Secure Message Authentication against Related-Key Attack

Rishiraj Bhattacharyya[1] and Arnab Roy[2]

[1] ENS de Lyon/INRIA, France. rishiraj.bhattacharyya@ens-lyon.fr
[2] SnT, Université du Luxembourg, Luxembourg. arnab.roy@uni.lu

Abstract. Security against related-key attacks is an important criteria for modern cryptographic constructions. In the related-key setting, the adversary has the ability to query the underlying function on the target key as well as on some related-keys. Although provable security against related-key attack has received considerable attention in recent years, most of the results in the literature aim to achieve pseudorandomness and semantic security and often lead to inefficient constructions.

In this paper, we formalize the notion of unpredictability in the related-key setting. We start with the definitions of related-key security of Message Authentication Codes and identify required properties of related-key derivation functions for provable security. We show that unlike PRFs, MACs can inherently tolerate related-key attacks against constant transformations. Next, we consider the construction of variable-input-length MACs from fixed-input-length related-key unpredictable functions. We present simple attacks against XCBC and TMAC. We present a general construction of related-key secure MACs. Our construction, instantiated with Enciphered CBC construction of Dodis, Pietrzak and Puniya (EUROCRYPT 2008), results into first provably secure domain extension of related-key secure unpredictable functions. Finally, we present two constructions of related-key secure MACs from DDH assumption. The first construction is extremely efficient and tolerates group-induced partial key transformations. The second construction achieves security against independent group-induced tranformations and is more efficient than the RK-PRFs achieved by Bellare and Cash (CRYPTO 2010).

Keywords: Message Authentication, Related-Key Attack, Domain Extension

1 Introduction

A series of cryptanalytic results have established the threat of related-key attacks as a mainstream cryptographic challenge. Introduced by Biham and Knudsen [17, 7] for block ciphers, related-key cryptanalysis has led to high profile attacks, ranging from key recovery [8] to distinguishers [9–11]. In a related-key setting, the secret key of a cryptosystem/primitive can be partially controlled by the adversary. Specifically, the adversary can apply key transformations to change the key and observe the outcome under the modified keys. A typical example of such transformation is fault injection attack.

Motivated by the cryptanalytic applications, Bellare and Kohno [6] initiated a theoretical study of related-key (RK) security of block ciphers, traditionally modelled as pseudorandom permutations (PRPs) and pseudorandom functions (PRFs). They defined related-key security with respect to a class of related-key-deriving (RKD) functions, Φ, which specifies the relations available to the adversary, and considered an adversary who can (adaptively) choose the relation from Φ during the attack. Although in some of the examples of [6], choice of RKD set makes the adversary quite powerful, they help to characterize the set of functions.

Despite of its importance in applied cryptography only a few positive results are known in the RK setting [2, 4, 6, 18]. Bellare and Kohno [6], followed by Lucks [18] considered the construction of RK secure pseudorandom functions and permutations from the ideal primitives like ideal cipher. Lucks introduced the notion of group induced RKD class where, if the keyspace forms a group under some given operation, then the RKD functions may be chosen by an adversary using this group-operation. An obvious example of such operation is bit wise exclusive or (XOR) operation of a key with some known constant (of same bit length as the key). In a breakthrough result, Bellare and Cash [3] constructed RK secure PRPs based on hardness of DDH/DLIN assumptions. Although this construction proves an important feasibility result , the solution is quite inefficient and hard to use in practice.

On the other hand, related-key distinguishers have been found for widely used block-ciphers including AES [11]. Naturally, concerns are mounting over the security of the primitives, designed based on these ciphers [19]. Specifically, security of applications like message authentication codes, where block-ciphers are used heavily as the underlying primitive, needs to be revisited in light of the related-key attacks. Although, most of the popular MAC constructions were proven to be pseudorandom assuming pseudorandomness of the underlying block cipher, much weaker security notion, like unpredictability, is sufficient for MACs. As AES and some other block-ciphers are believed to remain unpredictable, even against related-key attacks, a natural question is what security guarantee we can prove from this assumption. Specifically, *Can we achieve an efficient construction of Message Authentication Code, secure against related-key attacks, if we only assume related-key unpredictability from the underlying block ciphers ?*

Our Results. In this paper, we focus our attention to the security of message authentication codes against related-key attacks. Instead of modeling the block cipher as RK-PRP, we model underlying block cipher as only RK unpredictable. We reconsider several practical and popular constructions from the literature, and analyze them in the light of related-key attacks, towards their feasibility as related-key MACs. We also present two proofs of concept RK unpredictable functions, both based on DDH assumption. A more detailed description of our results follows.

Definitions We start with presenting general definition of unpredictability against related-key attacks. We consider two types of security of unforgeability. In the first type (called Weak Related-Key Unforgeability), adversary's prediction has to be on a fresh message, i.e. she can not predict the output of the function (on the target key) on a message, which she has queried earlier even on a related-key. In the stronger type (called Related-Key Unforgeability), adversary can be more powerful. She is allowed to forge a message, even if she has queried it on a related-key (although, not on the target key).

Handling constant functions We revisit the necessary conditions for the class of related-key transformations argued by Bellare and Kohno[5], specifically transformations mapping all keys to some constant. We present a simple proof that a general message authentication code is inherently secure against constant RKD functions. To the best of our knowledge, this results to a first symmetric key construction which can handle constant RKD transformations.

Cryptanalysis of popular MAC construction Next, we show negative results on many popular constructions. We show simple attacks against XCBC and TMAC. We also prove that, if the key of the MAC construction is viewed as a single key, ECBC and FCBC constructions do not guarantee unforgeability, irrespective of the strength of underlying block ciphers.

A Related-Key Secure Domain Extension The natural question that arises from the results of previous paragraph is whether any existing construction preserves unpredictability against a related-key adversary. For the general setting, most designs use the NI construction of [1]. The general idea behind the construction is a collision at the output would imply a collision at the compression function (by standard MD argument). Then one would try to design an efficient weak collision resistant compression function from unpredictable functions, and prove that a collision at the compression function output can be used to predict the output of the underlying functions. However, in the related-key scenario, this need not be the case. Indeed, the collision of the mode as well as the compression function may be with a related-key query. If the related key query was made later, then the previous approach will not work.

To solve this problem, we propose a Merkle-Damgård based construction (prefix-free NI) for related-key unpredictability. Specifically, our construction is a prefix free MD domain extension with an extra round at the end. Using this extra round, we prove that even if the collision is with a related-key query, input of the last round (during the evaluation of forgery output) is either new (hence can be used for prediction) or generates a collision with a previous query on the target key. Then one can extend the standard MD based arguments to find forgery on the underlying functions.

We instantiate this mode of operation by the enciphered CBC construction of Dodis, Pietrzak and Puniya [14], and prove that this gives a variable input length related-key unpredictable function from fixed input length related-key secure unpredictable functions and permutations.

A general construction of RK Unpredictable Functions. Our final contribution is a provably secure construction of Related Key Unpredictable function in the standard model. We instantiate this construction by two recent constructions in [12]. Our basic construction, secure against partial key-transformations, is much efficient in terms of keysize. Specifically, the keysize in our case is linear as compared to quadratic keysize in [3]. Our second construction is fully secure against component-wise group induced transformations. The construction of Bellare and Cash [3] can be seen as a special case of our construction. Additionally, the concept of key homomorphism in this work avoids the complexity of key malleability faced in [3]. Compared to Bellare-Cash construction, this construction is efficient in terms of exponentiation.

2 Overview of our technique

Claw-free RKD sets. In this work, like most of the previous positive results, we focus on claw-free related-key deriving (RKD) functions. Roughly speaking, a set Φ of RKD functions is called claw-free if for all but negiligible fraction of k, distinct functions ϕ_1 and ϕ_2 from Φ, $\phi_1(k) \neq \phi_2(k)$. We note that, Bellare, Cash, and Miller [4] have constructed related-key secure signature scheme where they could break this requirement. However, their construction heavily depends on the notion of ICR pseudorandom generator, which in turn depends on RK-secure pseudorandom functions. We stress that, no construction of RK-secure pseudorandom function against non-claw free RKD set is known till date, and constructions of [4] are not instantiable by current RK-secure PRFs. In such a situation, we consider the claw-free RKD sets as worthy target.

Handling Multiple Keys The most popular paradigm to design variable-input-length (VIL) MAC (or PRF) is the Hash then MAC (or Hash then PRF) approach. The message is first hashed by applying a collision resistant hash function, and then passed through an independent fixed input length MAC (PRF). Naturally, the key of such a construction contains the key(s) of the hash function (or the underlying primitive) and an independently sampled key of the final transformation. The key of the variable input length MAC is simply the concatenation of these sampled keys. The question is, how will the adversary change this key, i.e. should she consider functions which work independently over the individual keys? Or we can allow her to consider any claw-free RKD transformation over the keyspace (Cartesian product of the keyspace of the hash function and the final transformation) of the variable input length MAC.

In Section 6, we show that if we allow any claw-free RKD transformation over the keyspace, then multi-key constructions have an inherent limitation. Specifically, we show attacks on ECBC and FCBC, where the related-key adversary can turn a three key construction into a two key construction, using a claw-free RKD class.

We identify an alternative yet natural class of RKD functions, called component-wise transformation as a feasible target. A component-wise transformation over the keyspace $\mathcal{K}^n$ is an n-element vector of RKD functions over $\mathcal{K}$. Let $\phi = (\phi_1, \phi_2, \cdots, \phi_n)$ be such a vector where each ϕ_i is a function over $\mathcal{K}$. For any key $k = (k_1, k_2, \cdots, k_n)$, $\phi(k)$ is defined by $(\phi_1(k_1), \phi_2(k_2), \cdots, \phi_n(k_n))$. We remark that this idea of component-induced transformation is not new. In fact, constructions of RK-PRFs [3] were shown essentially for such classes. However, we are the first to formalize such idea.

Removing Unkeyed Collision Resistance Assumption One of the most important tools of the related-key secure VIL-PRF of [3] is an unkeyed collision resistant hash function with carefully chosen range. Thus, security of this PRF is based on the assumption of existence of unkeyed collision resistant hash function. This assumption is very strong (in fact, stronger than existence of one-way function) and thus undesirable. However, the problem is, if we consider keyed hash function, then that key is also subject to related key attack. It is not clear from [3], how to tackle that problem.

We solve this problem by introducing the notion of identity collision resistance and target preimage resistance for keyed hash functions. Intuitively, against an identity collision resistant hash function H with key k, a related-key adversary (which makes adaptive queries serially) will not be able to output, with significant probability, a message m such that $H_k(m)$ matches with the output of the (related-key) queries she already made. We prove such a notion along with a notion of target preimage resistance (lifted to the RK setting) is enough for the Hash and MAC construction. We also show how to construct such an hash function from length preserving related-key secure MACs/permutations. Although we faced some technical challenges (mentioned in the previous section), we solve them with an elegant prefix-free padding and Merkle-Damgård mode of operation.

Independent work

Independent to our work, Xagawa [20] also considered related key security of message authentication code extending the results of [13]. Some of his results are similar to our algebraic constructions in Section 10.

3 Notations and Security Definitions

Notations: If x is a string, $|x|$ denotes the length (number of characters) of the string, $x[i]$ denotes the i^{th} character of x, and $x_1||x_2||..||x_t$ denotes concatenation of t strings. For a finite set X , $|X|$ denotes the size of the set. $x \leftarrow_{\mathrm{R}} X$ means selecting an element x uniformly at random from the set X. $A \rightarrow x$ denotes that an algorithm A outputs x. $\mathsf{Func}(\mathcal{D}, \mathcal{R})$ denotes the set of all functions from $\mathcal{D}$ to $\mathcal{R}$. A family of functions $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ takes a key $k \in \mathcal{K}$ and an input $m \in \mathcal{D}$, and outputs $F(k, m)$. Throughout the paper F_k denotes the function $F(k, .)$. A block-cipher is a family of permutations $E : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{D}$ and E_k denotes the permutation $E(k, .)$ for $k \in \mathcal{K}$.

Unforgeability of a function family: The security of F as a MAC is expressed via the following security game, where $\mathcal{A}$ is an adversary with oracle access to F_k,

Game `UF-CMA`

- Setup: $k \leftarrow_{\mathrm{R}} \mathcal{K}$.
- Query Phase: $\mathcal{A}$ makes a set of queries $\mathcal{Q}$ to the oracle F_k

- $\textsf{Guess Phase}$: $\mathcal{A} \rightarrow (m, \sigma)$.
- $\textsf{Verify}$: If $m \notin \mathcal{Q}$ and $F_k(m) = \sigma$ then $\mathcal{A}$ wins, else $\mathcal{A}$ looses.

A family of function F is said to be (q, ℓ, ϵ) *unforgeable under chosen message attack* if for all adversary $\mathcal{A}$ who makes q queries with total size of the queries ℓ bits,

$$\mathbf{Adv}_F^{mac}(\mathcal{A}) \stackrel{def}{=} Prob[\mathcal{A} \text{ wins game } \texttt{UF-CMA}] \leq \epsilon.$$

We note that the notion of unforgeability is also known as the unpredictability.

Framework for Related-Key Attack In the related-key setting, security of a function family $F(\mathcal{K}, \mathcal{D}, \mathcal{R})$ is defined against a *related-key adversary*. At the beginning of the corresponding security game the adversary outputs a set of functions $\varPhi \subseteq \textsf{Func}(\mathcal{K}, \mathcal{K})$, called related-key deriving (RKD) functions. Throughout the game, the adversary has access to a *related-key oracle* $F_{\texttt{RK}}$. The oracle takes an ordered pair (m, ϕ) as input $(m \in \mathcal{D}, \phi \in \varPhi)$ and returns $F(\phi(k), m)$, where $F_k \in F(\mathcal{K}, \mathcal{D}, \mathcal{R})$ for some $k (\leftarrow_{\text{R}} \mathcal{K})$ unknown to the adversary.

If $\varPhi$ contains the identity function $\textsf{id}$ then $F_{\texttt{RK}}$ can also simulate the oracle $F(k, .)$. For the rest of the paper unless specified we will assume that $\varPhi$ includes the function $\textsf{id}$.

In [18] Lucks described an elegant way of choosing $\varPhi$ as a set of *group-induced transformations* when $(\mathcal{K}, *)$ is a group.

Definition 1. *(***Group Induced Transformations***[18]) Let $\mathcal{K}$ be a group under operation $\circ$. A group induced transformation is a set of functions, $\varPhi$, over $\mathcal{K}$ defined as*

$$\varPhi \stackrel{def}{=} \{\phi : \mathcal{K} \rightarrow \mathcal{K} | \exists \delta \in \mathcal{K} : \phi(k) = k \circ \delta\}$$

Another important family of RKD functions, called *partial transformations*, is also used in [6, 18]. Partial transformations restrict the adversary to choose a function which can change only a part of the entire key. For example if we have a family of functions with key space $\mathcal{K} \times \mathcal{K}$, then a partial key transformation ϕ' can be defined as $\phi'(k_1, k_2) = (k_1, \phi(k_2))$ where ϕ is an RKD function on $\mathcal{K}$.

Finally, we introduce the notion of component-induced key transformations for multiple-key constructions.

Definition 2. *(***Component-wise Transformations***) Let $\mathcal{K} = \mathcal{K}_1 \times \mathcal{K}_2 \times \cdots \mathcal{K}_n$ be a set of keys. A component-wise transformation is a set of functions $\varPhi$ over $\mathcal{K}$ defined as*

$$\begin{aligned}\varPhi \stackrel{def}{=} \{\phi = (\phi_1, \phi_2, \cdots, \phi_n) | \forall i, \phi_i : \mathcal{K}_i \rightarrow \mathcal{K}_i, \forall k = (k_1, k_2, \cdots, k_n) \in \mathcal{K} \\ \phi(k) = (\phi_1(k_1), \phi_2(k_2), \cdots, \phi_n(k_n))\}\end{aligned}$$

We stress that, in case of component-wise transformations each ϕ_i is applied on k_i and is independent to other k_js.

4 Unforgeability against Related-Key Attack

We start with a formal definition of the related-key security for MACs. Recall that, in the related-key setup, the adversary may query the oracle on a message and a related-key. The obvious way (analogous to [16], in the context of signature) to define the notion of related-key-unforgeability would be to ensure that the forgery m^* was never queried to the oracle with the relation id. However, the adversary may define the RKD function to be such that it agrees with id for all but negligable fraction of the keys. For such a function, the security gets broken trivially. In other words, such a restriction would force the RKD class to be claw-free. We present a general definition of related-key unforgeability through the following game between an adversary and the challenger. The adversary $\mathcal{A}$ has oracle access to $F_{\mathtt{RK}}$.

Game `RK-UF-CMA`

- Setup: $k \leftarrow_{\mathrm{R}} \mathcal{K}$, $\mathcal{A}$ gets the security parameter λ. $\mathcal{A}$ submits the description of the RKD class Φ. $\mathcal{Q} = \emptyset$.
- Query: $\mathcal{A}$ adaptively queries with (m, ϕ), the challenger returns $F(\phi(k), m)$. $\mathcal{Q} = \mathcal{Q} \cup (m, \phi)$.
- Guess: $\mathcal{A}$ outputs a forgery (m^*, σ^*).
- Verify: If $F(k, m^*) = \sigma^*$, and $\phi(k) \neq k$ for all $(m^*, \phi) \in \mathcal{Q}$ then $\mathcal{A}$ wins else $\mathcal{A}$ looses.

Definition 3. ***(Related-Key Unforgeability)*** *A family of functions F is said to be* (q, ℓ, ϵ) unforgeable under chosen message related-key attack *over the RKD set Φ if for all adversary $\mathcal{A}$ who makes q queries with total size of the queries ℓ bits,*

$$\boldsymbol{Adv}_F^{rk-mac}(\mathcal{A}, \Phi) \overset{def}{=} Prob[\mathcal{A} \text{ wins game } \texttt{RK-UF-CMA} \text{ with RKD set } \Phi] \leq \epsilon$$

where the probability is taken over the key k and the internal randomness of $\mathcal{A}$.

5 Properties of RKD Transformations

In this section, we analyze the necessary properties of Φ, the RKD transformation, necessary for related-key security of MAC. In [5], Bellare Kohno proposed two essential conditions, namely unpredictability and claw-free ness, for RKD functions for related-key security. Specifically, they proved that if Φ contains a constant function, then no block cipher can be pseudorandom against related-key attack over Φ. In a sharp contrast, we now prove that, a general message authentication code is inherently *secure* against constant RKD functions.

Theorem 1. *Let $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be a MAC. Let $\Phi \overset{def}{=} \{\phi_c : c \in \mathcal{K}, \forall k \in \mathcal{K}, \phi_c(k) = c\}$ be the set of constant RKD transformations. For all related-key* *adversary $\mathcal{A}_{RK}$ against related-key unforgeability of F over RKD set Φ, there exists adversary $\mathcal{A}$ such that*

$$\boldsymbol{Adv}_F^{rk-mac}(\mathcal{A}_{RK}, \Phi) \leq \boldsymbol{Adv}_F^{mac}(\mathcal{A})$$

Proof. The main idea of the proof is the following: the adversary $\mathcal{A}$ will simulate $\mathcal{A}_{RK}$. When $\mathcal{A}_{RK}$ queries with id, $\mathcal{A}$ will answer the queries by making query to its own oracle. However as the related-key functions are constant functions, $\mathcal{A}$ can answer any related-key query (m, ϕ_c) by computing $F(c, m)$ on its own[3]. Finally when $\mathcal{A}_{RK}$ outputs a forgery (m^*, σ^*), $\mathcal{A}$ outputs (m^*, σ^*). By the condition of the game `RK-UF-CMA`, (m^*, id) was never queried by $\mathcal{A}_{RK}$. Hence (m^*, id) was never queried by $\mathcal{A}$ as well. So, $\mathcal{A}$ succeeds whenever $\mathcal{A}_{RK}$ succeeds.

Insecurity against colliding functions The claw-freeness condition, however, is essential for security of related-key security of MAC. The attack of [5], involving addition and xor over the keyspace, can indeed recover the secret key, resulting a forgery. For detailed description of this attack, we refer the reader to Proposition 4.3 of [5].

6 Related-Key Attacks against popular MAC constructions

In this section we show examples of some simple related-key adversaries against some well known MAC constructions. We consider two popular variants of CBC-MAC, namely XCBC and TMAC. Constructions like ECBC and FCBC can also be attacked with a more aggressive class of transformations. Due to space constraint, the cryptanalysis of ECBC and FCBC are omitted in this proceedings version. All these constructions were proved to be secure under the assumption that underlying block cipher is PRP. Although our ultimate aim is to achieve a related-key secure MAC when the underlying primitive is related-key unforgeable, in the following examples we show that the XCBC and TMAC can be forged using related-key attack even if the underlying block ciphers are *related-key secure prp.*

Proposition 2 *XCBC is not related-key secure.*

Proof. The attack is extremely simple. Let n be the block length of the underlying block cipher. Consider a message $m = m_1 || m_2$ such that $|m_1| = |m_2| = n$. Let the RKD set chosen by adversary be $\mathcal{A}_{RK}$. $\Phi = \{\phi_i(k_1, k_2, k_2) = (k_1, k_2 \oplus i, k_3) : 0 < i < 2^{|k_2|}\} \cup \mathsf{id}$. $\mathcal{A}_{RK}$ makes a related-key query (m, ϕ_i) for any $i > 0$. Suppose σ be the answer. $\mathcal{A}_{RK}$ returns (m^*, σ) , where $m^* = m_1 || m_2 \oplus i$.

let $y = E_{k_1}(m_1)$. Then the last block operation is $E_{k_1}(y \oplus m_2 \oplus k_2)$. We know that $E_{k_1}(y \oplus m_2 \oplus (k_2 \oplus i)) = E_{k_1}(y \oplus (m_2 \oplus i) \oplus k_2)$. Hence $XCBC_{\mathtt{RK}}(m, \phi_i) = XCBC(m^*) = \sigma$. This implies (m^*, σ) is a valid forgery and $\mathbf{Adv}_{XCBC}^{rk-mac}(\mathcal{A}_{XCBC}, \Phi) = 1$.

[3] Note that, obvious description of ϕ_c leaks the constant c.

TMAC can be viewed as a variant of XCBC MAC and instead of using three keys it uses two keys in the construction. The last block operation of TMAC is given as $E_{k_1}(m' \oplus (k_2 \cdot u))$, where u is a constant polynomial in $GF(2^n)$ and the product is performed in the same field. The simplification of the product $x \cdot u$ is linear in x. Hence using a RKD set similar as above the adversary will be able to forge TMAC.

Corollary 1. *TMAC is not a secure MAC against related-key attack.*

Prewhitening key and RKA: Both the attacks described above exploit the use of prewhitening key. Suppose a MAC construction involves an operation of the form $E_{k'}(k * x)$ (where x is a chaining value independent of k and $*$ is a commutative-group induced operation) and k is independent of k' and other keys used in the construction. Then it is always possible to mount similar related-key attack as above.

7 Technical Tools

In this section we introduce the tools we use in our construction. First we introduce the notion of weak unforgeability against related-key attack, which essentially bridges the notion of unforgeability between the standard and the related-key settings.

Weak Unforgeability against Related-Key Attack

Definition 4. *(Key-Homomorphic MAC) Let $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be family of MACs. We say that F is key-homomorphic MAC if $\mathcal{K}$ and $\mathcal{R}$ are groups with efficient operations ($\circ$ and $*$ respectively) and for any fixed $m \in \mathcal{D}$, there is a group homomorphism form $\mathcal{K}$ to $\mathcal{R}$. Specifically, for any $k_1, k_2 \in \mathcal{K}$,*

$$F_{k_1 \circ k_2}(m) = F(k_1, m) * F(k_2, m)$$

Let F be a family of key-homomorphic MACs and Φ° a $(\mathcal{K}, \circ)$ group-induced RKD set. Essentially, for $\phi \in \Phi^\circ$, one can compute $F(\phi(k), m)$ by making queries to $F(k, m)$ and using the group homomorphism property of F. In the `RK-UF-CMA` game, the adversary is challenged to forge $F(k, .)$. Apparently, finding $F(\phi(k), .)$ from $F(k, .)$ does not directly help her. However, the adversary may first query the related-key oracle and get $F(\phi(k), m)$ for some m, then using the group homomorphism property, predict the value of $F(k, m)$. To see this, consider an adversary $\mathcal{A}$ who makes a query (m, ϕ) to $F_{\mathtt{RK}}$ for some $m \in \mathcal{D}$. Now, we know that $\phi(k) = k \circ \delta$ for $\delta \in \mathcal{K}$. So, $\mathcal{A}$ knows $\sigma_1 = F(\phi(k), m)$ and can compute $\sigma_2 = F(\delta, m)$ on her own as the family F is public. Hence, $\mathcal{A}$ successfully forges $F(k, .)$ with (m, σ) where $\sigma = \sigma_1 * \sigma_2^{-1}$.

We observe that, previous adversary $\mathcal{A}$ is not a unique-message adversary. Against a unique-message adversary of the `RK-UF-CMA` game, a key-homomorphic

MAC is related-key unforgeable over group induced Φ. Motivated by this observation, we introduce the notion of weak unforgeability against related-key attack. In this case, the adversary is not allowed to forge a message which she has queried even on some non-id RKD function.

Game `WeakRK-UF-CMA`

- Setup: $k \leftarrow_{\mathrm{R}} K$, $\mathcal{A}$ gets the security parameter λ. $\mathcal{A}$ submits the description of the RKD class Φ. $\mathcal{Q} = \emptyset$.
- Query: $\mathcal{A}$ adaptively queries with (m, ϕ), the challenger returns $F(\phi(k), m)$. $\mathcal{Q} = \mathcal{Q} \cup (m, \phi)$.
- Guess: $\mathcal{A}$ outputs a forgery (m^*, σ^*).
- Verify: If $F(k, m^*) = \sigma^*$, and $(m^*, \phi) \notin \mathcal{Q}$ for any ϕ then $\mathcal{A}$ wins else $\mathcal{A}$ looses.

Definition 5 (Weak RK-Unforgeability). *A family of functions F is said to be (q, ℓ, ϵ) weakly unforgeable under chosen message related-key attack (WRK-UF) over the RKD set Φ if for all adversary $\mathcal{A}$ who makes q queries with total size of the queries ℓ bits,*

$$\boldsymbol{Adv}_F^{wrk-mac}(\mathcal{A}, \Phi) \overset{def}{=} Prob[\mathcal{A} \textit{ wins game } \texttt{WeakRK-UF-CMA} \textit{ with RKD set } \Phi] \leq \epsilon$$

where the probability is taken over the key k and the internal randomness of $\mathcal{A}$.

For a key homomorphic MAC the following lemma can be proved in a straightforward way.

Lemma 1. *(***Key Homomorphic MAC is WRK-UF***) Let $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ be a family of key-homomorphic MACs. Let Φ be a claw-free set of group induced RKD functions. $\mathcal{F}$ is a secure WRK-UF over Φ. Specifically, for every (q, ℓ) adversary $\mathcal{A}$, there exists a (q, ℓ) adversary $\mathcal{A}_F$ such that*

$$\boldsymbol{Adv}_F^{wrk-mac}(\mathcal{A}, \Phi) \leq \boldsymbol{Adv}_F^{mac}(\mathcal{A})$$

Identity Fingerprint The main technical tool used in [3] in order to construct the RK secure PRF is the notion of *key fingerprint.* Informally, a key fingerprint (as defined in [3]) is a vector over the message space, such that under two different keys, outputs of the function will be different on at least one index. However, as observed in [4], this notion is too demanding and may not be achievable for some PRFs.

In this paper, we consider the following relaxed notion of key fingerprint.

Definition 6. *(***Identity Fingerprint***) Let $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ be family of functions and Φ be a set of RKD functions over $\mathcal{K}$. Let w be a d dimensional vector over $\mathcal{D}$. We call w an* identity-fingerprint *of F over Φ if*

$$Prob_{k \leftarrow_R \mathcal{K}} \Bigg[\forall \phi \in \Phi : \Big(F(k, w_1), F(k, w_2), \cdots, F(k, w_d) \Big) \neq \Big(F(\phi(k), w_1), F(\phi(k), w_2), \cdots, F(\phi(k), w_d) \Big) \Bigg] > 1 - \texttt{negl}$$

where $d = \mathcal{O}(|k|)$, $\texttt{negl}$ is some negligible function in terms of $|k|$.

We remark that, the identity key fingerprint notion of [4] is similar. As argued in [4], few distinct points from the domain can be considered as a candidate identity fingerprint for any practical block-cipher. Although we cannot prove it formally, such an assumption seems to be consistent with the premise of cryptanalysis.

ICTPR hash function In this paper we remove the collision resistant hash function assumption. In our framework, we encounter keyed hash function which is subject to tampering by the adversary. To achieve security even in such a scenario, we propose and use the notion of ICTPR hash functions.

An ICTPR hash function $H : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ has two properties: identity-collision (IC) resistance and target preimage (TP) resistance
IDENTITY COLLISION RESISTANCE. Roughly, the identity collision resistance ensures that, for (related-key) adversary with oracle access to $H_{\mathtt{RK}}$, output of a query on a message m and the secret key (i.e. query of the form (m, id)), does not collide with the output of some previous query (even on a related-key). The formal security game works in the following way.

Game ID-CR

- Setup: $k \leftarrow_{\mathrm{R}} \mathcal{K}$, $\mathcal{A}$ gets the security parameter λ. $\mathcal{A}$ submits the description of the RKD class Φ. $\mathcal{Q} = \emptyset$.
- Query: $\mathcal{A}$ adaptively queries with (m, ϕ), the challenger returns $H(\phi(k), m)$. $\mathcal{Q} = \mathcal{Q} \cup (m, \phi)$.
- Collision: $\mathcal{A}$ outputs a message m^*.
- Verify: If for some $(m, \phi) \in \mathcal{Q}$, $H(\phi(k), m) = H(k, m^*)$ and $(m^*, \mathsf{id}) \notin \mathcal{Q}$ then $\mathcal{A}$ wins else $\mathcal{A}$ looses.

Definition 7. *(***Identity Collision Resistant Hash Function***) Let $H : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ be family of hash functions and Φ be a set of RKD functions on $\mathcal{K}$. H is said to be (q, ℓ, ϵ)* identity collision resistant *(ICR) over the RKD set Φ if for all adversary $\mathcal{A}$ who makes q queries with total size of the queries ℓ bits,*

$$\boldsymbol{Adv}_H^{icr}(\mathcal{A}, \Phi) \overset{def}{=} Prob[\mathcal{A} \text{ wins game } \texttt{ID-CR} \text{ with RKD set } \Phi] \leq \epsilon$$

where the probability is taken over the key k and the internal randomness of $\mathcal{A}$.

TARGET PREIMAGE RESISTANCE AGAINST RELATED-KEY ATTACK. In addition to the identity collision resistance, we also need a notion of everywhere preimage resistance against related-key attacks. The preimage resistance game between an adversary $\mathcal{A}$ and a challenger for a hash function $H : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$ is described as following

Game RK-TPR

- Setup: $k \leftarrow_{\mathrm{R}} \mathcal{K}$, $\mathcal{A}$ gets the security parameter λ. $\mathcal{A}$ submits t targets $z_1, \cdots, z_t \in \mathcal{R}$, and the description of the RKD class Φ. $\mathcal{Q} = \emptyset$.
- Query: $\mathcal{A}$ adaptively queries with (m, ϕ), the challenger returns $H(\phi(k), m)$. $\mathcal{Q} = \mathcal{Q} \cup (m, \phi)$.
- Preimage: $\mathcal{A}$ outputs a message m^*.
- Verify: If $H(k, m^*) = z_i$, for some i then $\mathcal{A}$ wins else $\mathcal{A}$ looses.

Definition 8. *(***Related-Key Target Preimage Resistant Hash Function***) Let $H : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be family of hash functions and Φ be a set of RKD functions on $\mathcal{K}$. H is said to be (q, t, ℓ, ϵ)* related-key target preimage resistant *(RK-TPR) over the RKD set Φ if for all adversary $\mathcal{A}$ who submits t targets, makes q queries with total size of the queries ℓ bits,*

$$\boldsymbol{Adv}_H^{rk-tpr}(\mathcal{A}, \Phi) \overset{def}{=} Prob[\mathcal{A} \text{ wins game } \texttt{RK-TPR} \text{ with RKD set } \Phi] \leq \epsilon$$

where the probability is taken over the key k and the internal randomness of $\mathcal{A}$.

We define ICTPR advantage of an adversary $\mathcal{A}$ against a hash function H as

$$\mathbf{Adv}_H^{ictpr} = \mathbf{Adv}_H^{rk-tpr} + \mathbf{Adv}_H^{icr}$$

8 Construction of Related-Key secure MAC

In this section, we show a general construction of related-key secure MAC. The basic essence of our construction is essentially the Hash then MAC paradigm of An and Bellare [1], lifted to the related-key setting. In fact most of the proposed VIL-MAC constructions [15, 14] have been proved secure in this paradigm. The intuitive approach while extending the arguments of [1] would be to show that a suitable hash function H followed by a FIL-*related-key unforgeable* MAC F will give us a VIL-related-key secure MAC G. However, in the following theorem, we prove that, for claw-free RKD sets, if the hash function is ICTPR, it is enough for F only to be *weak related-key unforgeable* (cf. Definition 5).

Theorem 3. *Let $F : \mathcal{K}_1 \times \mathcal{D} \to \mathcal{R}$ be a weak related-key unforgeable MAC over RKD set Φ_1 with identity fingerprint $\overline{w} = (w_1, w_2, \cdots, w_d)$. Let $H : \mathcal{K}_2 \times \{0,1\}^* \to \mathcal{D}$ be a ICTPR hash function over the RKD set Φ_2. Let $G : (\mathcal{K}_1 \times \mathcal{K}_2) \times \{0,1\}^* \to \mathcal{R}$ be a family of function defined as*

$$G(k_1, k_2, m) \overset{def}{=} F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

where $k_1 \in \mathcal{K}_1, k_2 \in \mathcal{K}_2$. G is related-key unforgeable against chosen message attack over the component-induced RKD set $\Phi \overset{def}{=} \Phi_1 \times \Phi_2$. Specifically if there exists a (q, l) adversary $\mathcal{A}_G$ against G, then there exists a $(q, q \log |\mathcal{D}|)$ adversary $\mathcal{A}_F$ against F, and a (q, l) adversary $\mathcal{A}_H$ against H such that

$$\boldsymbol{Adv}_F^{wrk-mac}(\mathcal{A}_F, \Phi_1) + \boldsymbol{Adv}_H^{ictpr}(\mathcal{A}_H, \Phi_2) \geq \boldsymbol{Adv}_G^{rk-mac}(\mathcal{A}_G, \Phi)$$

Proof. Let $\tau_{\mathsf{id}} = F(k_1, w_1)\|F(k_1, w_2)\| \cdots \|F(k_1, w_d)$, and $\tau_{\phi_1} = F(\phi_1(k_1), w_1)\| F(\phi_1(k_1), w_2)\| \cdots \|F(\phi_1(k_1), w_d)$. The basic idea of the proof is the following. Let (m^*, σ) be a valid forgery. If $x^* = H(k_2, m^*\|\tau_{\mathsf{id}})$ does not collide with any previous H query (including the related-key oracles, thus maintaining identity collision resistance), or one of the w_is of the identity fingerprint $\overline{w}$ (thus maintaining target preimage resistance), then the query to $F(k, .)$ is new and was not queried even to the related-key oracle F_{RK}. Hence (x^*, σ) is a valid forgery against weak related-key unforgeable F_k. Hence we need to show that against any related-key adversary if x^* collides with the output of some previous H_{RK} query or $x^* \in \{w_1, \cdots, w_d\}$, ICTPR property of H_{k_2} can be broken. The arguments for those cases are straightforward. We refer the reader to the full version for the formal proof.

Up to this point, our approach closely matched with the approach of Bellare and Cash, who also used similar arguments. The difference comes in while constructing a ICTPR hash function. While [3] assumes an unkeyed collision resistant function with tailor-made range, we present a mode of operation based on fixed-input length related-key secure MAC (to construct VIL-related key unforgeable MAC) in the next section. We mention that given a keyed collision resistant hash function $H(k, .)$, one can easily get an ICTPR hash function (against claw-free transformations), $\hat{H}(k, .)$ defined as $\hat{H}(k, m) = k\|H(k, m)$. However, when constructing from block ciphers (as done in practice), this construction is trivially insecure (as it gives away the key). Additionally, to use it in Theorem 3, the final transformation requires to have a larger domain. On the other hand, our construction can be instantiated with a single related-key unpredictable function with independently sampled keys.

9 ICTPR from FIL-RKUF

In this section, we propose a mode of operation to construct a ICTPR hash function from length preserving related-key unforgeable MACs. Such a mode along with Theorem 3 will give us a variable input length MAC. We stress that **the proof works for any RKD set, i.e. if one starts with a fixed-input-length related-key unpredictable function, secure without the claw-free assumption on the RKD set, the resulting MAC remains secure without the claw-free assumption**.

We will describe the mode in two steps. First we shall describe a domain extension of fixed-input-length ICTPR compression function. Then we shall show that the enciphered CBC compression function of Dodis, Pietrzak, and Puniya [14] can be used to construct a fixed-input-length ICTPR compression function from length preserving related-key unforgeable MACs.

9.1 VIL-ICTPR Hash Function from ICTPR compression function

We shall use a variant of prefix free Merkle-Damgård iteration. Let $\mathcal{D} = \{0, 1\}^{2n}$, $\mathcal{R} = \{0, 1\}^n$, and $H' : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be a fixed-input-length ICTPR compression function.

Padding Rule Let m be input message. Let $len(m) = |m|$ be the length of the message. The message m is divided into blocks of $n-1$ bits. If $len(m)$ is not a multiple of $n-1$, the last block is padded with a bit 1 and sufficiently many 0s. After this padding let $m_1, m_2, \cdots, m_l$ be the blocks. The final padded message $\textsc{Pad}(m)$ will be the following

$$\textsc{Pad}(m) = y_1 \| y_2 \| \cdots \| y_l \| y,$$

where each $y_i = 0 \| m_i$, and $y = 1 \| len(m)$.

The Mode. Our mode is essentially the Merkle-Damgård mode with an extra round at the end with $1 \| 0^{n-1}$ as the message block. Formal algorithm of the iteration is the following

Algorithm 1 pseudo-code for the pfNI mode of operation

```
function $pfNI^{H'}(k, m)$
    $h_0 \leftarrow 0^n$
    $\textsc{Pad}(m) = y_1 || y_2 || \cdots || y_l || y$
    for $1 \le i \le l$ do
        $h_i \leftarrow H'(k, h_{i-1} || y_i)$
    $h_{l+1} \leftarrow H'(k, h_l || y)$
    $h \leftarrow H'(k, h_{l+1} || 1 || 0^{n-1})$
    return $h$
```

Security. Now we show that the $pfNI$ mode is ICTPR preserving. Let $H' : \mathcal{K} \times \{0,1\}^{2n} \to \{0,1\}^n$ be a compression function. We shall prove that, if there exists an adversary $\mathcal{A}_H$ against $H \overset{def}{=} pfNI^{H'}$ breaking the ICTPR property, then there is an adversary $\mathcal{A}_{H'}$ against the ICTPR property of H'. To show this, we need to show reductions for both identity collision resistance and target preimage resistance (cf. Section 7).

Simulation of H. $\mathcal{A}_{H'}$ has access to the oracle H'_{RK}. Simulation of oracle H_{RK} will be performed by querying H'_{RK}. During the simulation, $\mathcal{A}_{H'}$ maintains a list Q containing the queries to H'_{RK} and the corresponding responses.

Reduction for Identity Collision Resistance: Suppose $\mathcal{A}_H$ breaks the identity collision resistance of H. Recall that, identity collision resistance requires that no query (m^*, id) generates a collision with a previous (m, ϕ) (ϕ may or may not be id) query. Hence, $\mathcal{A}_H$ makes a (m^*, id) query to H such that $H(k, m^*) = H(\phi(k), m)$ and (m, ϕ) query was made before (m^*, id) query.

Let h_{ℓ^*+1} be the penultimate chaining value during the computation of $H(k, m^*)$. The following two cases can happen depending on whether h_{ℓ^*+1} was given as a response of some previous H'_{RK} query. Let $x = h_{\ell+1} \| 10^{n-1}$ be the last H' query during the computation of $H(\phi(k), m)$.

1. $h_{\ell^*+1} = IV$: If h_{ℓ^*+1} is equal to IV, then we can show a reduction breaking the target preimage resistance of H'. We analyze it in the reduction for target preimage resistance.

2. $H'(k, h_{\ell^*+1}\|\omega)$ **was not queried during the simulation for any** $\omega \in \{0,1\}^n$**:** The padding ensures that 10^{n-1} is the last message block of all the queries. Hence $h_{\ell^*+1} \neq h_{\ell+1}$. Moreover, $H'(k, h_{\ell^*+1}\|10^{n-1})$ has been queried after $H'(\phi(k), h_{\ell+1}\|10^{n-1})$.
   As $H(k, m^*) = H(\phi(k), m)$, obviously
   $$H'(k, h_{\ell^*+1}\|10^{n-1}) = H'(\phi(k), h_{\ell+1}\|10^{n-1}).$$
   This collision breaks the identity collision resistance property of H'.
3. $H'(k, h_{\ell^*+1}\|\omega)$ **was queried during the simulation for some** ω**:** If h_{ℓ^*+1} is not equal to IV, then h_{ℓ^*+1} matches with some chaining value during the simulation of the $pfNI$ mode on some previous (m', id) query. As $m^* \neq m'$, by standard argument of prefix free padding and collision resistance of Merkle-Damgård iteration, we will find a collision with some previous $H'(k, .)$ query.

Reduction for Target Preimage resistance: When $\mathcal{A}_H$ submits the set of "target images" $\{z_1, \cdots, z_t\}$, $\mathcal{A}_{H'}$ submits $T = \{IV, z_1, \cdots, z_t\}$. For each $H_{\mathsf{RK}}(m, \phi)$ query, $\mathcal{A}_{H'}$, simulates the $pfNI^{H'_{\mathsf{RK}}}$ by making queries H'_{RK}. She checks whether during the simulation, output of some $H'(k, .)$ query is in T. In such a case, she wins trivially. Note that, this takes care of the left out case in the reduction of identity collision resistance.

If none of the outputs are in T, and $\mathcal{A}_H$ outputs m^*, $\mathcal{A}_{H'}$ simulates the $pfNI$ mode and outputs the last compression function input $(h^*_\ell + 1\|\|10^{n-1})$ as the output.

So in all the cases, if $\mathcal{A}_H$ breaks the ICTPR property of H, $\mathcal{A}_{H'}$ breaks the ICTPR property of H'.

Lemma 2. *Let $H' : \mathcal{K} \times \{0,1\}^{2n} \to \{0,1\}^n$ be a compression function. Let $H : \mathcal{K} \times \{0,1\}^* \to \{0,1\}^n$ be a hash function defined as*
$$H(k, m) \overset{def}{=} pfNI^{H'}(k, m).$$
For all adversary $\mathcal{A}_H$ making q queries of total bit length l, there exists an adversary $\mathcal{A}'_H$ making $\lceil ql/(n-1)\rceil + q$ queries of total bit length $n(\lceil ql/(n-1)\rceil + q)$, such that
$$\mathbf{Adv}^{ictpr}_{H}(\mathcal{A}_H, \Phi) \leq \mathbf{Adv}^{ictpr}_{H'}(\mathcal{A}_{H'}, \Phi)$$

9.2 Constructing ICTPR hash function using Length Preserving RK-MAC

In this section we prove that the $pfNI$ mode instantiated with enciphered CBC-MAC compression function using a length-preserving, related-key-unforgeable function, gives a ICTPR hash function. Let $F : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a family of functions. The $EnCBC$ compression function based on length preserving function F is defined as $H'_{k_1,k_2}(x_1, x_2) = F(k_1, x_1) \oplus F(k_2, x_2)$.

Lemma 3. *Let $F : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a family of related-key unforgeable function over Φ with identity fingerprint $\overline{w} = \{w_1, \cdots, w_d\}$. Define $H' : (\mathcal{K} \times \mathcal{K}) \times \{0,1\}^{2n} \to \{0,1\}^n$ as*

$$H'_{k_1,k_2}(x_1, x_2) \overset{def}{=} F(k_1, x_1) \oplus F(k_2, x_2).$$

Define $H : (\mathcal{K} \times \mathcal{K}) \times \{0,1\}^* \to \{0,1\}^n$ as

$$H(k_1, k_2, m) \overset{def}{=} pfNI^{H'}(k_1, k_2, m)$$

Define $\Psi : \{0,1\}^{2\kappa} \to \{0,1\}^{2\kappa}$ as

$$((\Phi \setminus \{\mathsf{id}\}) \times \Phi) \cup (\mathsf{id}, \mathsf{id})$$

Then H is ICTPR against Related-Key Attack over the RKD set Ψ. For all adversary $\mathcal{A}_H$ making q queries of total bit length l, there exists an adversary $\mathcal{A}_F$ making $\lceil ql/(n-1) \rceil + q$ queries of total bit length $n(\lceil ql/(n-1) \rceil + q)$, such that

$$\mathbf{Adv}_H^{ictpr}(\mathcal{A}_H, \Psi) \leq \left(\frac{q^4}{2} + \frac{q^2 d}{2} \right) \mathbf{Adv}_F^{rk-mac}(\mathcal{A}_F, \Phi)$$

The most natural way to prove the above Lemma will be to show that $EnCBC$ construction, instantiated with RK-MAC gives an ICTPR compression function. However, there is an obstacle to prove such a claim. Recall that we want to show that when there is an ICTPR attack against the compression function, we can mount related-key forgery against the underlying RK-MAC. The general technique is to guess the colliding queries, and predict the output of chronologically last query. Unfortunately, the chronologically last query can indeed be on related-key(the target key of ICTPR attack may be derived from two separate target key queries made before the related-key query).

We give a direct proof the ICTPR security of the mode of operation, instantiated with EnCBC compression function. Specifically, we show that for both the conditions, described in the previous section, we can mount related-key forgery against the underlying MACs. We refer the reader to full version for the full proof.

10 Bellare-Cash construction is MAC preserving

Finally, as an application of Theorem 3, we show that the PRF construction of Bellare and Cash [3], can also be used to construct a related-key unforgeable MAC against chosen message attack. Note that, *this construction uses an unkeyed collision resistance hash function H.* Although, we focused on keyed hash function for all the previous results, we state this result to be complete in our analysis of related-key security of message authentication codes.

Theorem 4. *Let $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ be a weak related-key unforgeable MAC over RKD set Φ with identity fingerprint $\overline{w} = (w_1, w_2, \cdots, w_d)$. Let $H : \{0,1\}^* \to \mathcal{D} \setminus \{w_1, \cdots, w_d\}$ be a collision resistant hash function. Let $G : \mathcal{K} \times \{0,1\}^* \to \mathcal{R}$ be a family of functions defined as*

$$G(k, M) \overset{def}{=} F(k, H(M \| F(k, w_1) \| F(k, w_2) \| \cdots \| F(k, w_d))) \qquad k \in \mathcal{K}.$$

G is related-key unforgeable against chosen message attack over the RKD set Φ. Specifically if there exists a (q, l) adversary $\mathcal{A}_G$ against G, then there exists a $(q, q \log |D|)$ adversary $\mathcal{A}_F$ against F, and a (q, l) adversary $\mathcal{A}_H$ against H such that

$$\boldsymbol{Adv}_H^{cr}(\mathcal{A}_H) + \boldsymbol{Adv}_F^{wrk-mac}(\mathcal{A}_F, \Phi) \geq \boldsymbol{Adv}_G^{rk-mac}(\mathcal{A}_G, \Phi)$$

Proof (Proof Sketch). The proof is similar (infact, special case) to Theorem 3 and we skip the proof.

10.1 Security against partial key transformation from DDH assumption

In this section, we give a concrete construction of a related-key secure MAC based on the following MAC construction, due to Dodis, Kiltz, Pietrzak, and Wichs [12] based on the hash proof system of Cramer and Shoup.

MAC_{HPS}

- **Setup.** p is a large prime. $\mathbb{G}$ is a group of order p. g is a random generator of $\mathbb{G}$. $\hat{H} : \mathbb{G}^2 \times \mathcal{D} \to \mathbb{Z}_p$ is a collision resistant hash function. $\mathcal{K} = \mathbb{Z}_p^3$, $\mathcal{R} = \mathbb{G}^3$.
- **Key Generation**: the secret key is $k = (k_1, k_2, k_3) \leftarrow_{\mathrm{R}} \mathbb{Z}_p^3$.
- **MAC**: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is defined as

$$F(k_1, k_2, k_3, m) \overset{def}{=} (g \leftarrow_{\mathrm{R}} \mathbb{G}, V = g^{k1}, g^{k_2 \hat{H}(g, V, m) + k_3}) \qquad m \in \mathcal{D}, k_1, k_2, k_3 \in \mathbb{Z}_p.$$

For any element $k = (k_1, k_2, k_3) \in \mathcal{K}$ and $\Delta = (0, \delta_2, \delta_3) \in \mathbb{Z}_p^3$, define $k \circ \Delta = (k_1, k_2 + \delta_2, k_3 + \delta_3)$ where $+$ is addition modulo p. It is easy to check that $\mathcal{K}$ is a group under $\circ$. The group induced RKD class over $\mathcal{K}$ will be defined as $\Phi \overset{def}{=} \phi_\Delta(k) = (k \circ \Delta)$.

Although MAC_{HPS} is not key-homomorphic in general, but it is indeed key homomorphic over Φ. Hence, we get the following lemma.

Lemma 4. *MAC_{HPS} is weakly unforgeable against related-key attack over Φ.*

To use Theorem 4, it is now enough to prove the existence of a fingerprint for MAC_{HPS}. Due to space constraint we leave out the identity-fingerprint for MAC_{HPS} in this version.

Theorem 5. *Let $\mathbb{G}$ be a prime order group of p elements, g be a random generator of $\mathbb{G}$. Let w_1, w_2 be two distinct elements from $\mathcal{D}$. Suppose $H : \mathcal{D} \times \mathbb{G} \to$*

$\mathcal{D} \setminus \{w_1, w_2\}$ and $\hat{H} : \mathbb{G}^2 \times \mathcal{D} \to \mathbb{Z}_p$ be two collision resistant hash functions. Define $\mathcal{K} = \mathbb{Z}_p^3$, $\mathcal{R} = \mathbb{G}^3$. Define $G_{HPS} : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ defined as

$$G_{HPS}(k_1, k_2, k_3, m) \overset{def}{=} MAC_{HPS}(k_1, k_2, k_3, H(m, g, g^{k_1}, g^{k_2 \hat{H}(g,V,w_1)+k_3}, g^{k_2 \hat{H}(g,V,w_2)+k_3}))$$

Let $\mathcal{A}_G$ be an adversary against the related-key unforgeability of G under chosen message attack over RKD set Φ, and $\mathcal{A}_G$ makes q queries. Then we can construct an adversary $\mathcal{A}_{DDH}$ against the DDH problem in $\mathbb{G}$, an adversary $\mathcal{A}_H$ against collision resistance of H, and an adversary $\mathcal{A}_{\hat{H}}$ against collision resistance of $\hat{H}$ such that

$$\mathbf{Adv}_G^{rk-mac}(\mathcal{A}_G, \Phi) \leq \mathbf{Adv}_{\mathbb{G}}^{ddh}(\mathcal{A}_{DDH}) + \mathbf{Adv}_H^{cr}(\mathcal{A}_H) + \mathbf{Adv}_{\hat{H}}^{cr}(\mathcal{A}_{\hat{H}})$$

10.2 Towards full security

Previous construction, although very efficient in terms of the keysize, is only secure against partial key transformation. Now, we construct a related-key unforgeable MAC against a full group induced key transformation. The weak unforgeable MAC is based on another construction of Dodis et. al. [12] which is again based on weak PRF and arguments of Waters.

MAC_W

- **Setup.** p is a large prime. $\mathbb{G}$ is a group of order p. Message space is $\{0,1\}^\lambda$. $\mathcal{K} = \mathbb{Z}_p^{\lambda+1}$, $\mathcal{R} = \mathbb{G}^3$.
- **Key Generation**: the secret key is $k = (k_0, k_1, \cdots, k_\lambda) \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{\lambda+1}$.
- **MAC**: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ is defined as

$$F(k_0, k_1, \cdots, k_\lambda, m) \overset{def}{=} (g \leftarrow_{\mathrm{R}} \mathbb{G}, g^{k_0 + \sum_{i=1}^{\lambda} m[i] k_i})$$

For any element $k = (k_0, k_1, \cdots, k_\lambda) \in \mathcal{K}$ and $\Delta = (\delta_0, \delta_1, \cdots, \delta_\lambda) \in \mathbb{Z}_p^{\lambda+1}$, define $k \circ \Delta = (k_0 + \delta_0, \cdots, k_\lambda + \delta_\lambda)$ where $+$ is addition modulo p. It is easy to check that $\mathcal{K}$ is a group under $\circ$. The group induced RKD class over $\mathcal{K}$ will be defined as $\Phi \overset{def}{=} \phi_\Delta(k) = (k \circ \Delta)$.

MAC_W is key-homomorphic in an obvious way. Using Lemma 1

Lemma 5. *MAC_W is weakly unforgeable against related-key attack over Φ.*

Using Theorem 4, we get the following theorem

Theorem 6. *Let $\mathbb{G}$ be a prime order group of p elements. Let*

$$\overline{w} = \{0^\lambda, 10^{\lambda-1}, 010^{\lambda-2}, \cdots, 0^{\lambda-1}1\}$$

Suppose $H : \mathcal{D} \times \mathbb{G}^{2(\lambda+1)} \to \mathcal{D} \setminus \{\overline{w}\}$ be a collision resistant hash functions. Define $\mathcal{K} = \mathbb{Z}_p^{\lambda+1}$, $\mathcal{R} = \mathbb{G}^2$. Define $G_W : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ as

$$G_W(k, m) \overset{def}{=} MAC_W(k, H(m, MAC_W(k, 0^\lambda), MAC_W(k, 10^{\lambda-1}), \cdots, MAC_W(k, 0^{\lambda-1}1)))$$

Let $\mathcal{A}_G$ be an adversary against the related-key unforgeability of G_W under chosen message attack over RKD set Φ, and $\mathcal{A}_G$ makes q queries. Then we can construct an adversary $\mathcal{A}_{DDH}$ against the DDH problem in $\mathbb{G}$, an adversary $\mathcal{A}_H$ against collision resistance of H such that

$$\mathbf{Adv}_G^{rk-mac}(\mathcal{A}_G, \Phi) \leq \mathbf{Adv}_{\mathbb{G}}^{ddh}(\mathcal{A}_{DDH}) + \mathbf{Adv}_H^{cr}(\mathcal{A}_H)$$

11 Conclusion

Security against related-key attacks is currently considered as a major challenge for symmetric key cryptography. In this paper, we considered security of message authentication codes against related-key attacks. We formalized the security definitions and identified feasible key transformations. We also presented the first security analysis for domain extension of related-key secure unpredictable functions(MAC). However our reduction for the Enciphered CBC construction achieves a reduction-factor of $\mathcal{O}(2^{n/4})$ queries (Lemma 3). Finding constructions with improved security bound is an interesting open problem. Specifically, analysis of related-key security of Dodis-Steinberger construction [15] will be very interesting.

12 Acknowledgements

We thank Mridul Nandi for useful discussions. We also thank Damien Stehlé for important feedback on the initial draft of the paper. We are grateful to the anonymous reviewers of FSE 2013 for insightful comments. Part of this work was done when Rishi was at the Centre of Excellence in Cryptology of Indian Statistical Institute, Kolkata.

References

1. Jee Hea An and Mihir Bellare. Constructing VIL-MACs from FIL-MACs: Message authentication under weakened assumptions. In Michael J. Wiener, editor, *CRYPTO'99*, volume 1666 of *LNCS*, pages 252–269. Springer, August 1999.
2. Benny Applebaum, Danny Harnik, and Yuval Ishai. Semantic security under related-key attacks and applications. In *ICS*, pages 45–60, 2011.
3. Mihir Bellare and David Cash. Pseudorandom functions and permutations provably secure against related-key attacks. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 666–684. Springer, August 2010.
4. Mihir Bellare, David Cash, and Rachel Miller. Cryptography secure against related-key attacks and tampering. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 486–503. Springer, December 2011.
5. Mihir Bellare and Tadayoshi Kohno. A Theoretical Treatment of Related-Key Attacks: RKA-PRPs, RKA-PRFs, and Applications. In *EUROCRYPT*, pages 491–506, 2003.

6. Mihir Bellare and Tadayoshi Kohno. A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In Eli Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 491–506. Springer, May 2003.
7. Eli Biham. New types of cryptanalytic attacks using related keys. *Journal of Cryptology*, 7(4):229–246, 1994.
8. Eli Biham, Orr Dunkelman, and Nathan Keller. Related-key boomerang and rectangle attacks. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 507–525. Springer, May 2005.
9. Eli Biham, Orr Dunkelman, and Nathan Keller. A related-key rectangle attack on the full KASUMI. In Bimal K. Roy, editor, *ASIACRYPT 2005*, volume 3788 of *LNCS*, pages 443–461. Springer, December 2005.
10. Alex Biryukov, Orr Dunkelman, Nathan Keller, Dmitry Khovratovich, and Adi Shamir. Key recovery attacks of practical complexity on AES-256 variants with up to 10 rounds. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 299–319. Springer, May 2010.
11. Alex Biryukov, Dmitry Khovratovich, and Ivica Nikolic. Distinguisher and related-key attack on the full AES-256. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 231–249. Springer, August 2009.
12. Yevgeniy Dodis, Eike Kiltz, Krzysztof Pietrzak, and Daniel Wichs. Message authentication, revisited. In *Cryptology ePrint Archive*, 2012. `http://eprint.iacr.org/2012/059`.
13. Yevgeniy Dodis, Eike Kiltz, Krzysztof Pietrzak, and Daniel Wichs. Message authentication, revisited. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 355–374. Springer, April 2012.
14. Yevgeniy Dodis, Krzysztof Pietrzak, and Prashant Puniya. A new mode of operation for block ciphers and length-preserving MACs. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 198–219. Springer, April 2008.
15. Yevgeniy Dodis and John P. Steinberger. Message authentication codes from unpredictable block ciphers. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 267–285. Springer, August 2009.
16. Vipul Goyal, Adam O'Neill, and Vanishree Rao. Correlated-input secure hash functions. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 182–200. Springer, March 2011.
17. Lars R. Knudsen. Cryptanalysis of LOKI91. In Jennifer Seberry and Yuliang Zheng, editors, *AUSCRYPT'92*, volume 718 of *LNCS*, pages 196–208. Springer, December 1992.
18. Stefan Lucks. Ciphers secure against related-key attacks. In Bimal K. Roy and Willi Meier, editors, *FSE 2004*, volume 3017 of *LNCS*, pages 359–370. Springer, February 2004.
19. Thomas Peyrin, Yu Sasaki, and Lei Wang. Generic related-key attacks for HMAC. In *ASIACRYPT 2012*, pages 580–597. Springer, 2012.
20. Keita Xagawa. Message authentication codes secure against additively related-key attacks. Cryptology ePrint Archive, Report 2013/111, 2013. `http://eprint.iacr.org/2013/111`.