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \iff (E_{h_1}(m||k) \oplus (m||k) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \iff (h_1 = s_i,$ $h_2 = y_i \oplus (m||k), x_i = m||k)$. 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 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 case of $\mathcal{H}_{23}$, $f_{23}^k(h_1, m) := E_{h_1}(h_1) \oplus (m||k)$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \iff (E_{h_1}(h_1) \oplus (m||k) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \iff (h_1 = s_i = x_i,\ h_2 = y_i \oplus (m||k))$. Hence, $A(\tau) = \{(s_i, h_2, m) : h_2[R] = y_i[R] \oplus k, m = h_2[R] \oplus y_i[R]\}$ if $x_i = s_i$ otherwise it is an empty set.

3. In case of $\mathcal{H}_{27}$, $f_{27}^k(h_1, m) := E_{w_1}(w_1) \oplus (m||k)$ where $w_1 = h_1 \oplus (m||k)$. So, $(f^k(h_1, m) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \iff (E_{w_1}(w_1) \oplus (m||k) = h_2 \Leftrightarrow E_{s_i}(x_i) = y_i) \iff (h_1 = s_i \oplus (m||k),\ h_2 = y_i \oplus (m||k) = h_1 \oplus (y_i \oplus s_i), s_i = x_i)$. 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 case of $\mathcal{H}_{36}$, we can prove similarly 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 $1 \leq i \leq q$ we define some events.*

1. *C_i : adversary gets a collision after i^{th} query.*
2. *$\mathsf{PathColl}_i$: $\exists$ 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 : $\exists$ 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 the Observation 1.

Proposition 4. *For E1, E2, E3, and E4 hash families, the 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 paths from h_0 to h^* in T_i' with different labels for some h^*. As $\mathsf{PathColl}_{i-1}$ is not true $\exists$ 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 $\exists$ two arcs either $(h_1, h_2), (h_2, h_3)$ or $(h_1, h_3), (h_2, h_3)$ corresponding to τ_i. But this is not possible by the structure of $A'(\tau_i)$ (see Proposition 2) in case of E1, E2, E3 and E4 hash families. Similarly we can prove it when $P_1 = P_2$.

In case of 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. So assume that P_1 and P_2 are different and $\exists$ $(h_1, h_3), (h_2, h_3)$ 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 $P_2' = (h_0 \rightarrow h_1' \rightarrow \ldots \rightarrow h_b')$. So if $\mathsf{Succ}_{i'}$ is not true $\forall\, i'$ $1 \leq i' \leq i$ 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 $a + b$ new vertices for T_0 in $P_1' \cup P_2'$ and every time at most one arc can be added into $A_j(\tau_{i'})$ (because of the structure of $A_j(\tau_{i'})$) we have to add exactly one new vertex in each i'. As $h_1[R] = h_2[R]$. So, 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, $\mathsf{C}_q \Rightarrow \bigcup_{i=1}^{q} \mathsf{Succ}_i$ by above Proposition 4. So we have $\Pr[\mathcal{A} \text{ gets a collision}] \leq \sum_{i=1}^{q} \Pr[\mathsf{Succ}_i]$. In other hash families by above Proposition 4, $\Pr[\mathcal{A} \text{ gets a collision}] \leq \sum_{i=1}^{q} \Pr[\mathsf{C}_i | \neg\mathsf{C}_{i-1}] \leq \sum_{i=1}^{q} \Pr[\mathsf{Succ}_i]$. So it is enough 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*

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

Proof. Let $\mathcal{A}$ be an adversary attacking $\mathcal{H}_i$. Assume that $\mathcal{A}$ asks its oracles at most q total 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 case of E1 hash family. For the other hash families in E1, the proof is analogous to the proof of 1.

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

2. Case 2: $\sigma_i = -1$. $\mathsf{Succ}_i \Rightarrow y_i \oplus x_i \in O_{i-1}$ (See Proposition 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 case of 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 Proposition 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 case of 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 will be analogous to the proof of 23.

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

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

So we have the following theorem using Observation 2.

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

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

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

Proof. Suppose $\mathcal{A}$ is an adversary with respect to Coll for the hash family $\mathcal{H}_\imath$. We can construct an adversary $\mathcal{B}$ with respect to Coll for $H_\imath$ very easily. 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 $H_\imath^k$ then (M_1', M_2') is a collision pair for $H_\imath$. Note, whenever $\mathcal{A}$ asks for E-query/E^{-1}-query, $\mathcal{B}$ asks same query and output of the query is given to $\mathcal{A}$ as a response of the query made by $\mathcal{B}$. ■

In [1] we know the followings :

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

So, we can conclude from Theorem 2 and 3 that,

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

4 Some Attacks in Target Collision Resistant Game

The idea of attack : Here we will give a generic attack for all $\mathcal{H}_j$ for the game TColl (See Section 2). Commit $M_1 = (m_1||\ldots||m_q)$. we will describe later how these m_i's will be chosen. Then given random key k compute $\mathcal{H}^k_j(M_1)$ by using q many queries. We will obtain $h_1, \ldots h_q$ and $\mathcal{H}^k_j(M_1) = h_q$ where, $h_0 \rightarrow_{m_1} h_1 \rightarrow_{m_2} \ldots h_{q-1} \rightarrow_{m_q} h_q$. If we get one such $i < i'$ such that $h_i = h_{i'}$ then define $M_2 = m_1||\ldots||m_i||m_{i'+1}||\ldots m_q$. So, M_1 and M_2 will be a collision pair. Roughly h_i's are random string and Probability of success will be probability for 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 key of block cipher E is w in the definition of compression function then choose $m_i = 0$. So each w_i will be different as h_i's are different.
2. If key is h or m then choose $m_i = i$ and hence keys are different.
3. If key is v then choose m_i's so that inputs of compression functions are different. In this case we will study the lower bound separately.

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

Proof. Define D_i by the event that no collision occurs after i^{th} query and D is the event that the above attack fails after all queries i.e. it is 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 $\Leftrightarrow y_i \notin \{h_0, h_1, \ldots, h_{i-1}\} \oplus \alpha_j$. So, $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] = (1 - \frac{i}{2^n})$. So $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\imath}(q) \ge 1 - \prod_{i=1}^q (1 - \frac{i}{2^n}) \ge \frac{.3q(q-1)}{2^n}$ (the last inequality is followed from Proposition 5). ∎

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

1. E3 : In E2 family if $h \rightarrow_m h'$ then $(h \oplus (a||0)) \rightarrow_{m \oplus a} (h' \oplus (a||0))$. So, define $M_2 = m_1|| \ldots ||m_{i'}||(m_{i+1} \oplus a)|| \ldots ||m_{i'} \oplus a||m_{i+1}|| \ldots ||m_q$. Here, $a = h_i[R] \oplus h_{i'}[R]$. This will give collision because $\mathcal{H}_j(m_1|| \ldots ||m_{i'}||(m_{i+1} \oplus a)|| \ldots ||(m_{i'} \oplus a) = h_i$.
2. E4 : By Proposition 2 we have some $m'_{i'}$ such that $h_{i'-1} \rightarrow_{m'_{i'}} h_{i'}$. So define $M_2 = m_1|| \ldots ||m_{i-1}||m'_{i'}|| \ldots ||m_q$. This will give a collision.
3. E5 : This case is very similar to E4 so we skip this.

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

Proof. We use 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 $\Leftrightarrow$ $(y_i[R] \oplus \alpha =)\ h_i[R] \notin \{h_0[R], h_1[R], \ldots, h_{i-1}[R]\}$. So, $y_i \notin A - \{y_1, \ldots, y_{i-1}\}$ where $A = \{x; x[R] \oplus a = h_{i'}[R], 0 \leq i' \leq i-1\}$ and $|A| = i.2^{n-l}$. Hence $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] = (1 - \frac{i}{2^l})$. So $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\imath}(q) \geq 1 - \prod_{i=1}^{q}(1 - \frac{i}{2^l}) \leq \frac{\cdot 3q(q-1)}{2^l}$ (the last inequality is followed from Proposition 5).

When key is same as v then everything is 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 can not take previous $i-1$ outputs. So if $q \leq 2^{n-1}$, $Pr[\mathsf{D}_i|\mathsf{D}_{i-1}] \geq (1 - \frac{i}{2^{l-1}})$ and hence $\mathbf{Adv}^{\mathrm{TColl}}_{\mathcal{H}_\imath}(q) \geq \frac{\cdot 3q(q-1)}{2^{l-1}}$ ■

Attack for E2 Hash Family : We will consider $\mathcal{H}_{21}$ hash family from E2. Other cases are similar to that. Fix some $a > 0$ integer such that $(a+1)(a+2)/2 + a + 1 \geq q$. Let $m_1, \ldots m_a \in_R \{0,1\}^{n-l}$. Commit $M_1 = m_1|| \ldots ||m_a$ where m_i's are chosen like above (to make keys are different and note that in E2 there is no hash function with key v). Then given random key k compute $\mathcal{H}_{21}(M_1)$ using a queries (we have to do it by our convention). We will obtain $h_0, h_1 \ldots, h_a = \mathcal{H}_{21}(M_1)$. If $h_i = h_{i'}$ for some $i < i'$ then $M_2 = m_1|| \ldots m_i||m_{j+1}||m_a$. Output M_2. Otherwise run the loop in below for $q - a$ many times.

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

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

Proof. Here we have two possibility to get collision. In first case success probability is at least $\cdot 3a(a+1)/2^n$ by similar argument as above. In the second case $Pr[x[R] = k] \geq 1/2^l$ for each loop. Altogether we have success probability is at least $(q-a)/2^l$. One can write down the proof in more details. ■

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

Proof. It is given in [1] so we skip the proof. ■

5 Inversion Resistance of Extended Hash Family

5.1 Upper Bound

In the Inv game a random key k and a random h^* will be given where, $h^* \in \{0,1\}^n$. Then he will try to compute M in case of extended hash function or h, m in case of compression function such that $H_\imath^k(M) = h^*$ or $f_\imath^k(h,m) = h^*$. If he finds that then we will say that adversary wins. As we study in black-box model adversary can query E/E^{-1} similar to other games like Coll or TColl. So, 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 old vertex, size of O_i is one more than that of previous O_i. Definition of Succ_i is same as previous definition. Note that the definition of Succ_i involves old vertices. In that sense this definition is changed a little. Like C_i we define Inv_i which means that adversary gets inverse of h^* (i.e. adversary wins) after i^{th} query. It is very easy to check that $(\mathsf{Inv}_i | \neg \mathsf{Inv}_i)$ implies Succ_i. So for extended hash family we have one upper bound for probability of winning in the Inv game which will be same as that in Coll game (See Section 2 for upper bound). But we can have better bound for extended hash family using the theorem below.

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

Proof. The proof for single hash function and single compression function is given in [1]. Same proof will carry forward for hash family and compression family also. Intuitively finding inverse for extended hash family is stronger than finding that for compression function. ■

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

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

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 event $(\mathsf{Inv}_i | \neg \mathsf{Inv}_{i-1})$ implies the arc (h, h^*) corresponds to τ_i for some h (See Section 3). So, $E_{s_i}(x_i) = y_i \Leftrightarrow h \to_m h^*$ for some h and m. So $h^* = y_i \oplus (m||k)$ and $s_i = x_i$.

1. 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).
2. If $\sigma_i = -1$ then $\Pr[\mathsf{Inv}_i | \neg \mathsf{Inv}_{i-1}] \le 1/(2^n - i + 1) \le 1/2^{n-1}$.

So, $\mathbf{Adv}_{\mathcal{F}_\imath}^{\mathrm{Inv}}(q) \le \sum_{i=1}^{q} \Pr[\mathsf{Inv}_i | \neg \mathsf{Inv}_{i-1}] \le q/2^{l-1}$. ■

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

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

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

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

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

5.2 Some attacks in Inv game for Lower Bound

Attack 1 : When $\imath \in \{15, 17, 19, 20, 35, 36, 37\}$ i.e. when the corresponding compression are not inversion resistance-secure we can perform meet-in-the-middle-attack. Idea of the attack is presented in [1]. Given h_0 and h^* we compute two sets F and B such that $h \to h_1$ for every $h_1 \in F$ and $h_2 \to h^*$ for every $h_2 \in B$. Note we can construct B as the compression functions are not inversion resistance-secure. If we get an element in $F \cap B$ 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^*. So we have the following lower bound which is similar to the bound given in [1] and hence we skip the proof.

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

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

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

6 Conclusion

In this paper we first generalized the definition of PGV-hash functions into a PGV-hash families. In the new definitions we have more secure hash family (42 hash families) with respect to collision resistant and One-way. Unlike previous definitions it is a keyed family so we can study other security notion like target collision resistant. In fact all these 42 hash families become target collision resistant. As AES is treated as a good candidate for block cipher, we can implement these hash families using AES. Because of our results, only attack for these hash families should explore some internal weakness of AES. In other words, these hash families can be practically constructed using AES until we are getting some weakness of AES. The proof techniques used here are natural and direct to the security notions. So one can also study these proof techniques to have good ideas about using the black box model.

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, Springer-Verlag, pp. 320-335, 2002.
2. B. Preneel, R. Govaerts, and J. Vandewalle. *Hash functions based on block ciphers:A synthetic approach*, Advances in Cryptology-CRYPTO'93, LNCS, Springer-Verlag, pp. 368-378, 1994.
3. S. Even, and Y. Mansour. *A construction of a cipher from a single pseudorandom permutation*, Advances in Cryptology-ASIACRYPT'91, LNCS 739, Springer-Verlag, pp. 210-224, 1992.
4. J. Kilian, and P. Rogaway. *How to protect DES against exhaustive key search*, Journal of Cryptology, 14(1):17-35, 2001, Earlier version in CRYPTO' 96.
5. R. Winternitz. *A secure one-way hash function built from DES*, In Proceedings of the IEEE Symposium on Information Security and Privacy, pp. 88-90, IEEE Press, 1984
6. C. Shannon. *Communication theory of secrecy systems*, Bell Systems Technical Journal, 28(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, Springer-Verlag, pp. 470-484, 1997.
8. I. B. Damgard. *A design principle for hash functions*, Advances in Cryptology - Crypto'89, Lecture Notes in Computer Sciences, Vol. 435, Springer-Verlag, pp. 416-427, 1989.
9. R. Merkle. *One way hash functions and DES*, Advances in Cryptology - Crypto'89, Lecture Notes in Computer Sciences, Vol. 435, Springer-Verlag, pp. 428-446, 1989.
10. I. Mironov. *Hash functions: from Merkle-Damgard to Shoup*, Advances in Cryptology - Eurocrypt'01, Lecture Notes in Computer Science, Vol. 2045, Springer-Verlag, pp 166-181, 2001
11. M. Naor and M. Yung. *Universal one-way hash functions and their cryptographic applications*, Proceedings of the Twenty First Annual ACM Symposium on Theory of Computing, ACM Press, pp 33-43, 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, Springer-Verlag, pp 445-452, 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, Springer-Verlag, pp 334-345, 1998.

Appendix

$\imath$	$\jmath$	$h_i =$	**(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	$E_v(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^{l-1}$	$q^2/2^{l-1}$	$q/2^l$	$q/2^{l-1}$

Fig. 1. Summary of results about 64 extended hash families. Column 1 is our number $\imath$ for the function family (We write $\mathcal{F}_\imath$ for the compression function family and $\mathcal{H}_\imath$ for its induced extended hash family). Column 2 is 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 give our (target) collision resistance bounds. Columns 6 and 7 give our inversion resistance bounds.

$\imath$	$\jmath$	$h_i =$	**(T)CR LB**	**(T)CR UB**	**IR LB**	**IR UB**
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	–	–

Fig. 2. Summary of results about 64 extended hash families, continued.