Breaking and Fixing Cryptophia's Short Combiner

Bart Mennink and Bart Preneel

Dept. Electrical Engineering, ESAT/COSIC, KU Leuven, and iMinds, Belgium
firstname.lastname@esat.kuleuven.be

Abstract. A combiner is a construction formed out of two hash functions that is secure if one of the underlying functions is. Conventional combiners are known not to support short outputs: if the hash functions have n-bit outputs the combiner should have at least almost $2n$ bits of output in order to be robust for collision resistance (Pietrzak, CRYPTO 2008). Mittelbach (ACNS 2013) introduced a relaxed security model for combiners and presented "Cryptophia's short combiner," a rather delicate construction of an n-bit combiner that achieves optimal collision, preimage, and second preimage security. We re-analyze Cryptophia's combiner and show that a collision can be found in two queries and a second preimage in one query, invalidating the claimed results. We additionally propose a way to fix the design in order to re-establish the original security results.

Keywords: hash functions, combiner, short, attack, collision resistance, preimage resistance.

1 Introduction

A hash function combiner is a construction with access to two or more hash functions, and which achieves certain security properties as long as sufficiently many underlying hash functions satisfy these security properties. The first to formally consider the principle of combiners were Herzberg [18] and Harnik et al. [17]. Two classical examples are the concatenation combiner $C^{H_1,H_2}_{\text{concat}}(M) = H_1(M) \,\|\, H_2(M)$ and xor combiner $C^{H_1,H_2}_{\text{xor}}(M) = H_1(M) \oplus H_2(M)$. Combiners function as an extra security barrier, still offering the desired security even if one of the hash functions gets badly broken. As such, combiners find a wide range of applications, including TLS [6–8] and SSL [15] for which the combiner security was analyzed by Fischlin et al. [14]. We refer to Lehmann's PhD thesis [22] for a comprehensive exposition of combiners.

A combiner is called *robust* for some security property if this property holds as long as at least one of the underlying hash functions does. For instance, a combiner C^{H_1,H_2} based on hash functions H_1, H_2 is called robust for collision resistance if a collision attack on the combiner implies an attack on H_1 and H_2. Note that $C^{H_1,H_2}_{\text{concat}}$ is clearly robust for collision resistance, but C^{H_1,H_2}_{xor} is not. On the other hand, C^{H_1,H_2}_{xor} is robust for pseudorandomness [14], while $C^{H_1,H_2}_{\text{concat}}$

D. Gritzalis et al. (Eds.): CANS 2014, LNCS 8813, pp. 50–63, 2014.

is not. Similar results can be obtained for other security properties such as (second) preimage resistance and MAC security [14,18]. Various multi-property robust combiners have been designed by Fischlin et al. [11–13]. (Without going into detail, we refer to interesting results on the security of $C_{\text{concat}}^{H_1,H_2}$ beyond robustness, by Joux [21], Nandi and Stinson [27], Hoch and Shamir [19, 20], Fischlin and Lehmann [10], and Mendel et al. [24].)

The concatenation combiner is robust for collision resistance, but its output size is the sum of the output sizes of the underlying hash functions. At CRYPTO 2006, Boneh and Boyen [3] analyzed the question of designing a collision robust combiner with a shorter output size. This question got subsequently answered negatively by Canetti et al. [4] and Pietrzak [28,29]. In detail, Pietrzak [29] demonstrated that no collision robust combiner from two n-bit hash functions exists with output length shorter than $2n - \Theta(\log n)$. A similar observation was recently made for (second) preimage resistance by Rjaško [31] and Mittelbach [25].

These negative results are in part credited to the rather stringent requirements the model of robustness puts on the construction, being the explicit existence of a reduction. At ACNS 2013, Mittelbach [26] introduced a relaxed model where the combiner is based on ideal hash functions and no explicit reduction is needed. Throughout, we will refer to this model as the ideal combiner model, as opposed to the standard reduction-based robust combiner model. Intuitively, the model captures the case of security of the combiner if one of the underlying hash functions is ideal but the other one is under full control of the adversary. While the ideal combiner model puts stronger requirements on the underlying primitives, it allows to bypass the limitations of the robust combiner model. Particularly, it enables analysis of more complex designs and combines well with the indifferentiability framework of Maurer et al. [23] and its application to hash functions by Coron et al. [5].

Yet, it turns out to still be highly non-trivial to construct a secure combiner in the ideal combiner model. For instance, the above-mentioned xor combiner is not secure: if $H_1 = \mathcal{R}$ is an ideal hash function, the adversary can simply define $H_2 = \mathcal{R}$. Also, ideal combiner security is not immediately achieved for straightforward generalizations of this xor combiner. As expected, the concatenation combiner is secure in the ideal combiner model, but recall that it has an output size of $2n$ bits.

Mittelbach [26] also introduced an ingenious n-bit combiner $C_{\text{mit}}^{H_1,H_2}$ from n-bit hash functions that – in the ideal combiner model – achieves optimal $2^{n/2}$ collision security and 2^n preimage and second preimage security. Mittelbach's combiner is also known as "Cryptophia's short combiner." The design circumvents the impossibility results of [3,4,28,29] on the existence of short combiners in the standard combiner model. $C_{\text{mit}}^{H_1,H_2}$ is additionally proven to be a secure pseudorandom function and MAC in the robust combiner model. This result has been awarded as the best student paper of ACNS 2013.

At a high level, Mittelbach's combiner is a keyed combiner defined as

$$C_{\text{mit}}^{H_1,H_2}(k, M) = H_1(\mathsf{prep}_1(k_1, k_2, k_3, M)) \oplus H_2(\mathsf{prep}_2(k_4, k_5, k_6, M)) \,,$$

where $k = (k_1, \ldots, k_6)$ is a fixed key, and prep_1 and prep_2 are two well-thought preprocessing functions discarded from this introduction (cf. Sect. 4).

Our Contribution

We re-analyze the short combiner of Mittelbach, and show the existence of an adversary that generates collisions for $C_{\mathrm{mit}}^{H_1,H_2}$ in 2 queries and second preimages in 1 query. The adversary is in line with Mittelbach's ideal combiner model, where H_1 is a random oracle $\mathcal{R}$ and H_2 is pre-defined by the adversary. The crux of the attack lies in the observation that the two preprocessing functions may not be injective, depending on the adversarial choice of H_2, an oversight in the proof.

We additionally present a solution to fix Mittelbach's combiner, which requires a more balanced usage of the keys in each of the preprocessing functions. We also prove that this fix does the job, i.e., restores the claimed security bounds up to a constant factor.

Outline

The remainder of the paper is organized as follows. We introduce some preliminaries in Sect. 2. The ideal combiner model as outlined by Mittelbach is summarized in Sect. 3. Section 4 describes Mittelbach's short combiner $C_{\mathrm{mit}}^{H_1,H_2}$ in detail. Our attacks on $C_{\mathrm{mit}}^{H_1,H_2}$ are given in Sect. 5 and we discuss how it can be fixed in Sect. 6.

2 Preliminaries

For $n \in \mathbb{N}$, we denote by $\{0,1\}^n$ the set of bit strings of size n. By $\{0,1\}^*$ we denote the set of bit strings of arbitrary length. For two bit strings x, y, their concatenation is denoted $x\|y$ and their bitwise exclusive or (xor) as $x \oplus y$ (for which x and y are presumed to be equally long). The size of x is denoted $|x|$. For $b \in \{1,2\}$, we denote by $\bar{b} = 3 - b \in \{2,1\}$. If $\mathcal{X}$ is a set, we denote by $x \xleftarrow{\$} \mathcal{X}$ the uniformly randomly sampling of an element from $\mathcal{X}$. If $\mathcal{X}$ is, on the other hand, a distribution, we use the same notation to say that x is chosen according to the distribution.

A hash function family is defined as $H : \{0,1\}^\kappa \times \{0,1\}^* \to \{0,1\}^n$ for $\kappa, n \in \mathbb{N}$, where for every $k \in \{0,1\}^\kappa$, H_k is a deterministic function that maps messages M of arbitrary length to digests of fixed length n. In security games, the key will conventionally be randomly drawn and disclosed at the beginning of the security experiment; it is simply used to select a function H_k randomly from the entire family of functions. The key input to H_k is left implicit if it is clear from the context. A random oracle on n bits is a function $\mathcal{R}$ which provides a random output of size n for each new query [2].

We model adversaries $\mathcal{A}$ as probabilistic algorithms with black-box access to zero or more oracles $\mathcal{O}_1, \ldots, \mathcal{O}_n$, written as $\mathcal{A}^{\mathcal{O}_1,\ldots,\mathcal{O}_n}$. We assume the adversary

always knows the security parameters (often the input and output sizes of the combiner and underlying hash functions) and refrain from explicitly mentioning these as input to $\mathcal{A}$. We consider computationally unbounded adversaries whose complexities are measured by the number of queries made to their oracles. We assume that the adversary never makes queries to which it knows the answer in advance.

If X is a random variable, the min-entropy of X is defined as

$$\mathrm{H}_\infty(X) = -\log(\max_x \mathbf{Pr}\left(X = x\right)).$$

Note that we can equivalently define $\mathrm{H}_\infty(X)$ in terms of a predictor $\mathcal{A}$ that aims to guess X, denoted $\mathrm{H}_\infty(X) = -\log(\max_{\mathcal{A}} \mathbf{Pr}\left(X = \mathcal{A}\right))$ [1]. Following [1,9,26], we define the (average) conditional min-entropy of X conditioned on random variable Z as

$$\widetilde{\mathrm{H}}_\infty\left(X \mid Z\right) = -\log(\max_{\mathcal{A}} \mathbf{Pr}\left(X = \mathcal{A}^Z\right)),$$

where $\mathcal{A}$ is a predictor that participates in random experiment Z. It has been demonstrated that $\widetilde{\mathrm{H}}_\infty\left(X \mid Z\right) \geq \mathrm{H}_\infty(X, Z) - b$, where Z may take 2^b values [9,30].

3 Ideal Combiner Model

A (k, l)-combiner for security property prop is a construction based on l hash functions, that achieves prop security as long as k out of l hash functions satisfy this property. Most combiners known in literature are $(1, 2)$-combiners, considering a construction C^{H_1,H_2} from two hash functions H_1, H_2. We focus on this type of combiners. A robust black-box combiner for security property prop is a combiner C^{H_1,H_2} for which an attack under prop can be reduced to an attack on H_1 and H_2. Various results on robustness of combiners have been presented [11–14,17,18]. Pietrzak [29] proved that the output length of a collision secure black-box combiner is at least the sum of the output lengths of H_1 and H_2 (minus a logarithmic term in the output size of H_1, H_2). A similar observation was recently made for second preimage and preimage resistance by Rjaško [31] and Mittelbach [25].

At ACNS 2013, Mittelbach elegantly lifted the security of combiners to the ideal model. That is, the hash functions underpinning C^{H_1,H_2} are based on a random oracle. The model discards the explicit need of a reduction, and combines well with the indifferentiability framework of Maurer et al. [23] and its application to hash functions by Coron et al. [5]. Nevertheless, this model, and particularly capturing the fact that one of the hash functions may be non-ideal, is not at all straightforward. We paraphrase the model in our own terminology.

The prop security of a combiner $C^{H_1,H_2} : \{0,1\}^\kappa \times \{0,1\}^* \to \{0,1\}^n$ based on two hash functions $H_1, H_2 : \{0,1\}^{\kappa_h} \times \{0,1\}^* \to \{0,1\}^n$ is captured as follows (the model generalizes straightforwardly to other domains and ranges). Let $\mathcal{R}$ be

a random oracle and $k \xleftarrow{\$} \{0,1\}^\kappa$. Consider a two-stage adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ with unbounded computational power. $\mathcal{A}_1$ gets no input and outputs $b \in \{1,2\}$ and a description of an efficient stateless function $H^{\mathcal{R}} : \{0,1\}^{\kappa_h} \times \{0,1\}^* \to \{0,1\}^n$ which may make calls to $\mathcal{R}$. Then, $\mathcal{A}_2$, with oracle access to $(\mathcal{R}, H^{\mathcal{R}})$ and knowledge of the key k, aims to break security property prop for $C^{H^{\mathcal{R}},\mathcal{R}}$ (if $b = 1$) or $C^{\mathcal{R},H^{\mathcal{R}}}$ (if $b = 2$). Formally, the advantage of $\mathcal{A}$ is defined as follows:

$$\mathbf{Adv}_C^{\mathsf{prop}}(\mathcal{A}) = \mathbf{Pr}_{\mathcal{R},k}\left(\begin{array}{ll}(b, H^{\mathcal{R}}, st) \xleftarrow{\$} \mathcal{A}_1\,, & : \ \mathcal{A}_2^{H_1,H_2}(k, st) \text{ breaks} \\ (H_b, H_{\bar{b}}) \leftarrow (H^{\mathcal{R}}, \mathcal{R}) & \mathsf{prop} \text{ for } C^{H_1,H_2}\end{array}\right),$$

where the randomness is taken over the choice of random oracle $\mathcal{R}$, random key $k \in \{0,1\}^\kappa$, and coins of $\mathcal{A}$.

The formal descriptions of the security advantages slightly differ for various types of security properties. In general, for collision, preimage, and second preimage resistance the definitions show resemblances with, but are more complex than, the formalization of Rogaway and Shrimpton [32]. For collision security of C^{H_1,H_2}, the advantage of $\mathcal{A}$ is defined as

$$\mathbf{Adv}_C^{\mathsf{coll}}(\mathcal{A}) = \mathbf{Pr}_{\mathcal{R},k}\left(\begin{array}{ll}(b, H^{\mathcal{R}}, st) \xleftarrow{\$} \mathcal{A}_1\,, & M \neq M' \wedge \\ (H_b, H_{\bar{b}}) \leftarrow (H^{\mathcal{R}}, \mathcal{R})\,, & : \ C^{H_1,H_2}(M) = \\ (M, M') \xleftarrow{\$} \mathcal{A}_2^{H_1,H_2}(k, st) & C^{H_1,H_2}(M')\end{array}\right).$$

For (second) preimage resistance, we focus on everywhere (second) preimage resistance. In everywhere preimage resistance, $\mathcal{A}_1$ selects an image $Y \in \{0,1\}^n$ at the start of the experiment. In everywhere second preimage resistance, $\mathcal{A}_1$ selects a first preimage $M \in \{0,1\}^\lambda$ at the start of the experiment, for some $\lambda < \infty$. The advantages of $\mathcal{A}$ are as follows:

$$\mathbf{Adv}_C^{\mathsf{epre}}(\mathcal{A}) = \mathbf{Pr}_{\mathcal{R},k}\left(\begin{array}{ll}(b, H^{\mathcal{R}}, Y, st) \xleftarrow{\$} \mathcal{A}_1\,, & \\ (H_b, H_{\bar{b}}) \leftarrow (H^{\mathcal{R}}, \mathcal{R})\,, & : \ C^{H_1,H_2}(M) = Y \\ M \xleftarrow{\$} \mathcal{A}_2^{H_1,H_2}(k, st) & \end{array}\right),$$

$$\mathbf{Adv}_C^{\mathsf{esec}[\lambda]}(\mathcal{A}) = \mathbf{Pr}_{\mathcal{R},k}\left(\begin{array}{ll}(b, H^{\mathcal{R}}, M, st) \xleftarrow{\$} \mathcal{A}_1\,, & M \neq M' \wedge \\ (H_b, H_{\bar{b}}) \leftarrow (H^{\mathcal{R}}, \mathcal{R})\,, & : \ C^{H_1,H_2}(M) = \\ M' \xleftarrow{\$} \mathcal{A}_2^{H_1,H_2}(k, st) & C^{H_1,H_2}(M')\end{array}\right).$$

The notion of everywhere second preimage resistance is also known as target collision resistance [16] and implies conventional second preimage resistance where M is randomly drawn. (We note that Mittelbach [26] considered target collision resistance and conventional second preimage resistance separately. Additionally, we slightly simplified the notion of preimage resistance, considering the case $\mathcal{A}_1$ selects the image rather than a set $\mathcal{X}$ from which the first preimage is secretly and randomly drawn.)

4 Mittelbach's Combiner

We consider Cryptophia's combiner $C_{\mathrm{mit}}^{H_1,H_2} : \{0,1\}^{\kappa} \times \{0,1\}^* \to \{0,1\}^n$ from Mittelbach [26], where $\kappa = 6n$. Let $k_i \in \{0,1\}^n$ for $i = 1,\ldots,6$ be independently chosen keys, and write $k = (k_1,\ldots,k_6)$. Let $H_1, H_2 : \{0,1\}^* \to \{0,1\}^n$ be two hash functions. The combiner is given by

$$C_{\mathrm{mit}}^{H_1,H_2}(k,M) = H_1\left(\tilde{m}_1^1 \| \cdots \| \tilde{m}_\ell^1\right) \oplus H_2\left(\tilde{m}_1^2 \| \cdots \| \tilde{m}_\ell^2\right),$$

where the message $M \in \{0,1\}^*$ is first injectively padded into n-bit message blocks $m_1 \| \cdots \| m_\ell = M \| \mathsf{pad}(M)$ using some padding function pad, which are subsequently preprocessed as

$$\begin{aligned} \tilde{m}_j^1 &= H_1(1 \parallel m_j \oplus k_1) \oplus m_j \oplus k_2 \oplus H_2(1 \parallel m_j \oplus k_3)\,, \\ \tilde{m}_j^2 &= H_2(0 \parallel m_j \oplus k_4) \oplus m_j \oplus k_5 \oplus H_1(0 \parallel m_j \oplus k_6)\,, \end{aligned} \tag{1}$$

for $j = 1,\ldots,\ell$. We remark that we swapped k_1 with k_3 and k_4 with k_6 compared to the original specification [26].

5 Attack

In the security model we recaptured in Sect. 3, Mittelbach proved that $C_{\mathrm{mit}}^{H_1,H_2}$ achieves collision security up to $2^{(n+1)/2}$ queries and preimage and second preimage security up to 2^n queries.[1] In the next proposition, we show that the collision result is incorrect. After the result, we also explain why the attack directly implies a second preimage attack. The work of [26] as well as its full version do not state any properties of the padding function $\mathsf{pad}(M)$. We assume a 10^*-padding concatenated with length strengthening. For simplicity and without loss of generality, we assume that $|\mathsf{pad}(M)| \leq n$, which is the case if the message length is encoded with at most $n-1$ bits.

Proposition 1. *There exists an adversary $\mathcal{A}$ making 2 queries, such that $\mathbf{Adv}_{C_{\mathrm{mit}}}^{\mathsf{coll}}(\mathcal{A}) = 1$.*

Proof. Let $\mathcal{R}$ be a random oracle and $k_1,\ldots,k_6 \xleftarrow{\$} \{0,1\}^n$. We focus on an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ that finds a collision for $C_{\mathrm{mit}}^{\mathcal{R},H^{\mathcal{R}}}$, where $H^{\mathcal{R}}$ is the hash function defined by $\mathcal{A}_1$. Our adversary proceeds as follows. $\mathcal{A}_1$ outputs $b = 2$ and the following hash function $H^{\mathcal{R}}$:

$$H^{\mathcal{R}}(x) = \begin{cases} \mathcal{R}(x) \oplus y, \text{ if } x = 1 \| y \text{ for some } y \in \{0,1\}^n, \\ 0, \text{ otherwise.} \end{cases}$$

[1] The formal preimage result is slightly different, claiming security up to $2^{\mathrm{H}_\infty(\mathcal{X})}$ queries, where the first preimage is secretly and randomly drawn from an adversarially chosen set $\mathcal{X}$.

This simplifies the combiner to $C_{\mathrm{mit}}^{\mathcal{R},H^{\mathcal{R}}}(k,M) = \mathcal{R}\Big(\tilde{m}_1^1 \| \cdots \| \tilde{m}_\ell^1\Big)$, where

$$\tilde{m}_j^1 = \mathcal{R}(1 \parallel m_j \oplus k_1) \oplus k_2 \oplus k_3 \oplus \mathcal{R}(1 \parallel m_j \oplus k_3)\,,$$

for $j = 1, \ldots, \ell$. Next, the adversary $\mathcal{A}_2$ gets as input $(k_1, \ldots, k_6)$ and outputs colliding pair M and $M' = M \oplus k_1 \oplus k_3$ for some $M \in \{0,1\}^n$.

We proceed with showing that the colliding pair is valid. As $|M| = |M'| = n$, the messages are padded as $m_1 \| m_2 = M \| \mathsf{pad}(M)$ and $m_1' \| m_2' = M' \| \mathsf{pad}(M')$, where $m_1 = M$, $m_1' = M'$, and $m_2 = m_2'$. The latter implies $\tilde{m}_2^1 = \tilde{m}_2'^1$. The preprocessed $\tilde{m}_1^1$ and $\tilde{m}_1'^1$ satisfy

$$\begin{aligned}\tilde{m}_1^1 &= \mathcal{R}(1 \parallel M \oplus k_1) \oplus k_2 \oplus k_3 \oplus \mathcal{R}(1 \parallel M \oplus k_3) \\ &= \mathcal{R}(1 \parallel M \oplus k_3) \oplus k_2 \oplus k_3 \oplus \mathcal{R}(1 \parallel M \oplus k_1) = \tilde{m}_1'^1\,.\end{aligned}$$

Concluding, $\tilde{m}_1^1 \| \tilde{m}_2^1 = \tilde{m}_1'^1 \| \tilde{m}_2'^1$ and thus

$$C_{\mathrm{mit}}^{\mathcal{R},H^{\mathcal{R}}}(k,M) = \mathcal{R}\Big(\tilde{m}_1^1 \| \tilde{m}_2^1\Big) = \mathcal{R}\Big(\tilde{m}_1'^1 \| \tilde{m}_2'^1\Big) = C_{\mathrm{mit}}^{\mathcal{R},H^{\mathcal{R}}}(k,M')\,. \qquad \square$$

Proposition 2. *Let $\lambda < \infty$. There exists an adversary $\mathcal{A}$ making 1 query, such that $\mathbf{Adv}_{C_{\mathrm{mit}}}^{\mathsf{esec}[\lambda]}(\mathcal{A}) = 1$.*

Proof. In the attack of Prop. 1 the choice of M is independent of $(k_1, \ldots, k_6)$. Therefore, the attack also works if M is chosen by $\mathcal{A}_1$ at the beginning of the game. $\square$

The flaw in the security analysis of [26] lies in the fact that it only considers distributions of $\tilde{m}_j^c$ computed from $m_j, k_{3c-2}, k_{3c-1}, k_{3c}$ via (1) for $c \in \{1,2\}$, but never joint distributions of $\tilde{m}_j^c, \tilde{m}_j'^c$ given two messages m, m'. In more detail, Prop. 4.5 of the full version of [26] inadvertently assumes that $\tilde{m}^c$ and $\tilde{m}'^c$ are mutually distinct whenever $m \neq m'$. The preimage bound derived in [26] is nevertheless correct, and so are the analyses of $C_{\mathrm{mit}}^{H_1,H_2}$ as a pseudorandom function and MAC.

6 Fix

To fix Mittelbach's combiner $C_{\mathrm{mit}}^{H_1,H_2}$, we suggest to use an additional set of keys $l_1, l_2 \in \{0,1\}^n$ as separate input to H_1, H_2 in the preprocessing functions of (1). Consequently, we can leave out $m_j \oplus k_2$ and $m_j \oplus k_5$ from these functions as they have become redundant, and we can simply set $(k_4, k_6) = (k_1, k_2)$. (For the original $C_{\mathrm{mit}}^{H_1,H_2}$ these keys k_2, k_4, k_5, k_6 are necessary to guarantee preimage resistance.)

More formally, we suggest combiner $C^{H_1,H_2} : \{0,1\}^\kappa \times \{0,1\}^* \to \{0,1\}^n$, where $\kappa = 4n$. Let $k_1, k_2, l_1, l_2 \in \{0,1\}^n$ be independently chosen keys, and write $kl = (k_1, k_2, l_1, l_2)$. Let $H_1, H_2 : \{0,1\}^* \to \{0,1\}^n$ be two hash functions. The combiner is given by

$$C^{H_1,H_2}(kl, M) = H_1\Big(\tilde{m}_1^1 \| \cdots \| \tilde{m}_\ell^1\Big) \oplus H_2\Big(\tilde{m}_1^2 \| \cdots \| \tilde{m}_\ell^2\Big)\,,$$

where the message $M \in \{0,1\}^*$ is first injectively padded into n-bit message blocks $m_1 \| \cdots \| m_\ell = M \| \mathsf{pad}(M)$ using some padding function pad, which are subsequently preprocessed as

$$\begin{aligned} \tilde{m}_j^1 &= H_1(0 \parallel l_1 \parallel m_j \oplus k_1) \oplus H_2(0 \parallel l_2 \parallel m_j \oplus k_2)\,, \\ \tilde{m}_j^2 &= H_1(1 \parallel l_1 \parallel m_j \oplus k_1) \oplus H_2(1 \parallel l_2 \parallel m_j \oplus k_2)\,, \end{aligned} \tag{2}$$

for $j = 1, \ldots, \ell$.

This fix, indeed, guarantees that $\tilde{m}^c$ and $\tilde{m}'^c$ are mutually different whenever m, m' are, except with small probability. In the remainder of this section, we will prove that C^{H_1,H_2} indeed achieves the originally claimed security bounds for collision, preimage, and second preimage resistance up to an inevitable constant factor. For a proof on the robustness for pseudorandomness and MAC security we refer to [26].

Before we proceed, we remark explicitly that we require $H^{\mathcal{R}}$ to be a stateless hash function. In the artificial case in which $H^{\mathcal{R}}$ is allowed to hold state, C^{H_1,H_2} is insecure. An attack is given in App. A.

Security Proofs

For $c \in \{1,2\}$ we define preprocessing function $\tilde{m}^c(kl, m)$ on input of $kl = (k_1, k_2, l_1, l_2) \in \{0,1\}^{4n}$ and $m \in \{0,1\}^n$ as

$$\tilde{m}^c(kl, m) = H_1(c-1 \parallel l_1 \parallel m \oplus k_1) \oplus H_2(c-1 \parallel l_2 \parallel m \oplus k_2)\,.$$

These preprocessing functions correspond to the two equations of (2) for $c = 1, 2$. The remainder of the proof is as follows. In Lem. 1 we compute the (conditioned) min-entropies of the values $\tilde{m}^c$. This lemma is a direct generalization of Lems. 1 and 2 of [26]. Then, preimage security is proven in Thm. 1, collision security in Thm. 2, and second preimage security in Thm. 3.

Lemma 1. *Let $\mathcal{R}$ be an n-bit random oracle and let $kl \xleftarrow{\$} \{0,1\}^{4n}$. Let $H^{\mathcal{R}}$ be a hash function with access to $\mathcal{R}$ (but not using kl). Then, for all $c \in \{1,2\}$ and distinct $m, m' \in \{0,1\}^n$,*

$$\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^c(kl,m) \mid kl, m\right) \geq n - \log(q_H)\,, \tag{3}$$

$$\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^c(kl,m) \mid \tilde{m}^c(kl,m'), kl, m, m'\right) \geq n - 2\log(q_H)\,, \tag{4}$$

$$\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^c(kl,m) \mid \tilde{m}^{\bar{c}}(kl,m'), kl, m, m'\right) \geq n - 2\log(q_H)\,, \tag{5}$$

$$\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^c(kl,m) \mid \tilde{m}^{\bar{c}}(kl,m), kl, m\right) \geq n - 2\log(q_H)\,, \tag{6}$$

where q_H is the number of calls to $\mathcal{R}$ in one evaluation to $H^{\mathcal{R}}$. (We note that conditioning in (5-6) is done on $\tilde{m}^{\bar{c}}$, as opposed to $\tilde{m}^c$ in (4).)

Proof. The combiner C is symmetric, and without loss of generality we assume $b = 2$, hence $(H_1, H_2) = (\mathcal{R}, H^{\mathcal{R}})$, where $\mathcal{R}$ is an n-bit random oracle and $H^{\mathcal{R}}$ is defined by adversary $\mathcal{A}_1$. Also, $c = 1$ without loss of generality.

The min-entropy of (3) reads

$$\begin{aligned}
&\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^1(kl,m) \mid kl,m\right)\\
&= \widetilde{\mathrm{H}}_\infty\left(\mathcal{R}(0 \parallel l_1 \parallel m \oplus k_1) \oplus H^{\mathcal{R}}(0 \parallel l_2 \parallel m \oplus k_2) \;\middle|\; kl,m\right)\\
&= \widetilde{\mathrm{H}}_\infty\left(\mathcal{R}(0 \parallel l_1 \parallel \hat{m} \oplus \hat{k}_1) \oplus H^{\mathcal{R}}(0 \parallel l_2 \parallel \hat{m}) \;\middle|\; \hat{k}_1,l_1,l_2,\hat{m}\right),
\end{aligned}$$

where the second step is by substitution of $(\hat{k}_1,\hat{m}) = (k_1 \oplus k_2, m \oplus k_2)$ and by leaving out the redundant k_2 in the condition. Note that the evaluation of $\mathcal{R}$ is independent of the evaluation of $H^{\mathcal{R}}$, *unless* $H^{\mathcal{R}}(0 \parallel l_2 \parallel \hat{m})$ evaluates $\mathcal{R}(0 \parallel l_1 \parallel \hat{m} \oplus \hat{k}_1)$. Here, we recall that $\hat{k}_1, l_1, l_2$ are mutually independently and randomly drawn, but $\hat{m}$ is chosen by $\mathcal{A}_2$ and may depend on $(\hat{k}_1,l_1,l_2)$. The hash function $H^{\mathcal{R}}$ chosen by $\mathcal{A}_1$ makes q_H evaluations of $\mathcal{R}$, which can decrease the entropy by at most $\log(q_H)$ bits in any experiment. Thus, we find:

$$\begin{aligned}
&\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^1(kl,m) \mid kl,m\right)\\
&\geq \widetilde{\mathrm{H}}_\infty\left(l_1, \hat{m}(\hat{k}_1,l_1,l_2) \oplus \hat{k}_1 \;\middle|\; l_2, \hat{m}(\hat{k}_1,l_1,l_2)\right) - \log(q_H)\\
&\geq n - \log(q_H)\,.
\end{aligned}$$

We proceed with the min-entropy of (4):

$$\begin{aligned}
&\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^1(kl,m) \mid \tilde{m}^1(kl,m'),kl,m,m'\right)\\
&= \widetilde{\mathrm{H}}_\infty\left(\begin{array}{l}\mathcal{R}(0 \parallel l_1 \parallel m \oplus k_1) \oplus H^{\mathcal{R}}(0 \parallel l_2 \parallel m \oplus k_2) \;\big|\\ \mathcal{R}(0 \parallel l_1 \parallel m' \oplus k_1) \oplus H^{\mathcal{R}}(0 \parallel l_2 \parallel m' \oplus k_2), kl,m,m'\end{array}\right)\\
&\geq \widetilde{\mathrm{H}}_\infty\left(\begin{array}{l}\mathcal{R}(0 \parallel l_1 \parallel \hat{m} \oplus \hat{k}_1) \oplus H^{\mathcal{R}}(0 \parallel l_2 \parallel \hat{m}),\\ \mathcal{R}(0 \parallel l_1 \parallel \hat{m}' \oplus \hat{k}_1) \oplus H^{\mathcal{R}}(0 \parallel l_2 \parallel \hat{m}')\end{array} \;\middle|\; \hat{k}_1,l_1,l_2,\hat{m},\hat{m}'\right) - n\,, \qquad (7)
\end{aligned}$$

where we substituted $(\hat{k}_1,\hat{m},\hat{m}') = (k_1 \oplus k_2, m \oplus k_2, m' \oplus k_2)$ and left out redundant k_2. Here, we recall that $\hat{m} \neq \hat{m}'$, but both message blocks may depend on $\hat{k}_1,l_1,l_2$. Before proceeding, we pause to see what happens if we were considering the original combiner $C^{\mathcal{R},H^{\mathcal{R}}}_{\mathrm{mit}}$ of Sect. 4. In this case, l_1 and l_2 are absent. The entropy term in (7) then equals at most n if $\hat{m}' = \hat{m} \oplus \hat{k}_1$ (in case $H^{\mathcal{R}} = \mathcal{R}$), leading to a lower bound ≥ 0. Note that the attack of Sect. 5 takes the message blocks this way.

Returning to (7), as $\hat{k}_1,l_1,l_2$ are independently and randomly drawn and $\hat{m} \neq \hat{m}'$, the two terms in the min-entropy are independent, both achieve a min-entropy of at least $n - \log(q_H)$ (by (3)), and hence

$$\widetilde{\mathrm{H}}_\infty\left(\tilde{m}^1(kl,m) \mid \tilde{m}^1(kl,m'),kl,m,m'\right) \geq 2(n - \log(q_H)) - n \geq n - 2\log(q_H)\,.$$

The same reasoning applies to the min-entropies of (5) and (6), where for the latter we particularly use that the two evaluations of $\mathcal{R}$ are mutually independent due to the domain separation 1/0. □

Theorem 1. *For any adversary $\mathcal{A}$, where $\mathcal{A}_2$ makes $q_{\mathcal{A}}$ queries and where every evaluation of $H^{\mathcal{R}}$ makes at most q_H calls to $\mathcal{R}$, we have* $\mathbf{Adv}^{\mathsf{epre}}_{C}(\mathcal{A}) \leq (q_H^3 + 1)q_{\mathcal{A}}/2^n$.

Proof. Let $(H_b, H_{\bar{b}}) = (H^{\mathcal{R}}, \mathcal{R})$, where $\mathcal{R}$ is an n-bit random oracle and b and $H^{\mathcal{R}}$ are defined by adversary $\mathcal{A}_1$. Let Y be the target image. Consider an evaluation $C^{H_1,H_2}(kl, M)$, where M has not been evaluated so far. The evaluation constitutes a preimage if

$$C^{H_1,H_2}(kl, M) = \mathcal{R}(U^{\bar{b}}(M)) \oplus H^{\mathcal{R}}(U^b(M)) = Y\,, \tag{8}$$

for some random distributions $U^{\bar{b}}, U^b$ corresponding to (2). If this happens, at least one of the following two events occurred:

$$\begin{aligned}
&\mathsf{E}_1 :\ H^{\mathcal{R}}(U^b(M)) \text{ evaluates } \mathcal{R}(U^{\bar{b}}(M))\,,\\
&\mathsf{E}_2 :\ \neg\mathsf{E}_1 \ \wedge\ (8)\,.
\end{aligned}$$

By Lem. 1 equation (6) (or in fact a slight variation to ℓ blocks, which gives the same lower bound), $U^{\bar{b}}(M)$ given $U^b(M)$ has min-entropy at least $n - 2\log(q_H)$. In other words, any call to $\mathcal{R}$ by $H^{\mathcal{R}}$ evaluates $U^{\bar{b}}(M)$ with probability at most $2^{-(n-2\log(q_H))} = q_H^2/2^n$. As $H^{\mathcal{R}}$ makes q_H evaluations, E_1 happens with probability at most $q_H^3/2^n$. Regarding E_2, by $\neg\mathsf{E}_1$ the call to $\mathcal{R}$ is independent of $H^{\mathcal{R}}(U^b(M))$ and (8) holds with probability $1/2^n$.

As $\mathcal{A}$ has $q_{\mathcal{A}}$ attempts, it finds a preimage with probability at most $(q_H^3 + 1)q_{\mathcal{A}}/2^n$. □

Theorem 2. *For any adversary $\mathcal{A}$, where $\mathcal{A}_2$ makes $q_{\mathcal{A}}$ queries and where every evaluation of $H^{\mathcal{R}}$ makes at most q_H calls to $\mathcal{R}$, we have* $\mathbf{Adv}^{\mathsf{coll}}_{C}(\mathcal{A}) \leq (3q_H^3 + 1)q_{\mathcal{A}}^2/2^{n+1}$.

Proof. Let $(H_b, H_{\bar{b}}) = (H^{\mathcal{R}}, \mathcal{R})$, where $\mathcal{R}$ is an n-bit random oracle and b and $H^{\mathcal{R}}$ are defined by adversary $\mathcal{A}_1$. Consider two evaluations C^{H_1,H_2} of two distinct M, M'. The two evaluations constitute a collision if

$$\mathcal{R}(U^{\bar{b}}(M)) \oplus \mathcal{R}(U^{\bar{b}}(M')) = H^{\mathcal{R}}(U^b(M)) \oplus H^{\mathcal{R}}(U^b(M'))\,, \tag{9}$$

for some random distributions $U^{\bar{b}}, U^b$ corresponding to (2). If this happens, at least one of the following four events occurred:

$$\begin{aligned}
&\mathsf{E}_1 :\ U^{\bar{b}}(M) = U^{\bar{b}}(M')\,,\\
&\mathsf{E}_2 :\ H^{\mathcal{R}}(U^b(M)) \text{ evaluates } \mathcal{R}(U^{\bar{b}}(M))\,,\\
&\mathsf{E}_3 :\ H^{\mathcal{R}}(U^b(M')) \text{ evaluates } \mathcal{R}(U^{\bar{b}}(M))\,,\\
&\mathsf{E}_4 :\ \neg(\mathsf{E}_1 \vee \mathsf{E}_2 \vee \mathsf{E}_3) \ \wedge\ (9)\,.
\end{aligned}$$

By Lem. 1 equation (4), E_1 holds with probability at most $q_H^2/2^n$. Similar to the proof of Thm. 1, E_2 and E_3 happen with probability at most $q_H^3/2^n$ (by Lem. 1

equations (5) and (6)). Regarding E_4, by $\neg(\mathsf{E}_1 \vee \mathsf{E}_2 \vee \mathsf{E}_3)$ the call to $\mathcal{R}(U^{\bar{b}}(M))$ is independent of the other terms and (9) holds with probability $1/2^n$.

As $\mathcal{A}$ has $q_{\mathcal{A}}$ attempts, it finds a collision with probability at most $(2q_H^3 + q_H^2 + 1)\binom{q_{\mathcal{A}}}{2}/2^n \leq (3q_H^3 + 1)q_{\mathcal{A}}^2/2^{n+1}$. □

Theorem 3. *For any adversary $\mathcal{A}$, where $\mathcal{A}_2$ makes $q_{\mathcal{A}}$ queries and where every evaluation of $H^{\mathcal{R}}$ makes at most q_H calls to $\mathcal{R}$, we have $\mathbf{Adv}_C^{\mathsf{esec}[\lambda]}(\mathcal{A}) \leq (3q_H^3 + 1)q_{\mathcal{A}}/2^n$.*

Proof. The proof follows from the proof of Thm. 2 with the difference that the first message M is fixed in advance. □

Remark 1. We remark that the terms q_H^3 in fact also appear in the bounds of Mittelbach [26], though accidentally dropped out. (Lem. 2 of [26] considers $q = \max\{q_H, q_{\mathcal{A}}\}$, while Prop. 2 treats q as being $q_{\mathcal{A}}$.) That said, as $H^{\mathcal{R}}$ should be an efficient hash function, it is fair to assume that it makes a limited amount of evaluations of $\mathcal{R}$. Particularly, if $q_H = \mathcal{O}(1)$, we retain the original security bounds.

Remark 2. The results hold with the same bounds if the messages were padded into n'-bit message blocks for $n' < n$, and if $k_1, k_2 \in \{0,1\}^{n'}$. Changing the size of l_1, l_2 would, on the other hand, directly affect the security bounds.

Acknowledgments. This work was supported in part by the Research Fund KU Leuven, OT/13/071, and in part by the Research Council KU Leuven: GOA TENSE (GOA/11/007). Bart Mennink is a Postdoctoral Fellow of the Research Foundation – Flanders (FWO). The authors would like to sincerely thank the anonymous reviewers of CANS 2014, as well as Atul Luykx, Gregory Maxwell, and Arno Mittelbach, for their comments and suggestions.

References

1. Alwen, J., Dodis, Y., Wichs, D.: Leakage-resilient public-key cryptography in the bounded-retrieval model. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 36–54. Springer, Heidelberg (2009)
2. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: ACM Conference on Computer and Communications Security, pp. 62–73. ACM, New York (1993)
3. Boneh, D., Boyen, X.: On the impossibility of efficiently combining collision resistant hash functions. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 570–583. Springer, Heidelberg (2006)
4. Canetti, R., Rivest, R., Sudan, M., Trevisan, L., Vadhan, S.P., Wee, H.M.: Amplifying collision resistance: A complexity-theoretic treatment. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 264–283. Springer, Heidelberg (2007)
5. Coron, J.-S., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård revisited: How to construct a hash function. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 430–448. Springer, Heidelberg (2005)

6. Dierks, T., Allen, C.: The TLS protocol version 1.0. Request for Comments (RFC) 2246 (January 1999), http://tools.ietf.org/html/rfc2246
7. Dierks, T., Rescorla, E.: The transport layer security (TLS) protocol version 1.1. Request for Comments (RFC) 4346 (April 2006), http://tools.ietf.org/html/rfc4346
8. Dierks, T., Rescorla, E.: The transport layer security (TLS) protocol version 1.2. Request for Comments (RFC) 5246 (August 2008), http://tools.ietf.org/html/rfc5246
9. Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. SIAM Journal of Computing 38(1), 97–139 (2008)
10. Fischlin, M., Lehmann, A.: Security-amplifying combiners for collision-resistant hash functions. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 224–243. Springer, Heidelberg (2007)
11. Fischlin, M., Lehmann, A.: Multi-property preserving combiners for hash functions. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 375–392. Springer, Heidelberg (2008)
12. Fischlin, M., Lehmann, A., Pietrzak, K.: Robust multi-property combiners for hash functions revisited. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part II. LNCS, vol. 5126, pp. 655–666. Springer, Heidelberg (2008)
13. Fischlin, M., Lehmann, A., Pietrzak, K.: Robust multi-property combiners for hash functions. Journal of Cryptology 27(3), 397–428 (2014)
14. Fischlin, M., Lehmann, A., Wagner, D.: Hash function combiners in TLS and SSL. In: Pieprzyk, J. (ed.) CT-RSA 2010. LNCS, vol. 5985, pp. 268–283. Springer, Heidelberg (2010)
15. Freier, A., Karlton, P., Kocher, P.: The secure sockets layer (SSL) protocol version 3.0. Request for Comments (RFC) 6101 (August 2011), http://tools.ietf.org/html/rfc6101
16. Halevi, S., Krawczyk, H.: Strengthening digital signatures via randomized hashing. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 41–59. Springer, Heidelberg (2006)
17. Harnik, D., Kilian, J., Naor, M., Reingold, O., Rosen, A.: On robust combiners for oblivious transfer and other primitives. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 96–113. Springer, Heidelberg (2005)
18. Herzberg, A.: On tolerant cryptographic constructions. In: Menezes, A. (ed.) CT-RSA 2005. LNCS, vol. 3376, pp. 172–190. Springer, Heidelberg (2005)
19. Hoch, J.J., Shamir, A.: Breaking the ICE - finding multicollisions in iterated concatenated and expanded (ICE) hash functions. In: Robshaw, M. (ed.) FSE 2006. LNCS, vol. 4047, pp. 179–194. Springer, Heidelberg (2006)
20. Hoch, J., Shamir, A.: On the strength of the concatenated hash combiner when all the hash functions are weak. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part II. LNCS, vol. 5126, pp. 616–630. Springer, Heidelberg (2008)
21. Joux, A.: Multicollisions in iterated hash functions. application to cascaded constructions. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 306–316. Springer, Heidelberg (2004)
22. Lehmann, A.: On the Security of Hash Function Combiners. Ph.D. thesis, Technischen Universität Darmstadt, Darmstadt (2010)

23. Maurer, U.M., Renner, R., Holenstein, C.: Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 21–39. Springer, Heidelberg (2004)
24. Mendel, F., Rechberger, C., Schläffer, M.: MD5 is weaker than weak: Attacks on concatenated combiners. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 144–161. Springer, Heidelberg (2009)
25. Mittelbach, A.: Hash combiners for second pre-image resistance, target collision resistance and pre-image resistance have long output. In: Visconti, I., De Prisco, R. (eds.) SCN 2012. LNCS, vol. 7485, pp. 522–539. Springer, Heidelberg (2012)
26. Mittelbach, A.: Cryptophia's short combiner for collision-resistant hash functions. In: Jacobson, M., Locasto, M., Mohassel, P., Safavi-Naini, R. (eds.) ACNS 2013. LNCS, vol. 7954, pp. 136–153. Springer, Heidelberg (2013), Full version: Cryptology ePrint Archive, Report 2013/210
27. Nandi, M., Stinson, D.: Multicollision attacks on generalized hash functions. Cryptology ePrint Archive, Report 2004/330 (2004)
28. Pietrzak, K.: Non-trivial black-box combiners for collision-resistant hash-functions don't exist. In: Naor, M. (ed.) EUROCRYPT 2007. LNCS, vol. 4515, pp. 23–33. Springer, Heidelberg (2007)
29. Pietrzak, K.: Compression from collisions, or why CRHF combiners have a long output. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 413–432. Springer, Heidelberg (2008)
30. Reyzin, L.: Some notions of entropy for cryptography - (invited talk). In: Fehr, S. (ed.) ICITS 2011. LNCS, vol. 6673, pp. 138–142. Springer, Heidelberg (2011)
31. Rjaško, M.: On existence of robust combiners for cryptographic hash functions. In: Conference on Theory and Practice of Information Technologies - ITAT 2009. CEUR Workshop Proceedings, vol. 584, pp. 71–76 (2009)
32. Rogaway, P., Shrimpton, T.: Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 371–388. Springer, Heidelberg (2004)

A Breaking the Fix with Stateful $H^{\mathcal{R}}$

We present an attack on C^{H_1,H_2} of Sect. 6 in the artificial case that $H^{\mathcal{R}}$ is allowed to maintain state. We note that this attack does not invalidate the security proofs of Sect. 6, and it is solely presented for theoretical interest. In more detail, in the next proposition we show how to extend the attack of Prop. 1. The attack is more advanced, as $\mathcal{A}_2$ (who knows the l_i's) needs to pass those on to $H^{\mathcal{R}}$ (which does not know these). Note that $H^{\mathcal{R}}$ is, indeed, defined by $\mathcal{A}_1$ without a priori knowledge of the keys, but we assume $H^{\mathcal{R}}$ can hold state.

Proposition 3. *There exists an adversary $\mathcal{A}$ making 3 queries, such that* $\mathbf{Adv}^{\mathsf{coll}}_{C}(\mathcal{A}) = 1$.

Proof. Let $\mathcal{R}$ be a random oracle and $k_1, k_2, l_1, l_2 \xleftarrow{\$} \{0,1\}^n$. We focus on an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ that finds a collision for $C^{\mathcal{R},H^{\mathcal{R}}}$, where $H^{\mathcal{R}}$ is the hash

function defined by $\mathcal{A}_1$. Our adversary proceeds as follows. $\mathcal{A}_1$ outputs $b = 2$ and the following hash function $H^{\mathcal{R}}$. The function simply outputs $H^{\mathcal{R}}(x) = \mathcal{R}(x)$ until and including *the first time* it gets evaluated on $H^{\mathcal{R}}(x)$ for $x = 1\|y\|z$ for some $y, z \in \{0,1\}^n$. At this point, define $(l_2^\star, l_1^\star) = (y, z)$, and respond all *subsequent* queries as follows:

$$H^{\mathcal{R}}(x) = \begin{cases} \mathcal{R}(1\|l_1^\star\|z), \text{ if } x = 1\|l_2^\star\|z \text{ for some } z \in \{0,1\}^n, \\ 0, \text{ otherwise.} \end{cases} \tag{10}$$

Next, $\mathcal{A}_2$ gets as input (k_1, k_2, l_1, l_2). The first query $\mathcal{A}_2$ makes is $M = l_1$, which gets padded to $l_1\|\mathsf{pad}(l_1)$. Note that in this evaluation of $C^{\mathcal{R},H^{\mathcal{R}}}$, the first query to $H^{\mathcal{R}}$ is on input of $1\|l_2\|l_1$. The adversarial hash function is programmed in such a way that it defines $l_2^\star = l_2$ and $l_1^\star = l_1$. The adversary $\mathcal{A}_2$ ignores the outcome of the combiner evaluation.

For the remaining evaluations $H^{\mathcal{R}}$ operates as (10), and we can simplify the combiner to $C^{\mathcal{R},H^{\mathcal{R}}}(kl, M) = \mathcal{R}\Big(\tilde{m}_1^1\| \cdots \|\tilde{m}_\ell^1\Big)$, where

$$\tilde{m}_j^1 = \mathcal{R}(1 \parallel l_1 \parallel m_j \oplus k_1) \oplus \mathcal{R}(1 \parallel l_1^\star \parallel m_j \oplus k_2)\,,$$

for $j = 1, \ldots, \ell$. Next, the adversary $\mathcal{A}_2$ outputs colliding pair M and $M' = M \oplus k_1 \oplus k_2$ for some $M \in \{0,1\}^n$. The remainder of the proof follows Prop. 1, using $l_1^\star = l_1$. □

The second preimage attack of Prop. 2 generalizes similarly. A technicality occurs in the above attack as we assume the $H^{\mathcal{R}}$'s are evaluated in a sequential order. In other words, the attack may fail if $H^{\mathcal{R}}$ gets first evaluated for message block $\mathsf{pad}(l_1)$. A way to address this is to create a buffer. I.e., to make the first combiner evaluation on a concatenation of α l_1's, hence $M = l_1\| \cdots \|l_1$, and program $H^{\mathcal{R}}$ to define $l_1^\star$ as soon as it is "seen" α times.

We remark that the attacks suggest that there does not exist any combiner that achieves security against adversaries with state-maintaining $H^{\mathcal{R}}$: $\mathcal{A}_2$ can always pass on the secret keys to $H^{\mathcal{R}}$, be it in more complicated and elaborated ways than described in Prop. 3.