Black-Box Analysis of the Block-Cipher-Based Hash-Function Constructions from PGV

John Black[1], Phillip Rogaway[2], and Thomas Shrimpton[3]

[1] Dept. of Computer Science, University of Colorado, Boulder CO 80309, USA, `jrblack@cs.colorado.edu`, `www.cs.colorado.edu/~jrblack`

[2] Dept. of Computer Science, University of California, Davis, CA 95616, USA, and Dept. of Computer Science, Fac of Science, Chiang Mai University, 50200 Thailand, `rogaway@cs.ucdavis.edu`, `www.cs.ucdavis.edu/~rogaway`

[3] Dept. of Electrical and Computer Engineering, University of California, Davis, CA 95616, USA, `teshrim@ucdavis.edu`, `www.ece.ucdavis.edu/~teshrim`

Abstract. Preneel, Govaerts, and Vandewalle [**?**] considered the 64 most basic ways to construct a hash function $H\colon \{0,1\}^* \to \{0,1\}^n$ from a block cipher $E\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$. They regarded 12 of these 64 schemes as secure, though no proofs or formal claims were given. The remaining 52 schemes were shown to be subject to various attacks. Here we provide a formal and quantitative treatment of the 64 constructions considered by PGV. We prove that, in a black-box model, the 12 schemes that PGV singled out as secure really *are* secure: we give tight upper and lower bounds on their collision resistance. Furthermore, by stepping outside of the Merkle-Damgård approach to analysis, we show that an additional 8 of the 64 schemes are just as collision resistant (up to a small constant) as the first group of schemes. Nonetheless, we are able to differentiate among the 20 collision-resistant schemes by bounding their security as one-way functions. We suggest that proving black-box bounds, of the style given here, is a feasible and useful step for understanding the security of any block-cipher-based hash-function construction.

1 Introduction

Background. The most popular collision-resistant hash-functions (eg., MD5 and SHA-1) iterate a compression function that is constructed from scratch (i.e., one that doesn't use any lower-level cryptographic primitive). But there is another well-known approach, going back to Rabin [**?**], wherein one makes the compression function out of a block cipher. This approach has been less widely used, for a variety of reasons. These include export restrictions on block ciphers, a preponderance of 64-bit block lengths, problems attributable to "weak keys", and the lack of popular block ciphers with per-byte speeds comparable to that of MD5 or SHA-1. Still, the emergence of the AES has somewhat modified this landscape, and now motivates renewed interest in finding good ways to turn a block cipher into a cryptographic hash function. This paper casts some fresh light on the topic.

The PGV paper. We return to some old work by Preneel, Govaerts, and Vandewalle [?] that considered turning a block cipher E: $\{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ into a hash function H: $(\{0,1\}^n)^* \to \{0,1\}^n$ using a compression function f: $\{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ derived from E. For v a fixed n-bit constant, PGV considers all 64 compression functions f of the form $f(h_{i-1}, m_i) = E_a(b) \oplus c$ where $a, b, c \in \{h_{i-1},\ m_i,\ h_{i-1} \oplus m_i,\ v\}$. Then define the *iterated hash of f* as:

function $H\,(m_1 \cdots m_\ell)$
for $i \leftarrow 1$ **to** ℓ **do** $h_i \leftarrow f\,(h_{i-1}, m_i)$
return h_ℓ

Here h_0 is a fixed constant, say 0^n, and $|m_i| = n$ for each $i \in [1..\ell]$. Of the 64 such schemes, the authors of [?] regard 12 as secure. Another 13 schemes they classify as *backward-attackable*, which means they are subject to an identified (but not very severe) potential attack. The remaining 39 schemes are subject to damaging attacks identified by [?] and others.

Some missing results. The authors of [?] focused on attacks, not proofs. All the same, it seems to be a commonly held belief that it should be possible to produce proofs for the schemes they regarded as secure. Indeed [?] goes so far as to say that "For each of these schemes it is possible to write a 'security proof' based on a black box model of the encryption algorithm, as was done for the Davies-Meyer scheme [by Winternitz [?]]". This latter paper uses a black-box model of a block cipher—a model dating back to Shannon [?]—to show that the scheme we will later call H_5 is secure in the sense of preimage-resistance. Specifically, [?] shows that any algorithm (with E and E^{-1} oracles) that always finds a preimage under H_5 for a fixed value $y \in \{0,1\}^n$ will necessarily make at least 2^{n-1} expected oracle queries.

The model introduced by Winternitz for analyzing block-cipher-based hash functions was subsequently used by Merkle [?]. He gives black-box model arguments for H_1, and other functions, and considers questions of efficiency and concrete security. The black-box model of a block cipher has also found use in other contexts, such as [?, ?]. But, prior to the current work, we are unaware of any careful analysis in the literature, under any formalized model, for the collision-resistance of any block-cipher-based hash-function.

Summary of our results. This paper takes a more proof-centric look at the schemes from PGV [?], providing both upper and lower bounds for each. Some of our results are as expected, while others are not.

First we prove collision-resistance for the 12 schemes singled out by PGV as secure (meaning those marked "✓" or "FP" in [?]). We analyze these *group-1* schemes, $\{H_1, \ldots, H_{12}\}$, within the Merkle-Damgård paradigm. That is, we show that for each group-1 scheme $H_\imath$ its compression function $f_\imath$ is already collision resistant, and so $H_\imath$ must be collision resistant as well.

PGV's backward-attackable schemes (marked "B" in [?]) held more surprises. We find that eight of these 13 schemes are secure, in the sense of collision resistance. In fact, these eight *group-2* schemes, $\{H_{13}, \ldots, H_{20}\}$, are just as collision-resistant as the group-1 schemes.

Despite having essentially the same collision-resistance, the group-1 and group-2 schemes can be distinguished based on their security as one-way functions: we get a better bound on inversion-resistance for the group-1 schemes than we get for the group-2 schemes. Matching attacks (up to a constant) demonstrate that this difference is genuine and not an artifact of the security proof.

The remaining $44 = 64 - 20$ hash functions considered by PGV are completely insecure: for these *group-3* schemes one can find a (guaranteed) collision with two or fewer queries. This includes five of PGV's backward-attackable schemes, where [**?**] had suggested a (less effective) meet-in-the-middle attack (see Appendix A).

Other surprises emerged in the mechanics of carrying out our analyses. Unlike the group-1 schemes, we found that the group-2 schemes could not be analyzed within the Merkle-Damgård paradigm; in particular, these schemes are collision resistant even though their compression functions are not. We also found that, for one set of schemes, the "obvious attack" on collision resistance needed some subtle probabilistic reasoning to rigorously analyze.

The security of the 64 PGV schemes is summarized in Fig. 1 and Fig. 2, which also serve to define the different hash functions $H_\imath$ and their compression functions $f_\imath$. Fig. 3 gives a more readable description of $f_1, \ldots, f_{20}$. A high-level summary of our findings is given by the following chart. The model (and the meaning of q) will be described momentarily.

<table>
<tr><th>PGV Category</th><th>Our Category</th><th>Collision Bound</th><th>OWF Bound</th></tr>
<tr><td>✓or FP (12 schemes)</td><td>group-1: $H_{1..12}$ (12 schemes)</td><td>$\Theta(q^2/2^n)$</td><td>$\Theta(q/2^n)$</td></tr>
<tr><td rowspan="2">B (13 schemes)</td><td>group-2: $H_{13..20}$ (8 schemes)</td><td>$\Theta(q^2/2^n)$</td><td>$\Theta(q^2/2^n)$</td></tr>
<tr><td rowspan="2">group-3 (44 schemes)</td><td rowspan="2">$\Theta(1)$</td><td rowspan="2">$\Theta(1)$</td></tr>
<tr><td>F, P, or D (39 schemes)</td></tr>
</table>

Black-box model. Our model is the one dating to Shannon [**?**] and used for works like [**?**, **?**, **?**]. Fix a key-length κ and a block length n. An adversary A is given access to oracles E and E^{-1} where E is a random block cipher E: $\{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ and E^{-1} is its inverse. That is, each key $k \in \{0,1\}^\kappa$ names a randomly-selected permutation $E_k = E(k, \cdot)$ on $\{0,1\}^n$, and the adversary is given oracles E and E^{-1}. The latter, on input (k, y), returns the point x such that $E_k(x) = y$.

For a hash function H that depends on E, the adversary's job in attacking the collision resistance of H is to find distinct M, M' such that $H(M) = H(M')$. One measures the optimal adversary's chance of doing this as a function of the number of E or E^{-1} queries it makes. Similarly, the adversary's job in inverting H is to find an inverse under H for a random range point $Y \in \{0,1\}^n$. (See Section 2 for a justification of this definition.) One measures the optimal adversary's chance of doing this as a function of the total number of E or E^{-1} queries it makes.

$\imath$	$\jmath$	$h_i =$	**CR low-bnd**	**CR up-bnd**	**IR low-bnd**	**IR up-bnd**	
	1	$E_{m_i}(m_i) \oplus v$	1	1			a
	2	$E_{h_{i-1}}(m_i) \oplus v$	1	1			b
13	3	$E_{w_i}(m_i) \oplus v$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	4	$E_v(m_i) \oplus v$	1	1			a
	5	$E_{m_i}(m_i) \oplus m_i$	1	1			a
1	6	$E_{h_{i-1}}(m_i) \oplus m_i$	$.039(q-1)(q-3)/2^n$	$q(q+1)/2^n$	$0.4q/2^n$	$2q/2^n$	d
9	7	$E_{w_i}(m_i) \oplus mi$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	8	$E_v(m_i) \oplus m_i$	1	1			a
	9	$E_{m_i}(m_i) \oplus h_{i-1}$	1	1			f
	10	$E_{h_{i-1}}(m_i) \oplus h_{i-1}$	1	1			b
11	11	$E_{w_i}(m_i) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	12	$E_v(m_i) \oplus h_{i-1}$	1	1			b
	13	$E_{m_i}(m_i) \oplus w_i$	1	1			f
3	14	$E_{h_{i-1}}(m_i) \oplus w_i$	$.039(q-1)(q-3)/2^n$	$q(q+1)/2^n$	$0.4q/2^n$	$2q/2^n$	d
14	15	$E_{w_i}(m_i) \oplus w_i$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	16	$E_v(m_i) \oplus w_i$	1	1			f
15	17	$E_{m_i}(h_{i-1}) \oplus v$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	18	$E_{h_{i-1}}(h_{i-1}) \oplus v$	1	1			a
16	19	$E_{w_i}(h_{i-1}) \oplus v$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	20	$E_v(h_{i-1}) \oplus v$	1	1			a
17	21	$E_{m_i}(h_{i-1}) \oplus m_i$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	22	$E_{h_{i-1}}(h_{i-1}) \oplus m_i$	1	1			b
12	23	$E_{w_i}(h_{i-1}) \oplus m_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	24	$E_v(h_{i-1}) \oplus m_i$	1	1			b
5	25	$E_{m_i}(h_{i-1}) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	26	$E_{h_{i-1}}(h_{i-1}) \oplus h_{i-1}$	1	1			a
10	27	$E_{w_i}(h_{i-1}) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	28	$E_v(h_{i-1}) \oplus h_{i-1}$	1	1			a
7	29	$E_{m_i}(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	30	$E_{h_{i-1}}(h_{i-1}) \oplus w_i$	1	1			b
18	31	$E_{w_i}(h_{i-1}) \oplus w_i$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	32	$E_v(h_{i-1}) \oplus w_i$	1	1			b

Fig. 1. Summary of results. Column 1 is our number $\imath$ for the function (we write $f_\imath$ for the compression function and $H_\imath$ for its induced hash function). Column 2 is the number from [**?**] (we write $\hat{f}_\jmath$ and $\hat{H}_\jmath$). Column 3 defines $f_\imath(h_{i-1}, m_i)$ and $\hat{f}_\jmath(h_{i-1}, m_i)$. We write w_i for $m_i \oplus h_{i-1}$. Columns 4–7 give our collision-resistance and inversion-resistance bounds. Column 8 comments on collision-finding attacks: (a) $H(M)$ is determined by the last block only; two E queries; (b) Attack uses two E queries and one E^{-1} query; (c) Attack uses $q/2$ E queries and $q/2$ E^{-1} queries; (d) Attack given by Theorem 3; (e) Attack given by Theorem 4; (f) $H(M)$ independent of block order; two E queries; (g) Attack uses (at most) two E queries. We do not explore inversion resistance for schemes that are trivially breakable in the sense of collision resistance.

$\imath$	$\jmath$	$h_i =$	**CR low-bnd**	**CR up-bnd**	**IR low-bnd**	**IR up-bnd**	
19	33	$E_{m_i}(w_i) \oplus v$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
	34	$E_{h_{i-1}}(w_i) \oplus v$	1	1			b
	35	$E_{w_i}(w_i) \oplus v$	1	1			g
	36	$E_v(w_i) \oplus v$	1	1			b
20	37	$E_{m_i}(w_i) \oplus m_i$	$.3q(q-1)/2^n$	$3q(q+1)/2^n$	$0.15q^2/2^n$	$9(q+3)^2/2^n$	c
4	38	$E_{h_{i-1}}(w_i) \oplus m_i$	$.039(q-1)(q-3)/2^n$	$q(q+1)/2^n$	$0.4q/2^n$	$2q/2^n$	d
	39	$E_{w_i}(w_i) \oplus m_i$	1	1			g
	40	$E_v(w_i) \oplus m_i$	1	1			g
8	41	$E_{m_i}(w_i) \oplus h_{i-1}$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
	42	$E_{h_{i-1}}(w_i) \oplus h_{i-1}$	1	1			b
	43	$E_{w_i}(w_i) \oplus h_{i-1}$	1	1			g
	44	$E_v(w_i) \oplus h_{i-1}$	1	1			b
6	45	$E_{m_i}(w_i) \oplus w_i$	$.3q(q-1)/2^n$	$q(q+1)/2^n$	$0.6q/2^n$	$2q/2^n$	e
2	46	$E_{h_{i-1}}(w_i) \oplus w_i$	$.039(q-1)(q-3)/2^n$	$q(q+1)/2^n$	$0.4q/2^n$	$2q/2^n$	d
	47	$E_{w_i}(w_i) \oplus w_i$	1	1			g
	48	$E_v(w_i) \oplus w_i$	1	1			g
	49	$E_{m_i}(v) \oplus v$	1	1			a
	50	$E_{h_{i-1}}(v) \oplus v$	1	1			a
	51	$E_{w_i}(v) \oplus v$	1	1			g
	52	$E_v(v) \oplus v$	1	1			a
	53	$E_{m_i}(v) \oplus m_i$	1	1			a
	54	$E_{h_{i-1}}(v) \oplus m_i$	1	1			b
	55	$E_{w_i}(v) \oplus m_i$	1	1			g
	56	$E_v(v) \oplus m_i$	1	1			a
	57	$E_{m_i}(v) \oplus h_{i-1}$	1	1			f
	58	$E_{h_{i-1}}(v) \oplus h_{i-1}$	1	1			a
	59	$E_{w_i}(v) \oplus h_{i-1}$	1	1			g
	60	$E_v(v) \oplus h_{i-1}$	1	1			a
	61	$E_{m_i}(v) \oplus w_i$	1	1			f
	62	$E_{h_{i-1}}(v) \oplus w_i$	1	1			b
	63	$E_{w_i}(v) \oplus w_i$	1	1			g
	64	$E_v(v) \oplus w_i$	1	1			b

Fig. 2. Summary of results, continued. See the caption of Fig. 1 for an explanation of the entries in this table.

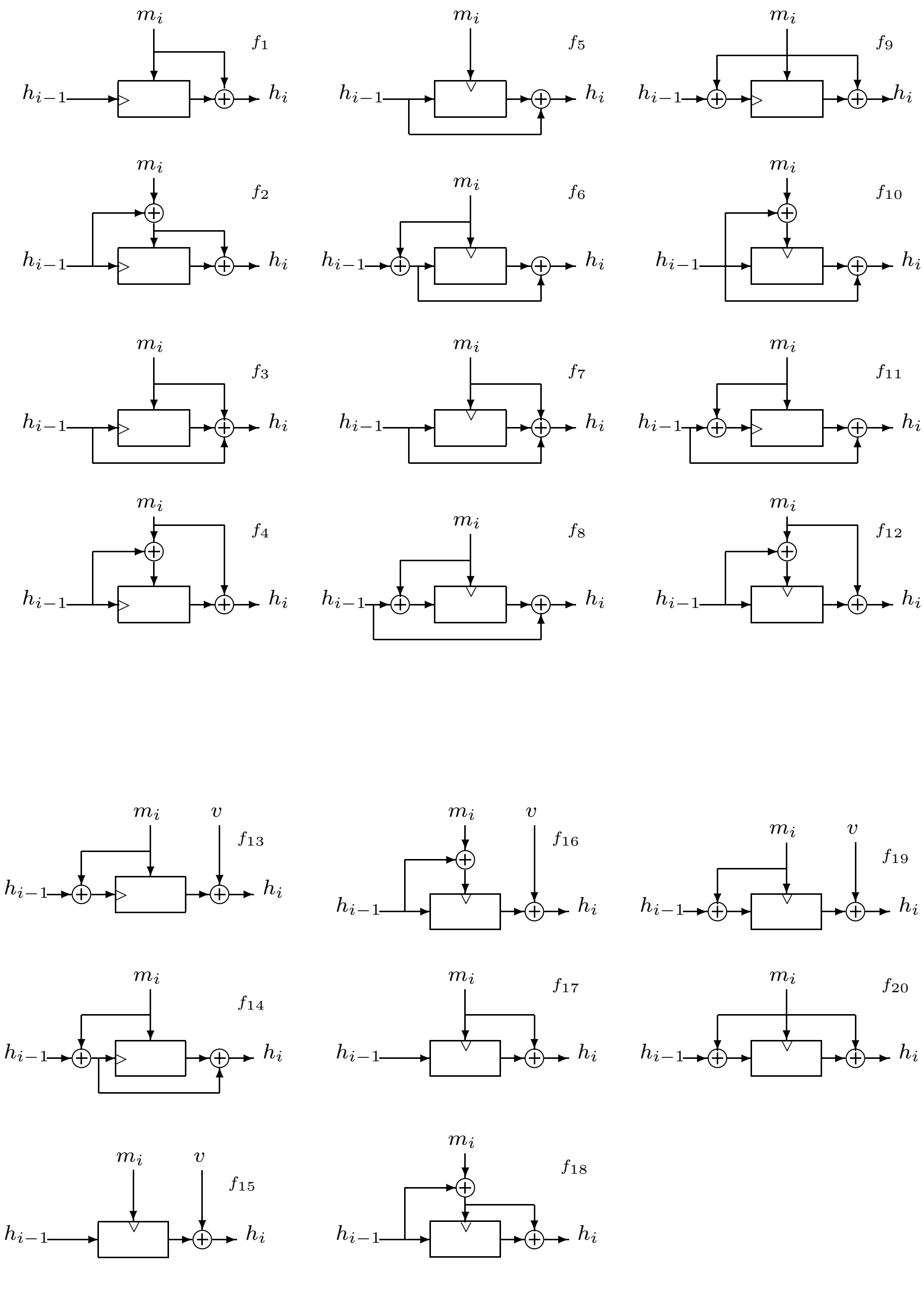

Fig. 3. The compression functions $f_1, \ldots, f_{20}$ for the 20 collision-resistant hash functions $H_1, \ldots, H_{20}$. A hatch marks the location for the key.

Discussion. As with [?], we do not concern ourselves with MD-strengthening [?, ?], wherein strings are appropriately padded so that any $M \in \{0,1\}^*$ may be hashed. Simple results establish the security of the MD-strengthened hash function H^* one gets from a secure multiple-of-block-length hash-function H. All of our attacks work just as well in the presence of MD-strengthening.

It is important not to read too much or too little into black-box results. On the one hand, attacks on block-cipher-based hash-functions have usually treated the block cipher as a black box. Such attacks are doomed when one has strong results within the black-box model. On the other hand, the only structural aspect of a block cipher captured by the model is its invertibility, so one must be skeptical about what a black-box-model result suggests when using a block cipher with significant structural properties, such as weak keys. With a block cipher like AES, one hopes for better. Overall, we see the black-box model as an appropriate first step in understanding the security of block-cipher-based hash-functions. Of course it would be nice to make due with standard assumptions, such as the block cipher being a pseudorandom function, but that assumption is insufficient for our purposes, and no sufficient assumption has been proposed.

Future directions. Though we spoke of AES as rekindling interest in block-cipher-based hash-function designs, we do not address what we regard as the most interesting practical problem in that vein: namely, how best to use an n-bit block cipher to make a hash function with output length larger than n bits. (Many people see $n = 128$ bits as an inadequate output length for a hash function, particularly in view of [?].) The current work does not answer this question, but it does lay the groundwork for getting there.

2 Definitions

Basic notions. Let $\kappa, n \geq 1$ be numbers. A *block cipher* is a map $E\colon \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ where, for each $k \in \{0,1\}^\kappa$, the function $E_k(\cdot) = E(k, \cdot)$ is a permutation on $\{0,1\}^n$. If E is a block cipher then E^{-1} is its inverse, where $E_k^{-1}(y)$ is the string x such that $E_k(x) = y$. Let $\text{Bloc}(\kappa, n)$ be the set of all block ciphers $E\colon \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$.

A (block-cipher-based) *hash function* is a map $H\colon \text{Bloc}(\kappa, n) \times D \to R$ where $\kappa, n, c \geq 1$, $D \subseteq \{0,1\}^*$, and $R = \{0,1\}^c$. The function H must be given by a program that, given M, computes $H^E(M) = H(E, M)$ using an E-oracle. Hash function $f\colon \text{Bloc}(\kappa, n) \times D \to R$ is a *compression function* if $D = \{0,1\}^a \times \{0,1\}^b$ for some $a, b \geq 1$ where $a + b \geq c$. Fix $h_0 \in \{0,1\}^a$. The *iterated hash* of compression function $f\colon \text{Bloc}(\kappa, n) \times (\{0,1\}^a \times \{0,1\}^b) \to \{0,1\}^a$ is the hash function $H\colon \text{Bloc}(\kappa, n) \times (\{0,1\}^b)^* \to \{0,1\}^a$ defined by $H^E(m_1 \cdots m_\ell) = h_\ell$ where $h_i = f^E(h_{i-1}, m_i)$. Set $H^E(\varepsilon) = h_0$. If the program for f uses a single query $E(k, x)$ to compute $f^E(m, h)$ then f (and its iterated hash H) is *rate-1*. We often omit the superscript E to f and H.

We write $x \xleftarrow{\$} S$ for the experiment of choosing a random element from the finite set S and calling it x. An *adversary* is an algorithm with access to one or more oracles. We write these as superscripts.

Collision resistance. To quantify the collision resistance of a block-cipher-based hash function H we instantiate the block cipher by a randomly chosen $E \in \mathrm{Bloc}(\kappa, n)$. An adversary A is given oracles for $E(\cdot,\cdot)$ and $E^{-1}(\cdot,\cdot)$ and wants to find a *collision* for H^E—that is, M, M' where $M \neq M'$ but $H^E(M) = H^E(M')$. We look at the number of queries that the adversary makes and compare this with the probability of finding a collision.

Definition 1 (Collision resistance of a hash function). Let H be a block-cipher-based hash function, $H\colon \mathrm{Bloc}(\kappa, n) \times D \to R$, and let A be an adversary. Then the advantage of A in finding collisions in H is the real number

$$\mathbf{Adv}_H^{\mathrm{coll}}(A) = \Pr\Big[E \xleftarrow{\$} \mathrm{Bloc}(\kappa, n);\ (M, M') \xleftarrow{\$} A^{E,E^{-1}} : \\ M \neq M' \ \wedge\ H^E(M) = H^E(M')\Big] \qquad \diamondsuit$$

For $q \geq 1$ we write $\mathbf{Adv}_H^{\mathrm{coll}}(q) = \max_A\{\mathbf{Adv}_H^{\mathrm{coll}}(A)\}$ where the maximum is taken over all adversaries that ask at most q oracle queries (ie, E-queries + E^{-1} queries). Other advantage functions are silently extended in the same way.

We also define the advantage of an adversary in finding collisions in a compression function $f\colon \mathrm{Bloc}(\kappa, n) \times \{0,1\}^a \times \{0,1\}^b \to \{0,1\}^c$. Naturally (h, m) and (h', m') collide under f if they are distinct and $f^E(h, m) = f^E(h', m')$, but we also give credit for finding an (h, m) such that $f^E(h, m) = h_0$, for a fixed $h_0 \in \{0,1\}^c$. If one treats the hash of the empty string as the constant h_0 then $f^E(h, m) = h_0$ amounts to having found a collision between (h, m) and the empty string.

Definition 2 (Collision resistance of a compression function). Let f be a block-cipher-based compression function, $f\colon \mathrm{Bloc}(\kappa, n) \times \{0,1\}^a \times \{0,1\}^b \to \{0,1\}^c$. Fix a constant $h_0 \in \{0,1\}^c$ and an adversary A. Then the advantage of A in finding collisions in f is the real number

$$\mathbf{Adv}_f^{\mathrm{comp}}(A) = \Pr\Big[E \xleftarrow{\$} \mathrm{Bloc}(\kappa, n);\ ((h, m), (h', m')) \xleftarrow{\$} A^{E,E^{-1}} : \\ \big((h, m) \neq (h', m') \ \wedge\ f^E(h, m) = f^E(h', m')\big) \ \vee\ f^E(h, m) = h_0\Big] \qquad \diamondsuit$$

Inversion resistance. Though we focus on collision resistance, we are also interested in the difficulty of inverting hash functions. We use the following measure for the difficulty of inverting a hash function at a random point.

Definition 3 (Inverting random points). Let H be a block-cipher-based hash function, $H\colon \mathrm{Bloc}(\kappa, n) \times D \to R$, and let A be an adversary. Then the advantage of A in inverting H is the real number

$$\mathbf{Adv}_H^{\mathrm{inv}}(A) = \Pr\Big[E \xleftarrow{\$} \mathrm{Bloc}(\kappa, n);\ \sigma \xleftarrow{\$} R;\ M \leftarrow A^{E,E^{-1}}(\sigma) :\ H^E(M) = \sigma\Big] \qquad \diamondsuit$$

The PGV hash functions. Fig. 1 and Fig. 2 serve to define $f_\imath[n]\colon \mathrm{Bloc}(\kappa, n) \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ and $\hat{f}_\jmath[n]\colon \mathrm{Bloc}(\kappa, n) \times (\{0,1\}^n \times \{0,1\}^n) \to \{0,1\}^n$

for $\imath \in [1..20]$ and $\jmath \in [1..64]$. These compression functions induce hash functions $H_\imath[n]$ and $\hat{H}_\jmath[n]$. Usually we omit writing the $[n]$.

Discussion. The more customary formalization for a one-way function speaks to the difficulty of finding a preimage for the image of a random domain point (as opposed to finding a preimage of a random range point). But a random-domain-point definition becomes awkward when considering a function H with an infinite domain: in such a case one would normally have to partition the domain into finite sets and insist that H be one-way on each of them. For each of the functions $H_1, \ldots, H_{20}$, the value $H_\imath^E(M)$ is uniform or almost uniform in $\{0,1\}^n$ when M is selected uniformly in $(\{0,1\}^n)^m$ and E is selected uniformly in $\mathrm{Bloc}(n,n)$. Thus there is no essential difference between the two notions in these cases. This observation justifies defining inversion resistance in the manner that we have. See Appendix B.

Definition 3 might be understood as giving the technical meaning of *preimage resistance.* However, a stronger notion of preimage resistance also makes sense, where the range value σ is a fixed point, not a random one, and one maximizes over all such points. Similarly, the usual, random-domain-point notion for a one-way function (from the prior paragraph) might be understood as a technical meaning of *2nd preimage resistance*, but a stronger notion makes sense, where the domain point M is a fixed string, not a random one, and one must maximize over all domain points of a given length. A systematic exploration of different notions of inversion resistance is beyond the scope of this paper.

Conventions. For the remainder of this paper we assume the following significant conventions. First, an adversary does not ask any oracle query in which the response is already known; namely, if A asks a query $E_k(x)$ and this returns y, then A does not ask a subsequent query of $E_k(x)$ or $E_k^{-1}(y)$; and if A asks $E_k^{-1}(y)$ and this returns x, then A does not ask a subsequent query of $E_k^{-1}(y)$ or $E_k(x)$. Second, when a (collision-finding) adversary A for H outputs M and M', adversary A has already computed $H^E(M)$ and $H^E(M')$, in the sense that A has made the necessary E or E^{-1} queries to compute $H^E(M)$ and $H^E(M')$. Similarly, we assume that a (collision-finding) adversary A for the compression function f computes $f^E(h,m)$ and $f^E(h',m')$ prior to outputting (h,m) and (h',m'). Similarly, when an (inverting adversary) A for H outputs a message M, we assume that A has already computed $H^E(M)$, in the sense that A has made the necessary E or E^{-1} queries to compute this value. These assumption are all without loss of generality, in that an adversary A not obeying these conventions can easily be modified to given an adversary A' having similar computational complexity that obeys these conventions and has the same advantage as A.

3 Collision Resistance of the Group-1 Schemes

The group-1 hash-functions $H_1, \ldots, H_{12}$ can all be analyzed using the Merkle-Damgård paradigm. Our security bound is identical for all of these schemes.

Theorem 1 (Collision resistance of the group-1 hash functions). Fix $n \geq 1$ and $\imath \in [1..12]$. Then $\mathbf{Adv}^{\mathrm{coll}}_{H_\imath[n]}(q) \leq q(q+1)/2^n$ for any $q \geq 1$. $\diamond$

The proof combines a lemma showing the collision-resistance of $f_1, \ldots, f_{12}$ with the classical result, stated for the black-box model, showing that a hash function is collision resistant if its compression function is.

Lemma 1 (Merkle-Damgård [?, ?] in the black-box model). Let f be a compression function f: $\mathrm{Bloc}(n,n) \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ and let H be the iterated hash of f. Then $\mathbf{Adv}^{\mathrm{coll}}_{H}(q) \leq \mathbf{Adv}^{\mathrm{comp}}_{f}(q)$ for all $q \geq 1$. $\diamond$

Lemma 2 (Collision resistance of the group-1 compression functions). Fix $n \geq 1$ and $\imath \in [1..12]$. Then $\mathbf{Adv}^{\mathrm{comp}}_{f_\imath[n]}(q) \leq q(q+1)/2^n$ for any $q \geq 1$. $\diamond$

Proof of Lemma 2: Fix a constant $h_0 \in \{0,1\}^n$. We focus on $f = f_1$; assume that case. Let $A^{?,?}$ be an adversary attacking the compression function f. Assume that A asks its oracles a total of q queries. We are interested in A's behavior when its left oracle is instantiated by $E \xleftarrow{\$} \mathrm{Bloc}(n,n)$ and its right oracle is instantiated by E^{-1}. That experiment is identical, from A's perspective, to the one defined in Fig. 4. Define $((x_1, k_1, y_1), \ldots, (x_q, k_q, y_q), \mathit{out})$ by running *SimulateOracles*(A, n). If A is successful it means that A outputs $(k, m), (k', m')$ such that one of the following holds: $(k, m) \neq (k', m')$ and $f(k, m) = f(k', m')$, or else $f(k, m) = h_0$. By our definition of f this means that $E_k(m) \oplus m = E_{k'}(m') \oplus m'$ for the first case, or $E_k(m) \oplus m = h_0$ for the second. By our conventions at the end of Section 2, either there are distinct $r, s \in [1..q]$ such that $(x_r, k_r, y_r) = (m, k, E_k(m))$ and $(x_s, k_s, y_s) = (m', k', E_{k'}(m'))$ and $E_{k_r}(m_r) \oplus m_r = E_{k_s}(m_s) \oplus m_s$ or else there is an $r \in [1..q]$ such that $(x_r, k_r, y_r) = (m, k, h_0)$ and $E_{k_r}(x_r) = h_0$. We show that this event is unlikely.

In the execution of *SimulateOracles*(A, n), for any $i \in [1..q]$, let C_i be the event that $y_i \oplus x_i = h_0$ or that there exists $j \in [1..i-1]$ such that either $y_i \oplus x_i = y_j \oplus x_j$. In carrying out the simulation of A's oracles, either y_i or x_i was randomly selected from a set of at least size $2^n - (i-1)$, so $\Pr[\mathsf{C}_i] \leq i/(2^n - i)$.

Algorithm *SimulateOracles*(A, n)
Initially, $i \leftarrow 0$ and $E_k(x) = \mathsf{undefined}$ for all $(k, x) \in \{0,1\}^n \times \{0,1\}^n$
Run $A^{?,?}$, answering oracle queries as follows:
  When A asks a query (k, x) to its left oracle:
    $i \leftarrow i + 1$; $k_i \leftarrow k$; $x_i \leftarrow x$; $y_i \xleftarrow{\$} \overline{\mathrm{Range}}(E_k)$; $E_k(x) \leftarrow y_i$; return y_i to A
  When A asks a query (k, y) to its right oracle:
    $i \leftarrow i + 1$; $k_i \leftarrow k$; $y_i \leftarrow y$; $x_i \xleftarrow{\$} \overline{\mathrm{Domain}}(E_k)$; $E_k(x_i) \leftarrow y$; return x_i to A
When A halts, outputting a string *out*:
  return $((x_1, k_1, y_1), \ldots, (x_i, k_i, y_i), \mathit{out})$

Fig. 4. Simulating a block-cipher oracle. $\mathrm{Domain}(E_k)$ is the set of points x where $E_k(x)$ is no longer $\mathsf{undefined}$ and $\overline{\mathrm{Domain}}(E_k) = \{0,1\}^n - \mathrm{Range}(E_k)$. $\mathrm{Range}(E_k)$ is the set of points where $E_k(x)$ is no longer $\mathsf{undefined}$ and $\overline{\mathrm{Range}}(E_k) = \{0,1\}^n - \mathrm{Range}(E_k)$.

By the contents of the previous paragraph, we thus have that $\mathbf{Adv}^{\mathrm{comp}}_{f[n]}(A) \le \Pr[\mathsf{C}_1 \vee \ldots \vee \mathsf{C}_q] \le \sum_{i=1}^{q} \Pr[\mathsf{C}_i] \le \sum_{i=1}^{q} \frac{i}{2^n-(i-1)} \le \frac{1}{2^n-2^{n-1}} \sum_{i=1}^{q} i$ if $q \le 2^{n-1}$. Continuing, our expression is at most $\frac{1}{2^{n-1}} \frac{q(q+1)}{2} = \frac{q(q+1)}{2^n}$. Since the above inequality is vacuous when $q > 2^{n-1}$, we may now drop the assumption that $q \le 2^{n-1}$. We conclude that $\mathbf{Adv}^{\mathrm{comp}}_{f[n]}(q) \le q(q+1)/2^n$.

The above concludes the proof for the case of f_1. Compression functions $f_{2..12}$ are similar. ▌

4 Collision Resistance of the Group-2 Schemes

We cannot use the Merkle-Damgård paradigm for proving the security of $H_{13..20}$ because their compression functions are *not* collision-resistant. Attacks for each compression function are easy to find. For example, one can break $f_{17}(h,m) = E_m(h) \oplus m$ as a compression function by choosing any two distinct $m, m' \in \{0,1\}^n$, computing $h = E_m^{-1}(m)$ and $h' = E_{m'}^{-1}(m')$, and outputting (h,m) and (h',m'). All the same, hash functions $H_{13..20}$ enjoy almost the same collision-resistance upper bound as $H_{1..12}$.

Theorem 2 (Collision resistance of the group-2 hash functions). Fix $n \ge 1$ and $\imath \in [13..20]$. Then $\mathbf{Adv}^{\mathrm{coll}}_{H_\imath[n]}(q) \le 3q(q+1)/2^n$ for all $q \ge 1$. ◇

Proof of Theorem 2: Fix constants $h_0, v \in \{0,1\}^n$. We prove the theorem for the case of H_{13}, where $f(h,m) = f_{13}(h,m) = E_{h \oplus m}(m) \oplus v$.

We define a directed graph $G = (V_G, E_G)$ with vertex set $V_G = \{0,1\}^n \times \{0,1\}^n \times \{0,1\}^n$ and an arc $(x,k,y) \to (x',k',y')$ in E_G if and only if $k' \oplus x' = y \oplus v$.

Let $A^{?,?}$ be an adversary attacking H_{13}. We analyze the behavior of A when its left oracle is instantiated by $E \xleftarrow{\$} \mathrm{Bloc}(n,n)$ and its right oracle is instantiated by E^{-1}. Assume that A asks its oracles at most q total queries. We must show that $\mathbf{Adv}^{\mathrm{coll}}_{H_{13}[n]}(A) \le 3q(q+1)/2^n$. Run the algorithm *SimulateOracles*(A,n). As A executes with its (simulated) oracle, color the vertices of G as follows:

- Initially, each vertex of G is *uncolored*.
- When A asks an E-query (k,x) and this returns a value y, or when A asks an E^{-1}-query of (k,y) and this returns x, then: if $x \oplus k = h_0$ then vertex (x,k,y) gets colored *red*; otherwise vertex (x,k,y) gets colored *black*.

According to the conventions at the end of Section 2, every query the adversary asks results in exactly one vertex getting colored red or black, that vertex formerly being uncolored.

We give a few additional definitions. A vertex of G is colored when it gets colored red or black. A path P in G is colored if all of its vertices are colored. Vertices (x,k,y) and (x',k',y') are said to collide if $y = y'$. Distinct paths P and P' are said to collide if all of their vertices are colored and they begin with red vertices

$\imath$	$h_i =$	$(x,k,y) \to (x',k',y')$ if	(x,k,y) red if	(x,k,y), (x',k',y') collide if
13	$E_{w_i}(m_i) \oplus v$	$y \oplus v = x' \oplus k'$	$x \oplus k = h_0$	$y = y'$
14	$E_{w_i}(m_i) \oplus w_i$	$k \oplus y = x' \oplus k'$	$x \oplus k = h_0$	$k \oplus y = k' \oplus y'$
15	$E_{m_i}(h_{i-1}) \oplus v$	$y \oplus v = x'$	$x = h_0$	$y \oplus v = x'$
16	$E_{w_i}(h_{i-1}) \oplus v$	$y \oplus v = x'$	$x = h_0$	$y \oplus v = x'$
17	$E_{m_i}(h_{i-1}) \oplus m_i$	$k \oplus y = x'$	$x = h_0$	$k \oplus y = k' \oplus y'$
18	$E_{w_i}(h_{i-1}) \oplus w_i$	$k \oplus y = x'$	$x = h_0$	$k \oplus y = k' \oplus y'$
19	$E_{m_i}(w_i) \oplus v$	$y \oplus v = x' \oplus k'$	$x \oplus k = h_0$	$y = y'$
20	$E_{m_i}(w_i) \oplus m_i$	$k \oplus y = x' \oplus k'$	$x \oplus k = h_0$	$k \oplus y = k' \oplus y'$

Fig. 5. Rules for the existence of arcs, the coloring of a vertex red, and when vertices are said to collide. These notions are used in the proof of Theorem 2.

and they end with colliding vertices. Let C be the event that, as a result of the adversary's queries, there are formed in G some two colliding paths.

Claim 1. $\mathbf{Adv}^{\mathrm{coll}}_{H_{13}[n]}(A) \leq \Pr[\mathsf{C}]$.

Claim 2. $\Pr[\mathsf{C}] \leq 3q(q+1)/2^n$.

The theorem follows immediately from these two claims, whose proofs can be found in the full paper [**?**]. Proofs for for $H_{14..20}$ can be obtained by adapting the proof for H_{13} using the rules from Fig. 5. ▮

5 Matching Attacks on Collision Resistance

In this section we show that the security bounds given in Sections 3 and 4 are tight: we devise and analyze attacks that achieve advantage close to the earlier upper bounds. Our results are as follows.

Theorem 3 (Finding collisions in $H_{1..4}$). Let $\imath \in [1..4]$ and $n \geq 1$. Then $\mathbf{Adv}^{\mathrm{coll}}_{H_\imath[n]}(q) \geq 0.039(q-1)(q-3)/2^n$ for any even $q \in [1..2^{(n-1)/2}]$. ◇

Let $\mathrm{Perm}(n)$ be the set of all permutations on $\{0,1\}^n$. Let $\mathcal{P}_q(\{0,1\}^n)$ denote the set of all q-element subsets of $\{0,1\}^n$. The proof of Theorem 3 uses the following technical lemma whose proof appears in the full paper [**?**].

Lemma 3. Fix $n \geq 1$. Then

$$\Pr\Big[\pi \stackrel{\$}{\leftarrow} \mathrm{Perm}(n);\, Q \stackrel{\$}{\leftarrow} \mathcal{P}_q(\{0,1\}^n) \;:\; \exists x, x' \in Q \text{ such that } x \neq x' \text{ and } \pi(x) \oplus x = \pi(x') \oplus x'\Big] \geq .039(q-1)(q-3)/2^n$$

for any even $q \in [1..2^{(n-1)/2}]$. ◇

Proof of Theorem 3: Consider the case $H^E = H_1^E$ and fix $h_0 \in \{0,1\}^n$. Let A be an adversary with the oracles E, E^{-1}. Let A select $m_1, \ldots, m_q \xleftarrow{\$} \{0,1\}^n$ and compute $y_j = E_{h_0}(m_j) \oplus m_j$, $j \in [1..q]$. If A finds $r, s \in [1..q]$ such that $r < s$ and $y_r = y_s$ then it returns (m_r, m_s); otherwise it returns (m_1, m_1) (failure). Let $\pi = E_{h_0}$. By definition π is a uniform element in Perm(n), so we can invoke Lemma 3 to see that the probability that A succeeds to find a collision among $m_1, \ldots, m_q$ under H is at least $.039(q-1)(q-3)/2^n$.

This attack and analysis extends to $H_{2..4}$ by recognizing that for each scheme and distinct one-block messages m and m' we have $H^E(m) = H^E(m')$ if and only if $\pi(x) \oplus x = \pi(x') \oplus x'$ where $\pi = E_{h_0}$ and x, x' are properly defined. For example, for H_2^E define $x = h_0 \oplus m$ and $x' = h_0 \oplus m'$. ▮

Analysis of collision-finding attacks on $H_{5..20}$ is considerably less technical than for $H_{1..4}$. The crucial difference is that in each of $H_{5..20}$ the block cipher is keyed in the first round by either the message m, or $m \oplus h_0$, where h_0 is a fixed constant. Hence when A hashes q distinct one-block messages it always observes q random values. See the full paper [**?**] for a proof of the following.

Theorem 4 (Finding collisions in $H_{5..20}$). Let $\imath \in [5..20]$ and $n \geq 1$. Then $\mathbf{Adv}^{\mathrm{coll}}_{H_\imath[n]}(q) \geq 0.3q(q-1)/2^n$ for any $q \in [1..2^{n/2}]$. ◇

6 Security of the Schemes as OWFs

From the perspective of collision resistance there is no reason to favor any particular scheme from $H_{1..20}$. However, in this section we show that these schemes can be separated based on their strength as one-way functions. In particular, for an n-bit block cipher, an adversary attacking a group-1 hash function requires nearly 2^n oracle queries to do well at inverting a random range point, while an adversary attacking a group-2 hash function needs roughly $2^{n/2}$ oracle queries to do the same job.

We begin with the theorem establishing good inversion-resistance for the group-1 schemes. The theorem is immediate from the two lemmas that follow it. The first result is analogous to Lemma 1. The second result shows that $f_{1..12}$ have good inversion-resistance. All omitted proofs can be found in the full version of the paper [**?**].

Theorem 5 (OWF security of the group-1 hash functions). Fix $n \geq 1$ and $\imath \in [1..12]$. Then $\mathbf{Adv}^{\mathrm{inv}}_{H_\imath[n]}(q) \leq q/2^{n-1}$ for any $q \geq 1$. ◇

Lemma 4 (Merkle-Damgård for inversion resistance). Let f be a compression function f: Bloc$(n,n) \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ and let H be the iterated hash of f. Then $\mathbf{Adv}^{\mathrm{inv}}_{H}(q) \leq \mathbf{Adv}^{\mathrm{inv}}_{f}(q)$ for all $q \geq 1$. ◇

Lemma 5 (Inversion resistance of the group-1 compression functions). Fix $n \geq 1$ and $\imath \in [1..12]$. Then $\mathbf{Adv}^{\mathrm{inv}}_{f_\imath[n]}(q) \leq q/2^{n-1}$ for any $q \geq 1$. ◇

Proof of Lemma 5: Fix a constant $h_0 \in \{0,1\}^n$. We focus on compression function $f^E = f_1^E$; assume that case. Let A be an adversary with oracles E, E^{-1} and input σ. Assume that A asks its oracles q total queries.

Define $((x_1,k_1,y_1),\ldots,(x_q,k_q,y_q), out)$ by running *SimulateOracles*(A,n). By our conventions at the end of Section 2, if A outputs (h,m) such that $E(h,m)\oplus m = \sigma$ then $(m,h,E(h,m)) = (x_i,k_i,y_i)$ for some $i \in [1..q]$. Let C_i be the event that (x_i,k_i,y_i) is such that $x_i \oplus y_i = \sigma$. In carrying out the simulation of A's oracles, either x_i or y_i was randomly assigned from a set of at least size $2^n - (i-1)$, so $\Pr[\mathsf{C}_i] \le 1/(2^n-(i-1))$. Thus $\Pr[(h,m) \leftarrow A^{E,E^{-1}}(z):\ E(h,m)\oplus m = \sigma] \le \Pr[\mathsf{C}_1 \vee \ldots \vee \mathsf{C}_q] \le \sum_{i=1}^{q}\Pr[\mathsf{C}_i] \le \sum_{i=1}^{q}\frac{1}{2^n-(i-1)} \le \frac{q}{2^n-2^{n-1}}$ if $q \le 2^{n-1}$. Continuing, our expression is at most $\frac{q}{2^{n-1}}$. Since the above inequality is vacuous when $q > 2^{n-1}$, we may now drop the assumption that $q \le 2^{n-1}$.

The above concludes the proof for the case of f_1. Compression functions $f_{2..12}$ are similar. ▮

We cannot use Lemma 4 to prove the security of the group-2 schemes because the associated compression functions are *not* inversion-resistant. An attack for each is easy to find. For example, consider $f_{13}(h,m) = E(h\oplus m, m)\oplus v$. For any point σ, the adversary fixes $k=0$, computes $m = E_0^{-1}(\sigma \oplus v)$, and returns (m,m), which is always a correct inverse to σ. Still, despite these compression functions being invertible with a single oracle query, there is a reasonable security bound for the group-2 schemes.

Theorem 6 (OWF security of the group-2 hash functions). Fix $n \ge 1$ and $\imath \in [13..20]$. Then $\mathbf{Adv}^{\mathrm{inv}}_{H_\imath[n]}(q) \le 9(q+3)^2/2^n$ for any $q > 1$. ◇

The proof of Theorem 6 makes use of the following lemma, which guarantees that, up to a constant, for messages of length greater than n-bits, the bounds we have computed for collision resistance hold for inversion resistance as well.

Lemma 6 (Collision resistance ⇒ inversion resistance). Fix $\imath \in [1..20]$ and $n \ge 1$. Let $\tilde{H} = H_\imath[n]$ restricted to domain $\mathrm{Bloc}(n,n) \times \bigcup_{i\ge 2}\{0,1\}^{in}$. Then $\mathbf{Adv}^{\mathrm{inv}}_{\tilde{H}}(q) \le 3\mathbf{Adv}^{\mathrm{coll}}_{H}(q+2) + q/2^{n-1}$ for any $q \ge 1$. ◇

Finally, we prove that the security bounds given in Theorems 5 and 6 are tight, by describing adversaries that achieve advantage very close to the upper bounds. The analysis falls into three groupings.

Theorem 7 (Attacking $H_{1..4}$ as OWFs). Fix $n \ge 1$ and $\imath \in [1..4]$. Then $\mathbf{Adv}^{\mathrm{inv}}_{H_\imath[n]}(q) \ge 0.4q/2^n$ for any $q \in [1..2^{n-2}]$. ◇

Theorem 8 (Attacking $H_{5..12}$ as OWFs). Fix $n \ge 1$ and $\imath \in [5..12]$. Then $\mathbf{Adv}^{\mathrm{inv}}_{H_\imath[n]}(q) \ge 0.6q/2^n$ for any $q \in [1..2^n-1]$. ◇

Theorem 9 (Attacking $H_{13..20}$ as OWFs). Fix $n \ge 1$ and $\imath \in [13..20]$. Then $\mathbf{Adv}^{\mathrm{inv}}_{H_\imath[n]}(q) \ge 0.15q^2/2^n$ for any even $q \in [2..2^{n/2}]$. ◇

The proofs for the above three theorems appear in the full paper.

Acknowledgments

Thanks to the anonymous reviewers for helpful comments and references.

John Black received support from NSF CAREER award CCR-0133985. This work was carried out while John was at the University of Nevada, Reno.

Phil Rogaway and his student Tom Shrimpton received support from NSF grant CCR-0085961 and a gift from CISCO Systems. Many thanks for their kind support.

A Fatal Attacks on Five of PGV's B-Labeled Schemes

In [**?**] there are a total of 13 schemes labeled as "backward attackable." We have already shown that eight of these, $H_{13..20}$, are collision resistant. But the remaining five schemes are completely insecure; each can be broken with two queries. Consider, for example, $H = \hat{H}_{39}$, constructed by iterating the compression function $f = \hat{f}_{39}$ defined by $f^E(h_{i-1}, m_i) = E_{m_i \oplus h_{i-1}}(m_i \oplus h_{i-1}) \oplus m_i$. For any $c \in \{0,1\}^n$ the strings $(h_0 \oplus c) \parallel (E_c(c) \oplus h_0)$ hashes to h_0, and so it so it takes only two queries to produce a collision. Variants of this attack, break the schemes $\hat{H}_{40}$, $\hat{H}_{43}$, $\hat{H}_{55}$ and $\hat{H}_{59}$ defined in Fig. 1. Namely, for $\hat{H}_{40}$, messages $(h_0 \oplus c) \parallel (E_v(c) \oplus h_0)$ collide; for $\hat{H}_{43}$, $(h_0 \oplus c) \parallel (E_c(c) \oplus h_0 \oplus c)$; for $\hat{H}_{55}$, $(h_0 \oplus c) \parallel (E_c(v) \oplus h_0)$; for $\hat{H}_{59}$, $(h_0 \oplus c) \parallel (E_c(v) \oplus h_0 \oplus c)$.

B Two Notions of Inversion Resistance

We defined $\mathbf{Adv}_H^{\mathrm{inv}}$ by giving the adversary a random range point $\sigma \in \{0,1\}^n$ and asking the adversary to find an H-preimage for σ. The usual definition for a one-way function has one choose a random domain point M, apply H, and ask then ask the adversary to invert the result.

Definition 4 (Conventional definition of a OWF). Let H be a block-cipher-based hash function, H: $\mathrm{Bloc}(\kappa, n) \times D \to R$, and let ℓ be a number such that $\{0,1\}^\ell \subseteq D$. Let A be an adversary. Then the advantage of A in inverting H on the distribution induced by applying H to a random ℓ-bit string is the real number

$$\mathbf{Adv}_H^{\mathrm{owf}}(A, \ell) = \Pr\Big[E \xleftarrow{\$} \mathrm{Bloc}(\kappa, n);\ M \xleftarrow{\$} (\{0,1\}^n)^\ell;\ \sigma \leftarrow H^E(M);\ M' \leftarrow A^{E,E^{-1}}(\sigma):\ H^E(M') = \sigma\Big] \qquad \diamondsuit$$

For $q \geq 0$ a number, $\mathbf{Adv}_H^{\mathrm{owf}}(q, \ell)$ is defined in the usual way, as the maximum value of $\mathbf{Adv}_H^{\mathrm{owf}}(A, \ell)$ over all adversaries A that ask at most q queries.

Though the $\mathbf{Adv}^{\mathrm{owf}}$ and $\mathbf{Adv}^{\mathrm{inv}}$ measures can, in general, be far apart, it is natural to guess that they coincide for "reasonable" hash-functions like $H_{1..20}$. In particular, one might think that the random variable $H_\imath^E(M)$ is uniformly distributed in $\{0,1\}^n$ if $M \xleftarrow{\$} \{0,1\}^{n\ell}$ and $E \xleftarrow{\$} \mathrm{Bloc}(n, n)$. Interestingly, this is not true. For example, experiments show that when $E \xleftarrow{\$} \mathrm{Bloc}(2, 2)$ and $M \xleftarrow{\$} \{0,1\}^4$

the string $H_1^E(M)$ takes on the value 00 more than a quarter of the time (in fact, 31.25% of the time) while each of the remaining three possible outputs (01, 10, 11) occur less than a quarter of the time (each occurs $22.91\overline{6}$% of the time). Still, for $H_{1..20}$, the two notions are close enough that we have used Definition 3 as a surrogate for Definition 4. The result is as follows.

Lemma 7. Fix $n \geq 1$ and $\imath \in [1..20]$. Then for any $q, \ell \geq 1$,

$$\left| \mathbf{Adv}^{\mathrm{inv}}_{H_\imath[n]}(q) - \mathbf{Adv}^{\mathrm{owf}}_{H_\imath[n]}(q, \ell) \right| \leq \ell / 2^{n-1} \qquad \diamondsuit$$

The proof of Lemma 7 is found in [**?**].