Security Analysis of a Design Variant of Randomized Hashing

Praveen Gauravaram[1], Shoichi Hirose[2], Douglas Stebila[3]

[1] Tata Consultancy Services, Australia
[2] University of Fukui, Japan
[3] McMaster University, Canada

Abstract. At EUROCRYPT 2009, Gauravaram and Knudsen presented an online birthday attack on the randomized hashing scheme standardized in NIST SP800-106. This attack uses a fact that it is easy to find fixed points for the Davies-Meyer-type compression functions of standardized hash functions such as those in the SHA-2 family. This attack is significant in that it is an attack on the target collision resistance (TCR) of the randomized hashing scheme which is claimed to be enhanced TCR (eTCR). TCR is a property weaker than eTCR. In this paper, we will present a randomized hashing scheme called RMC. We will also prove that RMC satisfies both TCR and eTCR in the random oracle model and in the ideal cipher model. In particular, the proof for the TCR security in the ideal cipher model implies that the attack by Gauravaram and Knudsen is not effective against RMC.

Keywords: Iterated hash function · Randomized hashing · Target collision resistance · Davies-Meyer compression function · Provable security
·

1 Introduction

Background. At EUROCRYPT 2009, Gauravaram and Knudsen [11] showed an online existential birthday forgery attack on the digital signatures based on a randomized hashing scheme that are enhanced Target-Collision-Resistant (eTCR) secure designed by Halevi and Krawczyk [13]. The randomized hashing was also standardized by U.S. National Institute of Standards and Technology in the SP 800-106 [22]. An interesting aspect of this attack is that it is an attack on the TCR property of the randomized hashing scheme. TCR is a property weaker than eTCR. Namely, an attack on the TCR property implies an attack on the eTCR property. In addition, the attack has a practical impact as it is applicable in the scenarios where a random value used as part of the signature computation is also used for randomized hashing, which is a recommended practice to save on the communication bandwidth from transmitting an additional random value used for randomized hashing.

Although digital signatures based on a randomized hashing scheme with the eTCR property have a practical advantage of not requiring to sign a random value along with the hash value, in some scenarios such as above, an attack on the eTCR property is not useful to forge randomize-hash-and-sign digital signatures [11] whereas an attack on the TCR property is. This argument is valid for both online and offline attacks on the eTCR property.

Our contribution. We will present a randomized hash function family which we call RMC. It simply feeds concatenation of the randomization input and a message block to each compression function in the iterated hash function. Similar to the randomized hash function family by Halevi and Krawczyk, RMC can be implemented without any modifications to iterated hash functions such as SHA-2 hash functions [7]. We will specify a preprocessing scheme for message input and randomization input to instantiate RMC with the use of iterated hash functions such as SHA-2 hash functions.

Actually, RMC is essentially equivalent to the strengthened Merkle-Damgård domain extension in the dedicated key setting [1] if instantiated with compression functions of SHA-2 hash functions. In the dedicated key setting, the underlying compression function takes as a part of input a key which is not secret but chosen uniformly at random. For compression functions of SHA-2 hash functions, it is natural to feed the key as a part of the message-block input.

Additionally, a negative result is shown for TCR and eTCR properties of RMC. It is shown that neither TCR nor eTCR are preserved by strengthened Merkle-Damgård in the dedicated key setting by Bellare and Ristenpart [1] and by Reyhanitabar, Susilo and Mu [24]. This also applies to RMC. Namely, RMC does not necessarily satisfy TCR and eTCR even if the underlying compression function satisfies TCR and eTCR, respectively.

In this paper, we will give a positive result on TCR and eTCR properties of RMC on a different assumption on the underlying compression function. More precisely, we will show that RMC satisfies both TCR and eTCR if the underlying compression function is an ideal primitive. The result implies that RMC provides better security with respect to TCR than the Halevi-Krawczyk randomized hash function family. In particular, it implies that RMC is secure against the online TCR attack by Gauravaram and Knudsen [11].

We remark that the idea for our RMC design has originated from the randomized hash function variant [8] wherein inputs to the RMX hash function were randomized at both prefix and suffix ends. Although the Gauravaram-Knudsen attack is not applicable to this variant due to suffix randomization, this attack can be combined with the herding-style attack [14] to mount an online birthday forgery attack which does not apply to RMC.

Organization. Some basic notions are introduced and the RMX randomized hash function family is reviewed in Sect. 2. The security notions of TCR and eTCR are formally defined for randomized hash function family in Sect. 3. The RMC randomized hash function family is presented in Sect. 4. It is also shown in the same section that the RMC hash function family satisfies TCR and eTCR in the random oracle model and in the ideal cipher model. In Section 5, a preprocessing scheme for message input and randomization input is described, which is used for instantiating the RMC randomized hash function family with widely deployed iterated hash functions such as SHA-2 hash functions.

2 Definitions

Let $\{0,1\}^*$ be the set of the binary sequences of arbitrary length including the empty sequence. The length of $x \in \{0,1\}^*$ is denoted by $|x|$. For x and y in $\{0,1\}^*$, $x\|y$ is their concatenation. $a \twoheadleftarrow A$ means that an element is chosen uniformly at random from a finite set A and assigned to a.

2.1 Deterministic hash function

A hash function takes as input an arbitrary-length message and outputs a fixed-length digest. A hash function is usually constructed by iterating a compression function, which takes as input a fixed-length message and outputs a fixed-length digest, by applying a mode of operation or domain extension transform such as Merkle-Damgård (MD) [5, 19]. In this paper, we consider MD as the hash function mode of operation albeit the extension of our analysis to other modes.

Let $f : \{0,1\}^n \times \{0,1\}^b \to \{0,1\}^n$ be a compression function which takes as input a b-bit message block and an n-bit chaining value and outputs a new n-bit chaining value. The MD mode of operation iterated over f takes as input a message M of length a multiple of b. M is divided into b-bit message blocks $M[1], M[2], \ldots, M[m]$, and is processed with MD to obtain the digest. The MD mode of operation iterated over f with an initialization vector IV, denoted by MD^f, is formally defined as follows: $\mathrm{MD}^f(IV, M) = V[m]$, where $V[0] \leftarrow IV$, $M[1]\|M[2]\| \cdots \|M[m] \leftarrow M$ and $V[i] \leftarrow f(V[i-1], M[i])$ for $i = 1$ to m.

Let $H^f : \{0,1\}^n \times \{0,1\}^* \to \{0,1\}^n$ be a deterministic hash function constructed by using the MD mode of operation iterated over f. H^f takes as input a message M of arbitrary length. With the application of a padding procedure, M is extended as $M\|pad$, which is processed by MD^f. The length of $M\|pad$ is a multiple of b. pad usually depends only on the length of M. A hash function H^f with an initialization vector IV is formally defined as $H^f(IV, M) = \mathrm{MD}^f(IV, M\|pad)$.

For deterministic hash functions such as SHA-256 and SHA-512, their initialization vectors are fixed and public. Thus, we will use the notations $\mathrm{MD}^f(M)$ and $H^f(M)$.

2.2 Randomized hash function family and RMX

A randomized hash function family is defined by a deterministic hash function with an auxiliary randomization input. Randomized hash function families were first introduced by Naor and Yung in the name of universal one-way hash functions (UOWHFs) [21]. The UOWHFs were called target-collision-resistant (TCR) hash functions by Bellare and Rogaway [2] and they satisfy TCR property which is *weaker* than collision resistance. Bellare and Rogaway [2] and later

Shoup [25] proposed and analyzed composition constructions to build TCR iterated hash functions from TCR compression functions. Halevi and Krawczyk [13] designed randomized hash functions with TCR and *stronger* property of enhanced Target Collision Resistance (eTCR) by using properties related to second preimage resistance of the compression function. One of their eTCR designs is called RMX.

The scope of this paper is in proposing design improvements for RMX hash function family, and we limit our design description to RMX. An RMX hash function family over H^f is defined by $\bar{\mathcal{H}} = \left\{\bar{H}^f_r \mid r \in \{0,1\}^c\right\}$, where $\bar{H}^f_r(M) = H^f(r \| (r \oplus M[1]) \| (r \oplus M[2]) \| \cdots \| (r \oplus M[m]))$. For simplicity, it is assumed that the length c of r equals the message-block length of f. It is also assumed that $M = M[1] \| M[2] \| \cdots \| M[m]$ and $|M[i]|$ equals the message-block length of f. The detailed specification for the general cases is given in NIST SP 800-106 [22].

2.3 Fixed points in block-cipher-based compression functions

Several practical block-cipher-based compression functions [23] such as Davies-Meyer [20], Matyas-Meyer-Oseas [18] and Miyaguchi-Preneel [23], that are provably collision resistant and (second) preimage resistant in the ideal cipher model [3, 4, 26], are easily differentiable from a fixed-input-length random oracle [17]. For example, it is easy to find *fixed points* for the Davies-Meyer compression function [20]. This weakness was exploited in several attacks on popular hash function frameworks [6, 9–12, 15, 16]. These attacks make use of *fixed points* in compression functions to generate birthday collision attacks that are used to find second preimages in much less than generic second preimage attack complexity.

3 TCR and eTCR of randomized hash function family

Let $\mathcal{H}$ be a randomized hash function family using a deterministic iterated hash function H^f randomized with an auxiliary random input. We formalize multi-target (enhanced) target-collision-resistance using the experiments given below:

$\underline{\mathrm{Exp}^{\mathsf{TCR}\text{-}t}_{\mathcal{H}}}$:
1. $st \leftarrow \perp$; $r_0 \leftarrow \perp$
2. For $i = 1$ to t: $(M_i, st) \leftarrow \mathcal{A}^f(r_{i-1}, st)$; $r_i \twoheadleftarrow \{0,1\}^c$
3. $(M^*, r^*) \leftarrow \mathcal{A}^f(\boldsymbol{r}, st)$
4. WIN iff $\exists i : (M_i \neq M^*) \wedge (r_i = r^*) \wedge (H^f_{r_i}(M_i) = H^f_{r^*}(M^*))$

$\underline{\mathrm{Exp}^{\mathsf{eTCR}\text{-}t}_{\mathcal{H}}}$:
1. $st \leftarrow \perp$; $r_0 \leftarrow \perp$
2. For $i = 1$ to t: $(M_i, st) \leftarrow \mathcal{A}^f(r_{i-1}, st)$; $r_i \twoheadleftarrow \{0,1\}^c$
3. $(M^*, r^*) \leftarrow \mathcal{A}^f(\boldsymbol{r}, st)$
4. WIN iff $\exists i : (r_i, M_i) \neq (r^*, M^*) \wedge (H^f_{r_i}(M_i) = H^f_{r^*}(M^*))$

An experiment is a game played by an adversary $\mathcal{A}$. $\mathcal{A}$ is given t first preimages. For each first preimage (M_i, r_i), message M_i is chosen by $\mathcal{A}$ adaptively, and the corresponding randomization input r_i is chosen uniformly at random after M_i. $\mathcal{A}$ wins in the experiment if $\mathcal{A}$ finds a second preimage for one of the given t

first preimages. The experiment for TCR requires that the randomization input of the second preimage is equal to that of the first preimage.

The TCR advantage of $\mathcal{A}$ is defined as follows:

$$\mathrm{Adv}_{\mathcal{H}}^{\mathsf{TCR}\text{-}t}(\mathcal{A}) = \Pr\big[\mathcal{A} \text{ wins in } \mathrm{Exp}_{\mathcal{H}}^{\mathsf{TCR}\text{-}t}\big] \ .$$

The eTCR advantage $\mathrm{Adv}_{\mathcal{H}}^{\mathsf{eTCR}\text{-}t}$ is defined analogously.

4 RMC hash function family

We propose RMC as a randomized hash function family which offers better security bounds against TCR attacks than the RMX hash function family. Let $\tilde{\mathcal{H}}$ be an RMC hash function family which uses MD mode as the underlying domain extension. A hash function $\tilde{H}_r^f$ in this family is formally defined as follows: $\tilde{H}_r^f(M) = H^f(r\|M[1])\|(r\|M[2])\| \cdots \|(r\|M[m])$, where $M[1]\|M[2]\| \cdots \|M[m] \leftarrow M$. Here, $r \in \{0,1\}^c$, and it is assumed that $b > c$, $|M| \equiv 0 \pmod{b-c}$ and $|M[i]| = b - c$ for $1 \leq i \leq m$. For arbitrary-length messages, a preprocessing function producing inputs to the deterministic hash function H^f is specified in the next section.

4.1 Rationale for the design choice of RMC

The key criterion is to choose a design so that an RMC randomized hash function family is not vulnerable to length-extended *fixed-point*-based birthday collision attacks used to find *online* TCR collision attacks on RMX hash functions [11, 12]. In a length-extended *fixed-point*-based birthday collision attack on an iterated hash function H^f, an adversary develops a colliding pair $(M, M\|M[\ell+1])$ such that $H(M) = H(M\|M[\ell+1])$, where M is an arbitrary ℓ-block message and $M[\ell+1]$ is a *fixed-point* message block for f such that $f(H(M), M[\ell+1]) = H(M)$. As demonstrated in [11, 12], this attack is also applicable on RMX randomized hash function families in the following way:

1. Adversary $\mathcal{A}$ is given t preimages $(M_1, r_1), (M_2, r_2), \ldots, (M_t, r_t)$.
2. $\mathcal{A}$ produces s random fixed points for f such that $f(V_j, N_j) = V_j$ for $1 \leq j \leq s$.
3. If $\mathcal{A}$ finds some i and j such that $\bar{H}_{r_i}^f(M_i) = V_j$, then $\mathcal{A}$ outputs $(M_i\|(r_i \oplus N_j), r_i)$ as the second preimage for (M_i, r_i).

This attack is a TCR collision attack in birthday complexity since it is successful with some significant probability if $ts = O(2^n)$.

This attack cannot be applied to RMC randomized hash function families as the compression function always takes a randomization input.

4.2 Security analysis

The TCR and eTCR security of the RMC randomized hash function family $\tilde{\mathcal{H}} = \left\{\tilde{H}_r^f \;\middle|\; f : \{0,1\}^n \times \{0,1\}^b \to \{0,1\}^n \wedge r \in \{0,1\}^c\right\}$ are analyzed in the random oracle model and in the ideal cipher model. In the random oracle model, the compression function f is assumed to be a fixed-input-length random oracle. In the ideal cipher model, it is assumed to be a Davies-Meyer compression function, that is, $f(v,x) = E_x(v) \oplus v$, where the block cipher E with block size n and key size b is chosen uniformly at random. In these ideal models, the advantage of an adversary is evaluated based on the number of calls to the ideal primitive. Let

$$\mathrm{Adv}_{\mathcal{H}}^{\mathsf{TCR}\text{-}t}(\ell,q) = \max_{\mathcal{A}} \Pr\left[\mathcal{A} \text{ wins in } \mathrm{Exp}_{\mathcal{H}}^{\mathsf{TCR}\text{-}t}\right] \ ,$$

where each preimage given to $\mathcal{A}$ has at most ℓ message blocks and $\mathcal{A}$ calls f at most q times, which exclude the number of calls required to compute the outputs for the t first preimages. $\mathrm{Adv}_{\mathcal{H}}^{\mathsf{eTCR}\text{-}t}(\ell,q)$ is defined similarly. Notice that a call to f in the ideal cipher model is a call to encryption E or decryption E^{-1}.

Theorem 1 given below quantifies the TCR security of the RMC hash function family. The proof is omitted due to the page limit.

Theorem 1. *Let q, t and ℓ be positive integers. Let $\alpha = \min\{t, \lfloor(\mathrm{e}\ln 2)c/(\ln c + \ln\ln 2)\rfloor\}$. Suppose that $t \le 2^c$.*

1. *If f is a random oracle, then $\mathrm{Adv}_{\tilde{\mathcal{H}}}^{\mathsf{TCR}\text{-}t}(\ell,q) \le (\alpha\ell+1)(t\ell+q)/2^n + 1/2^c$.*
2. *If f is a Davies-Meyer compression function with an ideal cipher, then*

$$\mathrm{Adv}_{\tilde{\mathcal{H}}}^{\mathsf{TCR}\text{-}t}(\ell,q) \le \frac{(\alpha\ell+1)(t\ell+q)}{2^n - (\alpha\ell+q)} + \frac{1}{2^c} \ .$$

It is implied by the result for the ideal cipher model shown in Theorem 1 that the collision attack on the RMX hash function family is not effective against the RMC hash function family.

Theorem 2 gives upper bounds on the eTCR advantage both in the random oracle model and in the ideal cipher model. The proof is also omitted.

Theorem 2. *Let q, t and ℓ be positive integers.*

1. *If f is a random oracle, then $\mathrm{Adv}_{\tilde{\mathcal{H}}}^{\mathsf{eTCR}\text{-}t}(\ell,q) \le (t\ell+1)(t\ell+q)/2^n$.*
2. *If f is a Davies-Meyer compression function with an ideal cipher, then*

$$\mathrm{Adv}_{\tilde{\mathcal{H}}}^{\mathsf{eTCR}\text{-}t}(\ell,q) \le \frac{(t\ell+1)(t\ell+q)}{2^n - (t\ell+q)} \ .$$

5 Randomized message preprocessing for hash functions

A randomized message preprocessing algorithm for an iterated hash function is specified for instantiation of the RMC randomized hash function family with widely deployed iterated hash functions such as SHA-256 and SHA-512. It is assumed that the iterated hash function uses a compression function $f : \{0,1\}^n \times \{0,1\}^b \to \{0,1\}^n$ and the Merkle-Damgård strengthening. For the iterated hash function, let l be the length of the binary representation of the input length. For example, $n = 256$, $b = 512$ and $l = 64$ for SHA-256, and $n = 512$, $b = 1024$ and $l = 128$ for SHA-512.

The message preprocessing algorithm takes as input $r \in \{0,1\}^c$ chosen uniformly at random and a message $M \in \{0,1\}^*$. It is assumed that $l + 1 \leq b - c$.

The algorithm first pads the message M with 10^k, where k is the minimum non-negative integer such that $|M| + k + 1 \equiv (b - c) - (l + 1) \pmod{b - c}$. Then, it divides $M \| 10^k$ into the blocks $M[1], M[2], \ldots, M[m]$ such that $|M[i]| = b - c$ for $1 \leq i \leq m - 1$ and $|M[m]| = (b - c) - (l + 1)$. Finally, it produces $(r \| M[1]) \| (r \| M[2]) \| \cdots \| (r \| M[m])$.

From the TCR security analysis in Sect. 4, since it is assumed that $t \leq 2^c$, where t is the number of the first preimages, it is recommended that $c \geq 128$ for SHA-256 and SHA-512. In addition, it is reasonable to assume that $c \leq n$, where n is the output length and $n < b - l - 1$ for SHA-256 and SHA-512. If $c = 128$, the number of calls of RMC to the compression function is about $4/3$ and $8/7$ times larger than that of RMX for SHA-256 and SHA-512, respectively. Table 1 summarizes the comparison for some other values of c.

Table 1. Performance comparison between RMC and RMX. Each entry τ means that the number of calls of RMC to the compression function is about τ times larger than that of RMX.

c	128	256	384	512
SHA-256	4/3	2	n/a	n/a
SHA-512	8/7	4/3	8/5	2

Acknowledgements

The second author was supported in part by JSPS KAKENHI Grant Number JP16H02828.

References

1. Bellare, M., Ristenpart, T.: Hash functions in the dedicated-key setting: Design choices and MPP transforms. In: Arge, L., Cachin, C., Jurdzinski, T., Tarlecki, A. (eds.) ICALP 2007. LNCS, vol. 4596, pp. 399–410. Springer (2007)
2. Bellare, M., Rogaway, P.: Collision-resistant hashing: Towards making UOWHFs practical. In: Kaliski Jr., B.S. (ed.) CRYPTO '97. LNCS, vol. 1294, pp. 470–484. Springer (1997)
3. Black, J., Rogaway, P., Shrimpton, T.: Black-box analysis of the block-cipher-based hash-function constructions from PGV. In: Yung, M. (ed.) CRYPTO 2002. Lecture Notes in Computer Science, vol. 2442, pp. 320–335. Springer (2002)

4. Black, J., Rogaway, P., Shrimpton, T., Stam, M.: An analysis of the blockcipher-based hash functions from PGV. J. Cryptology 23(4), 519–545 (2010)
5. Damgård, I.: A design principle for hash functions. In: Brassard, G. (ed.) CRYPTO '89. LNCS, vol. 435, pp. 416–427. Springer (1990)
6. Dean, R.D.: Formal Aspects of Mobile Code Security. Ph.D. thesis, Princeton University (1999)
7. FIPS PUB 180-4: Secure Hash Standard (SHS) (2015)
8. Gauravaram, P.: Generation of randomized messages for cryptographic hash functions, US Patent 9444619 B2 (2016)
9. Gauravaram, P., Kelsey, J.: Linear-XOR and additive checksums don't protect Damgård-Merkle hashes from generic attacks. In: Malkin, T. (ed.) CT-RSA 2008. LNCS, vol. 4964, pp. 36–51. Springer (2008)
10. Gauravaram, P., Kelsey, J., Knudsen, L.R., Thomsen, S.S.: On hash functions using checksums. Int. J. Inf. Sec 9(2), 137–151 (2010)
11. Gauravaram, P., Knudsen, L.R.: On randomizing hash functions to strengthen the security of digital signatures. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 88–105. Springer (2009)
12. Gauravaram, P., Knudsen, L.R.: Security analysis of randomize-hash-then-sign digital signatures. J. Cryptology 25(4), 748–779 (2012)
13. Halevi, S., Krawczyk, H.: Strengthening digital signatures via randomized hashing. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 41–59. Springer (2006)
14. Kelsey, J., Kohno, T.: Herding hash functions and Nostradamus attack. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 183–200. Springer (2006)
15. Kelsey, J., Lucks, S.: Collisions and near-collisions for reduced-round Tiger. In: Robshaw, M.J.B. (ed.) FSE 2006. LNCS, vol. 4047, pp. 111–125. Springer (2006)
16. Kelsey, J., Schneier, B.: Second preimages on n-bit hash functions for much less than 2^n work. In: Cramer, R. (ed.) EUROCRYPT 2005. Lecture Notes in Computer Science, vol. 3494, pp. 474–490. Springer (2005)
17. Kuwakado, H., Morii, M.: Indifferentiability of single-block-length and rate-1 compression functions. IEICE Fundamentals 90-A(10), 2301–2308 (2007)
18. Matyas, S.M., Meyer, C.H., Oseas, J.: Generating strong one-way functions with cryptographic algorithm. IBM Technical Disclosure Bulletin 27, 5658–5659 (1985)
19. Merkle, R.C.: One way hash functions and DES. In: Brassard, G. (ed.) CRYPTO '89. LNCS, vol. 435, pp. 428–446. Springer (1990)
20. Miyaguchi, S., Ohta, K., Iwata, M.: Confirmation that some hash functions are not collision free. In: Damgård, I.B. (ed.) EUROCRYPT '90. LNCS, vol. 473, pp. 326–343. Springer (1990)
21. Naor, M., Yung, M.: Universal one-way hash functions and their cryptographic applications. In: Proceedings of the 21st Annual ACM Symposium on Theory of Computing. pp. 33–43 (1989)
22. NIST SP 800-106: Randomized Hashing for Digital Signatures (2009)
23. Preneel, B., Govaerts, R., Vandewalle, J.: Hash functions based on block ciphers: A synthetic approach. In: Stinson, D.R. (ed.) CRYPTO '93. LNCS, vol. 773, pp. 368–378. Springer (1993)
24. Reyhanitabar, M.R., Susilo, W., Mu, Y.: Enhanced target collision resistant hash functions revisited. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 327–344. Springer (2009)
25. Shoup, V.: A composition theorem for universal one-way hash functions. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 445–452. Springer (2000)
26. Stam, M.: Blockcipher-based hashing revisited. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 67–83. Springer (2009)