Clever Arbiters versus Malicious Adversaries

On the Gap between Known-Input Security and Chosen-Input Security

Serge Vaudenay

EPFL
CH-1015 Lausanne, Switzerland
http://lasec.epfl.ch

Abstract. When moving from known-input security to chosen-input security, some generic attacks sometimes become possible and must be discarded by a specific set of rules in the threat model. Similarly, common practices consist of fixing security systems, once an exploit is discovered, by adding a specific rule to thwart it. To study feasibility, we investigate a new security notion: security against undetectable attacks. I.e., attacks which cannot be ruled out by any specific rule based on the observable behavior of the adversary. In this model, chosen-input attacks must specify inputs which are indistinguishable from the ones in known-input attacks. Otherwise, they could be ruled out, in theory.
Although non-falsifiable, this notion provides interesting results: for any primitives based on symmetric encryption, message authentication code (MAC), or pseudorandom function (PRF), known-input security is equivalent to this restricted chosen-input security in Minicrypt. Otherwise, any separation implies the construction of a public-key cryptosystem (PKC): for a known-input-secure primitive, any undetectable chosen-input attack transforms the primitive into a PKC.
In this paper, we develop the notion of security based on open rules. We show the above results. We revisit the notion of related-key security of block ciphers to illustrate these results. Interestingly, when the relation among the keys is specified as a black box, no chosen-relation security is feasible. By translating this result to non-black box relations, either no known-input security is feasible, or we can recognize any obfuscated relation by a fixed set of rules, or we can build a PKC. Any of these three results is quite interesting in itself.

1 Preamble

Children often use adaptive rules in their games. Indeed, ruling a game is usually the result of a learning process. There are also common practices to motivate games with adaptive rules. Irrespective on whether this is good or bad, computer security often relies on security patches, or new signatures in anti-virus systems, which appear once an exploit is known.

In cryptography, security definitions followed a similar learning process. We often have to rule out some specific attacks once we realize that no security is

feasible because there exists *generic* attacks. For instance, to model resistance to chosen-ciphertext attack (CCA), we define a game where the adversary can query a ciphertext to a decryption oracle. As he must distinguish whether a given ciphertext c encrypts a message m_0 or a message m_1, a first rule says that he cannot query the decryption oracle after c is determined. This is the security against "lunchtime attack" [30]. Another rule allows further queries, conditioned to that they are not equal to c. This is the standard CCA security [15,16]. Clearly, no security is feasible without this rule.

A more complicated case is the one of related-key attacks (RKA) [6,7,28]. In this model, the adversary can query a plaintext and a transformation of the key. There are many attacks based on some weird transformations. In Appendix A.1, we describe the attacks by Biham [8], Bellare and Kohno [3], Harris [22,23], and Bernstein [5]. These attacks show that no RKA security is feasible without some specific rules making these attacks forbidden. To rule them out, the easy way is to add some drastic rules such as the transformation must be of the form $k \mapsto k \oplus \Delta$. But the question of a minimal set of rules allowing any random-looking transformation remains.

In this work, we describe the security with adaptive rules as a game, with a challenger, an adversary, and a ruler trying to catch the malicious behavior of the adversary. The game consists of playing with an oracle to evaluate a keyed primitive f_K. So, we distinguish known-input security, where the inputs to the oracle are random, to chosen-input security, where the adversary selects the input. Ruling out malicious behaviors means to restrict to adversaries making chosen inputs indistinguishable from known inputs.

The Paper At A Glance

Setting. We consider a keyed primitive denoted $f_K(q)$. This primitive is set up with a key K and one bit b (which is supposed to be a hard-core bit of K, as we will explain later). An example to consider is given by $f_K(q) = \mathsf{Enc}_{\varphi(K)}(x)$ for $q = (\varphi, x)$, where Enc is some encryption function. In this case, the input φ is referred to as *relation* φ (in reminiscence of *related*-key security) and x as a plaintext.

We further consider the problem of guessing whether the coin b is a Head Or a Tail, i.e., the HOT *game*. In this game, the adversary ignores the key but he can make oracle queries to f_K. We distinguish between the case where q is chosen by the adversary and the case where q is selected based on a random distribution D, i.e., chosen-input attack vs. known-input attack.

On finding a minimal set of rules for related-key security. In related-key security, the adversary must provide a relation to the challenger. In a black-box model, this relation is provided in terms of access to an oracle (i.e., the inner structure of the relation is not visible). Otherwise, relations must be specified in terms of an executable code (or Turing machine). Since there is a double-exponential number of relations, we must consider only relations that can be implemented by a short code and specify a distribution for known-relation security. Alternatively,

we must substantially restrict the set of relations, e.g., by taking the set T_+ of all translations $\varphi(K) = K + \Delta$, given a group law "+" over the key-space. Even in that case, a separation between known-relation security and chosen-relation security induces a public-key cryptosystem.

Defining a sound model for related-key security appears to be challenging as many "trivial attacks" using convoluted relations have been discovered, e.g., [6,7]. For this reason, we introduce the notion of game with rules which could be updated incrementally. Indeed, our security game comprises an adversary, a challenger, and a ruler who performs a checking on that the adversary did not select unauthorized relations. This model is particularly useful to show the nonexistence of rules making security feasible.

Our focus. In this paper, we consider a restricted chosen-input model. Thus, we look at chosen inputs that are indistinguishable from the random ones present in known-input models.

Known-input security vs. certain chosen-input security. We observe that having a separation between known-input security and restricted chosen-input security yields the ability to construct a public-key cryptosystem. I.e., if we have a primitive secure against known-input attacks but vulnerable to some chosen-input attack, the cryptosystem's design is based on the primitive and it exploits the attack. In the Minicrypt world [26], public-key cryptography does not exist but symmetric cryptography does. So, therein we cannot have any separation. So, known-input security implies our restrictive chosen-input security.

Black-box vs. non black-box related-key security. We further show that the separation actually holds for related-key security when relations are considered as black-boxes. To remove black-box relations, we consider obfuscated white boxes. Since it is unlikely that one could build a cryptosystem from a block cipher and an adversary, we deduce that either no known-relation security is possible, or there is a generic way to break obfuscation schemes for relations.

Our contribution concisely. In this paper we formalize the notion of ruler/arbiter of a security game and the security notions linked to this. We prove that a gap between known-input security and permissive chosen-input security implies public-key cryptography. We show that the gap exists for related-key security in a black-box model. When removing the black-boxes using an obfuscation scheme, we deduce that either no known-relation security is feasible, or any obfuscation scheme is weak, or it makes a public-key cryptosystem.

Structure of this paper. In Section 2, we introduce some meta-security notions via the (formal) concepts of game, ruler, permissive ruler, known-input attack, and chosen-input attack. In Section 3 we show that a gap between known-input security and permissive chosen-input security implies public-key cryptography. In Section 4 we extend the Harris attack to break any cipher using related keys and we show that no permissive ruler can detect it when the relations used are

black-boxes. We further discuss on extending this with obfuscation and on the difficulty to identity the exact rules to make related-key security sound.

Related work. Related-key attacks independently appeared in Biham's [6,7] and Knudsen's [28]. Basically, an adversary has access to some encryption/decryption black boxes which relate to each other by some known- or chosen-relations by the adversary. Concretely, the adversary makes (φ, x) queries for a relation φ and a plaintext x and gets back $\mathsf{Enc}_{\varphi(K)}(x)$, i.e., the encryption of x under key $\varphi(K)$. (In chosen ciphertext attacks, the adversary can query (φ, y) to get back $\mathsf{Enc}^{-1}_{\varphi(K)}(y)$.) In the literature, cryptanalysts have looked for relations φ such that the adversary could get an advantage in this model. Although the relevance of this model has been controversial, it is widely admitted that, for some applications, these attacks can pose a real threat. Indeed, in some applications, keys can be updated in a way which might be known (or influenced) by an adversary. To make the attack model as general as possible, it is tempting to allow *any* relation. Unfortunately, we can then show that no security is feasible without more restrictions.

Bellare and Kohno [3] studied a formal model for related-key security. Their model had to be relative to a set of authorized permutations. It works in the ideal cipher model (that is, when the block cipher is random and only usable through specific oracle accesses) and when the relations selected by the adversary are not cipher-dependent (that is, to evaluate a relation, we shall not have any access to the encryption or decryption oracles). They proposed some sufficient conditions for identifying authorized relations. These results were extended by Farshim, Paterson, Albrecht, and Watson [18] by allowing relations to depend on the ideal cipher but obeying extra conditions.

Lucks [29] studied related-key security based on partial transformations, i.e., relations modifying only a part of the key. Another approach by Goldenberg and Liskov [19] shows that related-key security can be achieved by (and can make) a related-key pseudorandom bit. Bellare and Cash [2] constructed one by using public-key cryptography techniques.

Other similar existential results exist. For instance, Pietrzak [31] shows that for any $k \geq 2$, either there exists a secure key agreement protocol working with k messages, or the sequential composition of $(k-1)$-adaptively secure PRF is a k-adaptively secure PRF. (k-adaptive security refers to adversaries allowed to make up to k round of queries where queries in the same round as selected at the same time.) So, in the Minicrypt world where we have no key agreement protocol, sequential composition transform non-adaptive security into adaptive security.

Notations. In what follows, we will consider asymptotic security notions.[1] That is, cryptographic algorithms and parameters shall depend on a security param-

[1] Exact, i.e., not asymptotic, security could also be considered, but it would require heavier notations.

eter λ. Specifically, these algorithm run in time which is polynomial in λ. Adversaries defeating the security requirements do so via computations that are polynomial in terms of λ. For readability, the parameter λ will be omitted from certain notations.

A function negl is *negligible* if for any integer d we have $\mathsf{negl}(\lambda) \in \mathcal{O}(\lambda^{-d})$. A function whose inverse is polynomially bounded is not negligible.

We will be using the Hoeffding bound [24]: for $X_1, \ldots, X_n$ independent identically distributed (i.i.d.) Bernoulli random variables of expected value p and $t \geq 0$,

$$\Pr\left[\frac{1}{n}\sum_{i=1}^{n} X_i \geq p + t\right] \leq e^{-2nt^2} \quad , \quad \Pr\left[\frac{1}{n}\sum_{i=1}^{n} X_i \leq p - t\right] \geq e^{-2nt^2}$$

In the special case where $t = |p - \frac{1}{2}|$, we obtain that the majority of $X_1, \ldots, X_n$ does not correspond to the most likely value of X_1 with probability at most $e^{-2n(p-\frac{1}{2})^2}$. This will be referred to as the Chernoff bound [10].

2 Ruler-Based Security Models

We define here some meta-security notions, encapsulated in the security game $\Gamma_{\mathcal{F}}(\mathcal{A}, \lambda)$ for a primitive $\mathcal{F}$. These notions comprise the adversary $\mathcal{A}$, the challenger $\mathcal{C}$, the advantage that $\mathcal{A}$ may have at winning this game, and a special measure of the latter called *uniform advantage*.

Keyed Primitive. Throughout this paper, we consider a "keyed primitive" $\mathcal{F}$ defined by the following: 1. a generator Gen generating a coin $b \in \{0,1\}$ and some $K \in \mathcal{K}$, i.e., $(b, K) \leftarrow \mathsf{Gen}$; 2. an algorithm $f_K(q)$ taking as input a key K, a "query" $q \in \mathcal{D}$. The function f_K may be probabilistic. Again, a natural choice for related-key security would be to consider $f_K(q) = \mathsf{Enc}_{\varphi(K)}(x)$ for $q = (\varphi, x)$, where Enc is some encryption function. Here, φ is called a "relation".

Definition 1 (The $\Gamma_{\mathcal{F}}(\mathcal{A}, \lambda)$ Security Game). *Given a keyed primitive $\mathcal{F}_\lambda$ depending on some security parameter λ, we consider a* game $\Gamma_{\mathcal{F}}(\mathcal{A}, \lambda)$ *between two principles called an* adversary $\mathcal{A}$ *and a* challenger $\mathcal{C}$*. The adversary is arbitrary. The challenger is specified in the game. Both are probabilistic interactive Turing machines running with expected polynomial time in terms of λ. The game consists of setting up both $\mathcal{A}$ and $\mathcal{C}$ with some independent random coins ρ and $\rho_{\mathcal{C}}$ (respectively), then running an interactive protocol between them and waiting for a final outcome $\Gamma_{\mathcal{F}}(\mathcal{A}, \lambda) = 0$ or 1. If the outcome is 0, we say that the adversary wins.*

The advantage *of the adversary is*

$$\mathsf{Adv}_{\Gamma_{\mathcal{F}}}(\mathcal{A}, \lambda) = \Pr_{\rho, \rho_{\mathcal{C}}}[\Gamma_{\mathcal{F}}(\mathcal{A}, \lambda) = 0] - \Pr_{\rho, \rho_{\mathcal{C}}}[\Gamma_{\mathcal{F}}(\mathcal{A}, \lambda) = 1]$$

where the probability goes over all random coins. We say that $\mathcal{F}$ is Γ-secure if for any $\mathcal{A}$, $\mathsf{Adv}_{\Gamma_{\mathcal{F}}}(\mathcal{A}, \lambda)$ is negligible in terms of λ. The uniform advantage *of*

the adversary is

$$\mathsf{UAdv}_{\Gamma_{\mathcal{F}}}(\mathcal{A},\lambda) = \min_{\rho_{\mathcal{C}}} \left(\Pr_{\rho}[\Gamma_{\mathcal{F}}(\mathcal{A},\lambda)=0] - \Pr_{\rho}[\Gamma_{\mathcal{F}}(\mathcal{A},\lambda)=1] \right)$$

In the definition, the advantage is the difference between the probability to win and the probability to lose. We could have taken $\Pr_{\rho}[\Gamma_{\mathcal{F}}(\mathcal{A},\lambda)=0] - \frac{1}{2}$ as a definition. We prefer our formalism since it facilitates the extension to games producing a third possible outcome. And, indeed, we are going to consider games which can abort, so the outcome could be 0, 1, or abort.

We also defined the notion of *uniform advantage* as it is easy to amplify (as shown in Lemma 2 below). It captures high advantages whatever the coins used by the challenger.

Head-Or-Tail game. We consider the following Head-Or-Tail game (HOT), which we denote by $\mathsf{HOT}_{\mathcal{F}}(\mathcal{A})$:

1. Using fresh coins from $\rho_{\mathcal{C}}$, run Gen to generate b and K.
2. Run $\mathcal{A}(\rho)$ iteratively and answer its queries q_i by $y_i = f_K(q_i)$. I.e., $q_1 = \mathcal{A}(\rho)$, $q_2 = \mathcal{A}(y_1;\rho)$, $q_3 = \mathcal{A}(y_1,y_2;\rho)$, ... If f_K is probabilistic, running $y_i = f_K(q_i)$ assumes independent coins which are taken from $\rho_{\mathcal{C}}$.
3. Whenever $\mathcal{A}$ stops making queries and outputs a bit γ, stop and yield $b \oplus \gamma$. I.e., $\mathcal{A}$ wins if $\gamma = b$.

The primitive is stateless in the sense that queries $f_K(q)$ produce a distribution which only depends on q throughout the execution of the game. The last output of the adversary is a bit denoted by γ. The primitive $\mathcal{F}$ is secure if no adversary can guess b by playing with f_K. Later, we use b set as a function of K (a hard-core bit of K).

Amplification of uniform advantages. The notion of uniform advantage relates to advantages that do not depend on the random coins of the challenger. This notion is convenient for amplifying an advantage of λ^{-d} to $1-\mathsf{negl}$. The following lemma shows this exactly.

Lemma 2 (Amplification Lemma). *If a polynomial adversary $\mathcal{A}$ has a uniform advantage ε in the $\mathsf{HOT}_{\mathcal{F}}$ game, where $\varepsilon = \Omega(\lambda^{-d})$ for some d, then we can build a polynomial adversary with uniform advantage $1-\mathsf{negl}(\lambda)$ in the $\mathsf{HOT}_{\mathcal{F}}$ game.*

This extends to any other game in which the following holds: 1. the challenger is stateless (that is, its state before any query is fully determined by its random tape $\rho_{\mathcal{C}}$); 2. the outcome of the game is a function $g(\gamma,\rho_{\mathcal{C}})$, with $\rho_{\mathcal{C}}$ being the random coins of $\mathcal{C}$ and γ being the bit eventually produced by the adversary.

Proof. Due to the assumptions, the adversary has two possible choices for the output γ. Furthermore, for any $\rho_{\mathcal{C}}$, $\mathcal{A}$'s choice γ leads to $b \oplus \gamma = 0$ with probability $p \geq \frac{1+\varepsilon}{2}$ over ρ. We define an adversary who simulates the adversary $\mathcal{A}$ repeatedly

$k = \lambda^{2d+1}$ times and who finally outputs $\bar{\gamma}$ to be the majority of $\mathcal{A}$'s outputs γ. Since the challenger is stateless and the advantage is uniform, whatever the key and the unique coins used by the challenger, every iteration of the adversary makes independent γs such that $b \oplus \gamma = 0$ with probability p. Due to the Chernoff bound [10], with probability less than $e^{-2k(p-\frac{1}{2})^2} \leq e^{-\frac{k\varepsilon^2}{2}} \leq e^{-\Omega(\lambda)} = \mathsf{negl}(\lambda)$, the majority $\bar{\gamma}$ of all γs is such that $b \oplus \bar{\gamma} \neq 0$. □

Known-input security. We now define the notion of known-input security. For that, we need to specify the distribution of randomly selected relations.

Definition 3 (D-known-input (KI) security). *Consider a distribution D over $\mathcal{D}$ which is polynomially samplable. We say that the adversary $\mathcal{A}$ in the $\mathsf{HOT}_{\mathcal{F}}(\mathcal{A})$ game is D-KI if each of his queries q is either identical to a previous query[2] or a freshly sampled random query following the distribution D. These queries are sampled independently.*

We say that $\mathcal{F}$ is HOT-KI-secure for D if for all D-KI adversary the advantage in the corresponding HOT game is negligible.

Rulers. When defining restricted chosen-input security, we will introduce some new rules in the game which will be enforced by an extra process called "ruler". Typically, we will require that inputs chosen by the adversary are indistinguishable from inputs sampled in a known-input attack.

Given a keyed primitive $\mathcal{F}$ as in the HOT game, we will define the RHOT game involving a ruler. A *ruler* is a probabilistic polynomial time Turing machine $\mathcal{R}$ which produces a bit given a possible view of the challenger.

Definition 4 (Rulers and Ruled Games). *Given the list $q_1, \ldots, q_n$ of queries from the adversary and the random coins $\rho_{\mathcal{C}}$ of the challenger in the HOT game,* ruler $\mathcal{R}$ *computes a bit denoted $\tilde{b} = \mathcal{R}(q_1, \ldots, q_n, \rho_{\mathcal{C}}; \rho_{\mathcal{R}})$. The ruled-game RHOT runs as follows:*

Game $\mathsf{RHOT}_{\mathcal{F}}(\mathcal{A}, \mathcal{R})$*:*

1: *pick $\rho_{\mathcal{R}}$ at random*
2: *run the HOT game as before until γ is set, denote $q_1, \ldots, q_n$ the queries from $\mathcal{A}$ and $\rho_{\mathcal{C}}$ the coins of the challenger*
3: $\tilde{b} \leftarrow \mathcal{R}(q_1, \ldots, q_n, \rho_{\mathcal{C}}; \rho_{\mathcal{R}})$
4: ***if*** $\tilde{b} = 1$ ***then***
5: *return* abort
6: ***else***
7: *return* $b \oplus \gamma$
8: ***end if***

When $\tilde{b} = 0$, we say that $\mathcal{A}$ follows the rules *of $\mathcal{R}$. Otherwise, we say that $\mathcal{R}$* rules over *$\mathcal{A}$.*

The advantage *of $\mathcal{A}$ for ruler $\mathcal{R}$ is*

$$\mathsf{Adv}_{\mathsf{RHOT}_{\mathcal{F}}}(\mathcal{A}, \mathcal{R}) = \Pr[\mathsf{RHOT}_{\mathcal{F}}(\mathcal{A}, \mathcal{R}) = 0] - \Pr[\mathsf{RHOT}_{\mathcal{F}}(\mathcal{A}, \mathcal{R}) = 1]$$

[2] Since $\mathcal{F}$ may be probabilistic, it may be useful to repeat a query.

We say that $\mathcal{F}$ is RHOT*-secure for a class of rulers if for all $\mathcal{A}$ there is a ruler $\mathcal{R}$ in this class such that* $\mathsf{Adv}_{\mathsf{RHOT}_{\mathcal{F}}}(\mathcal{A},\mathcal{R})$ *is negligible.*

The ruler captures the common practice of encompassing known threats in a model for attack-detection. I.e., in related-key security, some "trivial attacks" breaking any cipher can be deployed. (See Appendix A.1.) However, these attacks use specific relations which can be added to the security model, i.e., for the ruler to check on the fly if such "trivial attacks" are taking place.

The advantage is

$$\begin{aligned}\mathsf{Adv}_{\mathsf{RHOT}_{\mathcal{F}}}(\mathcal{A},\mathcal{R}) &= \Pr[\mathsf{RHOT}_{\mathcal{F}}(\mathcal{A},\mathcal{R})=0]-\Pr[\mathsf{RHOT}_{\mathcal{F}}(\mathcal{A},\mathcal{R})=1]\\ &= \Pr[\tilde{b}=0](\Pr[\gamma=b|\tilde{b}=0]-\Pr[\gamma\neq b|\tilde{b}=0])\end{aligned}$$

That is, this is the advantage given that the game follows the rules defined by the ruler, multiplied by the probability to follow the rules. It is necessary to consider the probability to follow the rules since an adversary with high advantage but almost never in the legal case would certainly be insignificant for security.

The HOT game can be seen as a RHOT game in which the ruler would always output 0, i.e., $\mathcal{R}$ would allow every "behavior" of $\mathcal{A}$. Conversely, a ruler answering 1 too often would make $\mathcal{F}$ trivially secure, i.e., if all is forbidden, then no attack is possible. Thus, to make security non-trivial we require rulers that are, in some sense, permissive, i.e., they allow the adversary to play as long as we cannot see any malicious behavior.

Definition 5 (Permissive ruler). *Given a keyed primitive $\mathcal{F}$ and a polynomially samplable distribution D over the set $\mathcal{D}$ of inputs, we say that a ruler $\mathcal{R}$ is permissive for D if for any D-*KI *adversary $\mathcal{A}$, the probability that $\mathcal{R}$ rules over $\mathcal{A}$ in the* RHOT *game is negligible.*

The above definition says that D-permissive rulers allow adversaries to select inputs by sampling D. Clearly, the AND/OR of a polynomial number of permissive rules is also a permissive rule.

Chosen-input security. We now give a restricted notion of chosen-input (CI) security called PCI.

Definition 6 (D-permissive chosen-input (PCI) security). *Given a keyed primitive $\mathcal{F}$, consider a polynomially samplable distribution D over the set $\mathcal{D}$ of inputs. We say that $\mathcal{F}$ is* RHOT-PCI*-secure for D if it is* RHOT*-secure for the class of all permissive rulers for D. I.e., for any adversary $\mathcal{A}$, there is a D-permissive ruler $\mathcal{R}$ such that* $\mathsf{Adv}_{\mathsf{RHOT}}(\mathcal{A},\mathcal{R})$ *is negligible.*

So, we only rule out CI attacks whose behavior can be distinguished from the one of KI attacks.

3 The PCI/KI Gap Includes Public-Key Cryptography

3.1 Our Result

We show that if there exists a non-adaptive CI-adversary $\mathcal{A}$ successfully attacking a keyed primitive in front of any D-permissive ruler $\mathcal{R}$ and if the primitive resists KI attacks, then we can construct a cryptosystem from $\mathcal{F}$ and $\mathcal{A}$.

Lemma 7. *Consider a keyed primitive $\mathcal{F}$ over the key domain $\mathcal{K}$ in which the generator* Gen *produces balanced coin b. Assume that $\mathcal{F}$ is* HOT-KI*-secure for a distribution D. If there is a non-adaptive adversary $\mathcal{A}$ in the* RHOT *game with advantage $1-\mathsf{negl}(\lambda)$ for all permissive rulers for D, then we have a public-key cryptosystem defined by the following:*

- The key generation: *pick a secret key ρ and the public key $q=(q_1,\ldots,q_n)\leftarrow\mathcal{A}(\rho)$, the non-adaptive queries by $\mathcal{A}$.*
  We write $(\rho,q)=\mathsf{PKGen}(\rho)$. I.e., ρ is the secret key and q is the public one.
- The encryption of a bit β: *pick $\rho_{\mathcal{C}}=(\rho^0_{\mathcal{C}},\rho^1_{\mathcal{C}},\ldots)$ randomly and do: $(b,K)\leftarrow\mathsf{Gen}(\rho^0_{\mathcal{C}})$, $y\leftarrow(f_K(q_i;\rho^i_{\mathcal{C}}))_{i=1,\ldots,n}$, and $e\leftarrow\beta\oplus b$.*
  We write $(y,e)=\mathsf{PKEnc}_q(\beta;\rho_{\mathcal{C}})$.
- The decryption of (y,e): *do $b'\leftarrow\mathcal{A}(y;\rho)\oplus e$.*
  We write $b'=\mathsf{PKDec}_\rho(y,e)$.

This cryptosystem is correct and secure.

This lemma uses an adversary $\mathcal{A}$ producing its set of queries q non-adaptively, which is the public key. Then, the encryption of β is the answers to the queries (with a fresh (b,K) and some fresh coins for f_K) together with $\beta\oplus b$. The bit b can be guessed by $\mathcal{A}$ for decryption. The high advantage of $\mathcal{A}$ makes the cryptosystem correct. Any decryption algorithm $\mathcal{E}$ would imply a permissive ruler to detect the behavior of $\mathcal{A}$. Since $\mathcal{A}$ cannot be ruled over, the cryptosystem is secure.

Proof. Let us assume that an adversary as above exists. Let q and y be the vectors of query-inputs and query-outputs to and from the challenger, respectively.

$\mathsf{PKDec}_\rho(y,e)=\beta$ is equivalent to $b=\mathcal{A}(y;\rho)$, i.e., to $\mathcal{A}$ winning in the HOT game. By definition, for any permissive ruler $\mathcal{R}$, we have

$$1-\mathsf{negl}(\lambda)=\mathsf{Adv}(\mathcal{A},\mathcal{R})=\Pr[\mathcal{R}\text{ accepts}]\,(1-2\Pr[\mathsf{PKDec}_\rho(y,e)\neq\beta|\mathcal{R}\text{ accepts}])$$

We apply this to the ruler $\mathcal{R}$ who always accepts. We obtain that the probability that (y,e) does not decrypt to β is negligible. This holds for any β. So, the cryptosystem satisfies correctness. So, what remains to be proven is its security.

Consider some algorithm $\mathcal{E}(q,y,e)$ trying to decrypt (y,e) given a public key q, and let $\varepsilon_q=\Pr[\mathcal{E}(q,\mathsf{PKEnc}_q(\beta;\rho_{\mathcal{C}}))=\beta]-\frac{1}{2}$ over a random $\rho_{\mathcal{C}}$ and β, for the public key q fixed. We want to show that $E(\varepsilon_q)$ is negligible over ρ, for $q=\mathcal{A}(\rho)$.

We construct rulers $\mathcal{R}_d(q,\rho_{\mathcal{C}};\rho_{\mathcal{R}})$ based on $\mathcal{E}$ as follows. For a number of $k=\lambda^{2d+1}$ random $\rho_{\mathcal{C}}(j)$ and β_j, encrypt β_j under coins $\rho_{\mathcal{C}}(j)$ with public key q

and get the ciphertext $(y_j, e_j) = \mathsf{PKEnc}_q(\beta_j; \rho_{\mathcal{C}}(j))$. Then, count for how many j's we have $\beta_j = \mathcal{E}(q, y_j, e_j)$. If this number is above the threshold $t = k(\frac{1}{2}+\lambda^{-d})$, then rule over $\mathcal{A}$. Note that this ruler makes no use of $\rho_{\mathcal{C}}$. It is just testing the public key q and it aborts if $\mathcal{E}(q, ., .)$ breaks this with an advantage that is too large. We will show that $\mathcal{R}_d$ is permissive and deduce that it rules over $\mathcal{A}$ with negligible probability. Consequently, $E(\varepsilon_\rho) \leq \lambda^{-d} + \mathsf{negl}(\lambda)$. As it holds for all d, we conclude that $E(\varepsilon_\rho)$ is negligible.

We first show that $\mathcal{R}_d$ is permissive. We consider an arbitrary D-KI adversary $\mathcal{A}'$ generating n KI queries $q' = (q'_1, \ldots, q'_n)$, receiving the responses y' based on some coins $\rho_{\mathcal{C}}$, and producing a final bit γ'. We want to show that $\Pr[\mathcal{R}_d(q'; \rho_{\mathcal{C}}) = 1] = \mathsf{negl}(\lambda)$. (We recall that the outcome γ'' is not provided to the ruler.) To do so, we must estimate $\varepsilon_{q'}$.

For this, we construct another adversary $\mathcal{A}''$ who makes the same queries q' as $\mathcal{A}'$ but computes his final γ'' in a special way. We define $\mathcal{A}''$ as follows: $\mathcal{A}''(\rho'')$ simulates $\mathcal{A}'(\rho')$ with some fresh coins ρ' taken from ρ'', sets $q'' = q'$ and gets the responses $y'' = y'$ based on some coins $\rho_{\mathcal{C}}$. Then, it picks some random bit e and computes $\gamma_1 = \mathcal{E}(q', y', e) \oplus e$. Note that for q' fixed, if b is the bit generated by Gen from $\rho_{\mathcal{C}}$ we have $\gamma_1 = b$ with probability $\frac{1}{2} + \varepsilon_{q'}$. In addition to this, $\mathcal{A}''$ picks a random β and computes $\gamma_2 = \mathcal{E}(q', \mathsf{PKEnc}_{q'}(\beta)) \oplus \beta$. For q' fixed, we have $\gamma_2 = 0$ with probability $\frac{1}{2} + \varepsilon_{q'}$. The final answer is $\gamma'' = \gamma_1 \oplus \gamma_2$. So, we have $\gamma'' = b$ with probability $\frac{1}{2} + 2\varepsilon^2_{q'}$, which is the probability for $\mathcal{A}''$ to win the $\mathsf{HOT}_{\mathcal{F}}(\mathcal{A}'')$ game. Clearly, $\mathsf{Adv}_{\mathsf{HOT}_{\mathcal{F}}}(\mathcal{A}'') = 4E(\varepsilon^2_{q'})$ over the random choice of ρ' and $q' = \mathcal{A}'(\rho')$. Since $\mathcal{A}''$ is a D-KI adversary, due to D-KI security, we obtain that $E(\varepsilon^2_{q'})$ is negligible.

Let B be the event that $\varepsilon^2_{q'} \leq \lambda^{-2d-2}$ for d fixed. Since $\Pr[\neg B] \leq \lambda^{2d+2} E(\varepsilon^2_{q'})$, we have that $\Pr[\neg B]$ is negligible. So, B holds except in negligible cases. When B holds, we have $\varepsilon_{q'} \leq \lambda^{-d-1}$. The Hoeffding bound [24] deduces that $\mathcal{R}_d$ aborts with a probability bounded by $e^{-2\lambda(1-\lambda^{-1})^2}$, which is negligible. So, the overall probability that $\mathcal{R}_d$ aborts on queries q' is negligible when B holds, and other cases are negligible. So, $\mathcal{R}_d$ aborts on queries made by an arbitrary KI adversary $\mathcal{A}'$ with negligible probability. Therefore, $\mathcal{R}_d$ is permissive.

We now go back to the adversary $\mathcal{A}$ using the permissive ruler $\mathcal{R}_d$. Due to our assumptions, $\mathcal{R}_d$ rules over $\mathcal{A}$ with negligible probability. If $\varepsilon_\rho \geq \lambda^{-d}$, then by applying same reasoning as above, we obtain that the probability for the adversary $\mathcal{A}$ to pass the ruler's test is less than $e^{-2\lambda(\varepsilon_\rho \lambda^d - 1)^2}$, which is negligible. Since $\mathcal{R}_d$ rules over $\mathcal{A}$ with negligible probability, the probability that $\varepsilon_\rho \geq \lambda^{-d}$ is negligible. So, we must have $E(\varepsilon_\rho) \leq \lambda^{-d} + \mathsf{negl}(\lambda)$. We deduce then that $\Pr[\mathcal{E}(q, y, \beta \oplus b) = \beta] - \frac{1}{2} = \mathcal{O}(\lambda^{-d})$ for a random public key q, a random K, and a random β.

We apply this result for every d and obtain that $\Pr[\mathcal{E}(q, \mathsf{PKEnc}_q(\beta; \rho_{\mathcal{C}})) = \beta] - \frac{1}{2}$ is negligible for any β and any polynomial $\mathcal{E}$. Therefore, the cryptosystem is secure. □

Extension to adaptive adversaries. Clearly, this result extends to adaptive adversaries but with a cryptosystem replaced by a public cryptography protocol [32].

Namely, the encryption becomes interactive, but it can still be carried out with public information. I.e., Alice starts with a message m and a public key; Bob starts with a secret key and ends with m, but m remains private. We conclude this part as follows.

Theorem 8. *Consider a keyed primitive $\mathcal{F}$. Assume that $\mathcal{F}$ is* HOT-KI*-secure for a given distribution D. If there exists an adversary $\mathcal{A}$ in the* RHOT *game with advantage $1-\mathsf{negl}(\lambda)$ for the class of D-permissive rulers, then we can construct a public cryptography protocol based on $\mathcal{F}$ and $\mathcal{A}$.*

The Minicrypt case. Using the Minicrypt hypothesis [26] that public-key cryptosystems do not exist but one-way functions do, security in the known-relations model implies security in the chosen-relations model with permissive rulers in the following two cases:

- in a weak form in the sense that it is ensured that no adversary has an advantage $1 - \mathsf{negl}(\lambda)$;
- in a uniform form in the sense that it is ensured that no adversary has a uniform advantage $1/\mathsf{Poly}(\lambda)$ (due to Lemma 2).

Assuming that doing public-key cryptography from symmetric cryptography is impossible (which is supported by Rudich [32]), we obtain that known-input security implies permissive chosen-input (weak or uniform) security, for all $\mathcal{F}$ based on symmetric cryptography. If we do have known-input security, for any CI adversary, there must be a permissive ruler making its advantage negligible.

3.2 Concrete Constructions of Cryptosystems

As a nice example of application of Lemma 7, we show that we can obtain the ElGamal cryptosystem by this result.

Let a family $(G, g, n, h)_\lambda$ of tuples, with G being a finite Abelian group, g being an element of prime order n, and h being a Boolean function such that $\Pr[h(g^x) = 0] - \frac{1}{2}$ is negligible when $x \in \{0, \ldots, n-1\}$. We assume the following facts: 1. there exist algorithms which are polynomially bounded and compute products and inverses in G; 2. $\log n$ is polynomially bounded; 3. there is a polynomially bounded algorithm to compute $h(x)$, for x in the subgroup $\langle g \rangle$ generated by g.

We define $\mathcal{F}$ as follows: Gen picks K and defines $b = h(g^K)$. Then, $f_K(q) = q^K$. We consider the uniform distribution D over $\langle g \rangle$. A chosen input attack could select $q = g$ and deduce b from the response $f_K(q)$ but this can be ruled out by the rule saying that $q = g$ is not allowed (indeed, it does not look like random). Later, we will randomize q so that it cannot be detected by permissive rules. In relation to Section 3.1, we have the following result.

Lemma 9. *If the decisional Diffie-Hellman problem is hard in $(G, g, n)_\lambda$, then f is* HOT*-secure against D-KI attacks.*

Proof. In KI attack settings, the adversary gets random (q_i, q_i^K) pairs.

When n is prime, it reduces to the case where a single pair is given. This is so since the adversary could sample other pairs with the same distribution by simply raising the unique pair to some random power. If the decisional Diffie-Hellman problem is hard in G, then —given (g, q, q^K)— it is hard to infer $h(g^K)$. So, f resists to D-known-relation attacks. □

After referring to known-input security, we now elaborate on chosen-input attack. A CI adversary choosing $q = g^\rho$ with ρ random is indistinguishable from the KI case. However, such adversary can easily compute $b = h(y^{\frac{1}{\rho}})$ given $y = q^K$. So, we are in the situation where we can construct a public-key cryptosystem, following Lemma 7. (The public key is one q. The secret key is ρ such that $q = g^\rho$. To encrypt β, we pick a random K and compute both $y = q^K$ and $e = \beta \oplus b$. To decrypt (y, e), we compute $e \oplus h(y^{\frac{1}{\rho}})$.) So, we obtain a kind of ElGamal cryptosystem [17], or some hybrid construction based on the Diffie-Hellman key exchange [13].

4 Related-Key Security

We apply here our approach to model (in)security for the case of related-key attacks. We first present previous approaches to this. We then extend our model to black-box relations to support related-key attacks. Next, we show that we cannot reach security in this model for the uniform distribution among all permutations over $\mathcal{K}$. Finally, we discuss on obfuscation.

Similar results would hold for Key-Dependent Input (KDI) security. For this, we would define $f_K(\varphi, x) = \mathsf{Enc}_K(\varphi(K))$ (See Appendix B). Also, these are special cases for leakage-resilience as defined by $f_K(\varphi, x) = \varphi(K, x)$.

4.1 The Black-Box Approach

In this section, we consider a black-box model, in which relations are provided by the adversary in terms of a black-box oracle access.

Definition 10 (Black-box adversary, black-box ruler). *A black-box adversary $\mathcal{A}$ for the* RHOT *game, denoted as a* BBRHOT-adversary*, is an adversary who provides relations φ_i in terms of a stateless oracle access. The challenger (and the ruler) can freely query each oracle defined by the adversary. A primitive $\mathcal{F}$ is* BBRHOT-PCI-secure for D *if for any CI-adversary there is a D-permissive ruler making the advantage negligible.*

We define $\mathcal{F}$ by $f_K(\varphi, x) = e_{\varphi(K)}(x)$ for a keyed function e and $b = b(K)$ for a nonzero linear function b. The domain $\mathcal{D}$ of (φ, x) queries is $\mathcal{S}_\mathcal{K} \times \mathcal{M}$, the product of the set $\mathcal{S}_\mathcal{K}$ of permutations φ over $\mathcal{K}$ and the domain $\mathcal{M}$ of x. We show that BBRHOT-PCI-security for the uniform distribution over $\mathcal{D}$ is not possible. For this, we show that there is a CI-adversary which can break any keyed function e in the HOT game, and that this adversary passes any permissive ruler in the

black-box model. That is, by extending the Harris attack [22,23], we mount a key-bit recovery attack in the black-box relation model.

Theorem 11. *Given a keyed function $e_K(x)$ and a nonzero linear function $b(K)$ over the domain $\mathcal{K}$ of K and the domain $\mathcal{M}$ of x, we define a keyed primitive $\mathcal{F}$ via $(K, b) \leftarrow \mathsf{Gen}$ and $f_K(\varphi, x) = e_{\varphi(K)}(x)$, with $K \in \mathcal{K}$, $b = b(K)$, and $\varphi \in \mathcal{S}_{\mathcal{K}}$ a permutation over $\mathcal{K}$. We assume that $(k \mapsto e_k(x))_{x \in \mathcal{M}}$ is a collision-resistant family of functions over $\mathcal{K}$.*[3]

If one-way functions exist, there is a non-adaptive polynomially bounded PCI-adversary $\mathcal{A}$ in the BBRHOT *game for the uniform distribution D over $\mathcal{D}$, $\mathcal{A}$ having a uniform advantage of $1 - \mathsf{negl}(\lambda)$.*

This theorem shows that some attacks exist for this distribution D. They cannot be detected by analyzing the chosen-relations in a black-box manner. Therefore, permissive related-key security is not possible in a black-box setting.

Proof. Lemma 12 below shows that there is one adversary $\mathcal{A}$, using a single query (φ, x), with uniform advantage in the HOT game being $\frac{1}{2} - \mathsf{negl}(\lambda)$. So, we can use the Amplification Lemma 2 with $k = \lambda$ iterations. Lemma 12 further says that for $\rho_{\mathcal{C}}$ fixed, φ selected by $\mathcal{A}$ is a PRP while x is uniform and independent. The amplification uses independent queries with same distribution. So, a permissive ruler in the black box model cannot rule over the amplified adversary. □

Lemma 12. *We assume a keyed primitive e such that $(k \mapsto e_k(x))_{x \in \mathcal{M}}$ is a collision-resistant family of functions over $\mathcal{K}$. Let b be a nonzero linear function from $\mathcal{K}$ to $\{0, 1\}$. If one-way functions exist, there is a polynomially bounded adversary $\mathcal{A}$ using a single query (φ, x) in the* HOT *game with uniform advantage $\frac{1}{2} - \mathsf{negl}(\lambda)$.*

Furthermore, φ and x are independent, φ is a PRP, and x is uniform.

Proof. Essentially, we construct an adversary by using the Harris [22,23] attack, but we obfuscate the relation and the leaking information. This latter adversary is using a single CI (φ, x) and a linear bit $b(K)$. This simplifies the HOT game as follows:

Game $\mathsf{HOT}_{\mathcal{F}}(\mathcal{A})$:
1: initialize $\mathcal{A}$ with some random coins ρ
2: set $K \in \mathcal{K}$ at random
3: $(\varphi, x) \leftarrow \mathcal{A}(\rho)$
4: $y \leftarrow e_{\varphi(K)}(x)$
5: $\gamma \leftarrow \mathcal{A}(y; \rho)$
6: return $\gamma \oplus b(K)$

The uniform advantage $\mathsf{UAdv}_{\mathsf{HOT}_{\mathcal{F}}}(\mathcal{A})$ of $\mathcal{A}$ is $\mathsf{UAdv}_{\mathsf{HOT}}(\mathcal{A}) = \min_K 2\Pr[\gamma = b(K)] - 1$.

We define $g_x(K) = e_K(x)$. Let ε be a fixed vector such that $b(\varepsilon) = 1$.

[3] I.e., given a random x, it is hard to find $k \neq k'$ such that $e_k(x) = e_{k'}(x)$. This could be the case, e.g., when $\mathcal{M}$ is much larger than $\mathcal{K}$.

Given $\sigma \in \mathcal{S}_\mathcal{K}$, a Boolean function F over $\mathcal{M}$, and $x \in \mathcal{M}$, we define $\mathsf{bit}_{\sigma,F,x}(K) = F \circ g_x \circ \sigma(K)$, a Boolean function extracting a bit of K and

$$\varphi_{\sigma,F,x}(K) = \begin{cases} \sigma(K') \text{ for } K' \in \{K, K \oplus \varepsilon\} \text{ s.t.} \\ \qquad \mathsf{bit}_{\sigma,F,x}(K') = b(K) & \text{if } \mathsf{bit}_{\sigma,F,x}(K) \neq \mathsf{bit}_{\sigma,F,x}(K \oplus \varepsilon) \\ \sigma(K) & \text{otherwise} \end{cases}$$

Note that $\varphi_{\sigma,F,x}$ is a permutation. Indeed, given $y = \varphi_{\sigma,F,x}(K)$ we can recover the pair $\{K, K \oplus \varepsilon\}$ by computing $\sigma^{-1}(y)$ and its XOR to ε. Then, we can figure out whether $\mathsf{bit}_{\sigma,F,x}(K) = \mathsf{bit}_{\sigma,F,x}(K \oplus \varepsilon)$ by computing the two bits. If they are equal, then $K = \sigma^{-1}(y)$. If they are different, K is the only one such that $\mathsf{bit}_{\sigma,F,x}(\sigma^{-1}(y)) = b(K)$.

Let σ_{ρ_1} be a pseudorandom permutation (PRP) over $\mathcal{K}$, and let F_{ρ_2} be a Boolean pseudorandom function (PRF) with domain $\mathcal{M}$.

We define the adversary $\mathcal{A}$ for the game $\mathsf{HOT}_\mathcal{F}$. $\mathcal{A}$ picks ρ_1, ρ_2, x from ρ and defines $\varphi = \varphi_{\sigma_{\rho_1},F_{\rho_2},x}$. The only query made by $\mathcal{A}$ is (φ, x). Then, using the response $y = e_{\varphi(K)}(x) = g_x(\varphi(K))$, we define $\gamma = \mathcal{A}(y; \rho) = F_{\rho_2}(y) = F_{\rho_2} \circ g_x(\varphi(K))$ as the final output.

Since ρ_1 resp. ρ_2 are only used inside σ_{ρ_1} resp. F_{ρ_2} within the algorithm of $\mathcal{A}$, then the computations of σ and F can be outsourced to some oracle and $\mathcal{A}$ needs not ρ any more. Since σ is a PRP and F is a PRF, the outcome of the (polynomially bounded) game is indistinguishable from the resulting outcome if we were to use a random pair (σ, F) with uniform distribution. We can thus make the assumption that σ is a uniformly distributed permutation and that F is a randomly distributed function, and assume that $\varphi_{\sigma,F,x}$ is defined from σ and F instead of σ_{ρ_1} and F_{ρ_2}.

In Lemma 13, we show that for any x, the relation $\varphi_{\sigma,F,x}$ is a PRP. So, φ is a PRP independent from the uniform x. Furthermore, Lemma 13 shows that $\gamma = b(K)$ with probability close to $\frac{3}{4}$ when K is fixed. So, the uniform advantage is close to $\frac{1}{2}$. □

Lemma 13. *Let $(g_x)_{x\in\mathcal{M}}$ be a collision-resistant family of functions over $\mathcal{K}$, $\varepsilon \in \mathcal{M}$, and b be a linear form over $\mathcal{M}$ such that $b(\varepsilon) = 1$. Let σ be a random Boolean permutation and F be a random function on $\mathcal{M}$. Given x, we define $\varphi(K) = \sigma(K')$ where $K' \in \{K, K \oplus \varepsilon\}$ is such that $F \circ g_x \circ \sigma(K') = b(K)$ if $F \circ g_x \circ \sigma(K) \neq F \circ g_x \circ \sigma(K \oplus \varepsilon)$ and $K' = K$ otherwise.*

Given a fixed key x, φ is indistinguishable from a uniformly distributed permutation.

Given $k \in \mathcal{K}$ fixed and x uniformly distributed. $\Pr[F \circ g_x(\varphi(k)) = b(k)] = \frac{3}{4} - \mathsf{negl}(\lambda)$.

Proof. Given x fixed, we consider a distinguisher $\mathcal{R}$ playing with the φ oracle. Let E be the event that $\mathcal{R}$ queries φ with two keys K and K' such that $K \neq K'$, $K \neq K' \oplus \varepsilon$, and $g_x(\sigma(K)), g_x(\sigma(K \oplus \varepsilon)), g_x(\sigma(K')), g_x(\sigma(K' \oplus \varepsilon))$ are not pairwise different. Clearly, this adversary translates to a polynomial algorithm to find collisions on g with success probability $\Pr[E]$. But, by underlying assumptions, this must be negligible. So, we assume that E does not occur in the execution

of $\mathcal{R}$. Let S denote the union of all $\{K, K \oplus \varepsilon\}$ of all K's which are queried by $\mathcal{R}$ to φ. We obtain that $g_x \circ \sigma$ is injective on S.

We say that two permutations π and π' are equivalent if for all K in S, the two unordered pairs $\{\pi(K), \pi(K \oplus \varepsilon)\}$ and $\{\pi'(K), \pi'(K \oplus \varepsilon)\}$ are the same. We note that φ is always equivalent to σ. We will show that if we select σ in a given equivalence class Class and pick F at random, then φ restricted to S will be a uniformly distributed element of Class. Indeed, the ordering of a pair for φ is locally defined by the ordering of σ on the same pair and the values of F related to this pair. In addition to this, if we flip the order for σ and we complement the two related bits in F on this pair, then we obtain the inverse order for φ. (Since $g_x \circ \sigma$ is injective on S, note that the F values to flip are independent from the others.) Therefore the mapping $(\sigma, F) \mapsto \varphi$ is balanced for $\sigma \in \mathsf{Class}$ and F random. So, it is balanced over the permutation set. Therefore, φ is uniformly distributed.

The φ construction is such that $F \circ g_x(\varphi(K)) = b(K)$ when $F \circ g_x \circ \sigma(K) \neq F \circ g_x \circ \sigma(K \oplus \varepsilon)$ and only for half of the K's in the other case. Given a fixed k, let E_k be the event that $g_x(\sigma(k)) = g_x(\sigma(k \oplus \varepsilon))$. Since σ transforms the $(k, k \oplus \varepsilon)$ pair into a random pair of different keys, we have $\Pr[E_k] = p_{\mathsf{coll}}$ where $p_{\mathsf{coll}} = \Pr[g_x(K) = g_x(K') | K \neq K']$ when K and K' are independent and uniformly distributed. If E_k does not occur, the probability over F that $F \circ g_x(\varphi(k)) = b(k)$ corresponds to the case where the pair is mapped by F to different bits or to two bits equal to $b(k)$, so

$$\Pr[F \circ g_x(\varphi(k)) = b(k) | \neg E_k] = \frac{3}{4}$$

Similarly,

$$\Pr[F \circ g_x(\varphi(k)) = b(k) | E_k] = \frac{1}{2}$$

So,

$$\Pr[F \circ g_x(\varphi(k)) = b(k)] = \frac{3}{4}(1 - p_{\mathsf{coll}}) + \frac{1}{2} p_{\mathsf{coll}} = \frac{3}{4} - \frac{1}{4} p_{\mathsf{coll}}$$

□

4.2 On Obfuscation

Theorem 11 relies on obfuscating the Harris attack behind pseudorandom permutations and functions so that no ruler would recognize the structure of the relation in a black-box manner. In this construction, we have $\varphi = \varphi_{\sigma_{\rho_1}, F_{\rho_2}, x}$. When moving to a non-black-box model, relations must be specified in terms of a code which could try to obfuscate the relation as well. Namely, the adversary could provide some code $\mathsf{Obf}(\varphi)$ obfuscated by some algorithm Obf so that there is an execution algorithm Exe such that for all x, $\mathsf{Exe}(\mathsf{Obf}(\varphi), x) = \mathsf{Exe}(\varphi, x)$.

Assuming that $\mathsf{Obf}(\varphi_{\sigma_{\rho_1}, F_{\rho_2}, x})$ and $\mathsf{Obf}(\varphi)$ for φ random cannot be distinguished, then Theorem 8 says that we can construct a public-key cryptosystem based on $\mathcal{F}$ and $\mathsf{Obf}(\varphi_{\sigma_{\rho_1}, F_{\rho_2}, x})$. Namely, a public key would be the obfuscated

relation and the secret key would consist of the ρ values. Since this construction is unlikely to be feasible due to the separation between symmetric cryptography and public-key cryptography, we deduce that for any Obf, there must be a ruler to tell $\mathsf{Obf}(\varphi_{\sigma_{\rho_1},F_{\rho_2},x})$ and $\mathsf{Obf}(\varphi)$ apart.

We could try to obfuscate φ using white-box cryptography [11,12] or any obfuscation mechanism [1,25]. Our result shows that there must be a generic way to defeat these techniques in that case. So, it is likely to be a hard task to find the appropriate ruler.

5 Conclusion

We have formalized security notions in which the adversary tries to win against a challenger while a ruler is watching him. This gave definitions for known-input and permissive chosen-input security. We have shown that a gap between these notions implies a public cryptography protocol construction. As for related-key security, we have shown that the gap exists when providing relations in terms of black-boxes. When removing black-boxes, we deduced that all obfuscation schemes can be defeated by a ruler, or we can construct a public-key cryptosystem from a block cipher, pseudorandom permutations, and the obfuscation scheme, or no known-relation security exists.

Acknowledgements

We thank Jorge Nakahara, Martijn Stam, and Ioana Boureanu for many valuable remarks in earlier versions of this paper.

References

1. B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S.P. Vadhan, K. Yang. On the (Im)possibility of Obfuscating Programs. In *Advances in Cryptology CRYPTO'01*, Santa Barbara, California, U.S.A., Lecture Notes in Computer Science 2139, pp. 1–18, Springer-Verlag, 2001.
2. M. Bellare, D. Cash. Pseudorandom Functions and Permutations Provably Secure against Related-Key Attacks. In *Advances in Cryptology CRYPTO'10*, Santa Barbara, California, U.S.A., Lecture Notes in Computer Science 6223, pp. 666–684, Springer-Verlag, 2010.
3. M. Bellare, T. Kohno. A Theoretical Treatment of Related-Key Attacks: RKA-PRPs, RKA-PRFs, and Applications. In *Advances in Cryptology EUROCRYPT'03*, Warsaw, Poland, Lecture Notes in Computer Science 2656, pp. 491–506, Springer-Verlag, 2003.
4. M. Bellare, P. Rogaway. The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. In *Advances in Cryptology EUROCRYPT'06*, St. Petersburg, Russia, Lecture Notes in Computer Science 4004, pp. 409–426, Springer-Verlag, 2006.
5. D.J. Bernstein. Private communication. 2010.

6. E. Biham. New Types of Cryptanalytic Attacks Using related Keys. In *Advances in Cryptology EUROCRYPT'93*, Lofthus, Norway, Lecture Notes in Computer Science 765, pp. 398–409, Springer-Verlag, 1994.
7. E. Biham. New Types of Cryptanalytic Attacks Using Related Keys. *Journal of Cryptology*, vol. 7, pp. 229–246, 1994.
8. E. Biham. How to Decrypt or even Substitute DES-Encrypted Messages in 2^{28} steps. *Information Processing Letters*, vol. 84, pp. 117–124, 2002.
9. J. Black, P. Rogaway, T. Shrimpton. Encryption-Scheme Security in the Presence of Key-Dependent Messages. In *Selected Areas in Cryptography'02*, St. John's, Newfoundland, Canada, Lecture Notes in Computer Science 2595, pp. 62–75, Springer-Verlag, 2002.
10. H. Chernoff. A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the sum of Observations. *Annals of Mathematical Statistics*, vol. 23 (4), pp. 493-507, 1952.
11. S. Chow, P.A. Eisen, H. Johnson, P.C. van Oorschot. White-Box Cryptography and an AES Implementation. In *Selected Areas in Cryptography'02*, St. John's, Newfoundland, Canada, Lecture Notes in Computer Science 2595, pp. 250–270, Springer-Verlag, 2002.
12. S. Chow, P.A. Eisen, H. Johnson, P.C. van Oorschot. A White-Box DES Implementation for DRM Applications. In *Security and Privacy in Digital Rights Management DRM'02*, Washington, DC, USA, Lecture Notes in Computer Science 2696, pp. 1–15, Springer-Verlag, 2003.
13. W. Diffie, M. E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, vol. IT-22, pp. 644–654, 1976.
14. W. Diffie, M.E. Hellman. Exhaustive Cryptanalysis of the NBS Data Encryption Standard. *Computer*, vol. 10, pp. 74–84, 1977.
15. D. Dolev, C. Dwork, M. Naor. Non-Malleable Cryptography. In *Proceedings of the 23rd ACM Symposium on Theory of Computing*, New Orleans, Louisiana, U.S.A., pp. 542–552, ACM Press, 1991.
16. D. Dolev, C. Dwork, M. Naor. Non-Malleable Cryptography. *SIAM Journal of Computing*, vol. 30, pp. 391–437, 2000.
17. T. ElGamal. A Public-key Cryptosystem and a Signature Scheme based on Discrete Logarithms. *IEEE Transactions on Information Theory*, vol. IT-31, pp. 469–472, 1985.
18. P. Farshim, K. Paterson, M. Albrecht, G. Watson On Cipher-Dependent Related-Key Attacks in the Ideal-Cipher Model. In *Fast Software Encryption'11*, Lyngby, Denmark, Lecture Notes in Computer Science 6733, pp. 128–145, Springer-Verlag, 2011.
19. D. Goldenberg, M. Liskov. On Related-Secret Pseudorandomness. In *Theory of Cryptography TCC'10*, Zürich, Switzerland, Lecture Notes in Computer Science 5978, pp. 255–272, Springer-Verlag, 2010.
20. I. Haitner, T. Holenstein. On the (Im)Possibility of Key Dependent Encryption. In *Theory of Cryptography TCC'09*, San Fransisco CA, USA, Lecture Notes in Computer Science 5444, pp. 202–219, Springer-Verlag, 2009.
21. S. Halevi, H. Krawczyk. Security under Key-Dependent Inputs. In *14th ACM Conference on Computer and Communications Security*, Alexandria VA, USA, pp. 466–475, ACM Press, 2007.
22. D.G. Harris. Generic Ciphers are More Vulnerable to Related-Key Attacks than Previously Thought. Presented at WCC'09.
23. D.G. Harris. Critique of the Related-Key Attack Concept. *Design, Codes, and Cryptography*, vol. 59, pp. 159–168, 2011.

24. W. Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, vol. 58, pp. 13–30, 1963.
25. D. Hofheinz, J. Malone-Lee, M. Stam. Obfuscation for Cryptographic Purposes. In *Theory of Cryptography TCC'07*, Amsterdam, The Netherlands, Lecture Notes in Computer Science 4392, pp. 214–232, Springer-Verlag, 2007.
26. R. Impagliazzo. A Personal View of Average-Case Complexity. In *Structure in Complexity Theory Conference SCT'95*, Minneapolis, MN, USA, pp. pp. 134–147, IEEE, 1995.
27. J. Kelsey, B. Schneier, D. Wagner. Key-Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES. In *Advances in Cryptology CRYPTO'96*, Santa Barbara, California, U.S.A., Lecture Notes in Computer Science 1109, pp. 237–251, Springer-Verlag, 1996.
28. L.R. Knudsen. Cryptanalysis of LOKI91. In *Advances in Cryptology AUSCRYPT'92*, Gold Coast, Queensland, Australia, Lecture Notes in Computer Science 718, pp. 196–208, Springer-Verlag, 1993.
29. S. Lucks. Ciphers Secure against Related-Key Attacks. In *Fast Software Encryption'04*, Delhi, India, Lecture Notes in Computer Science 3017, pp. 359–370, Springer-Verlag, 2004.
30. M. Naor, M. Yung. Public-key Cryptosystems Provably Secure against Chosen Ciphertext Attacks. In *Proceedings of the 22nd ACM Symposium on Theory of Computing*, Baltimore, Maryland, U.S.A., pp. 427–437, ACM Press, 1990.
31. K. Pietrzak. Composition Implies Adaptive Security in Minicrypt. In *Advances in Cryptology EUROCRYPT'06*, St. Petersburg, Russia, Lecture Notes in Computer Science 4004, pp. 328–338, Springer-Verlag, 2006.
32. S. Rudich. The Use of Interaction in Public Cryptosystems. In *Advances in Cryptology CRYPTO'91*, Santa Barbara, California, U.S.A., Lecture Notes in Computer Science 576, pp. 242–251, Springer-Verlag, 1992.

A On Related-Key Security

A.1 Some Attacks to be Ruled Over

We list here some non-dedicated attacks in related-key settings. The purpose of this list is to keep in mind some necessary rules to be considered when developing a feasible security model.

In a folklore attack, the adversary uses ℓ queries (φ_i, x_i), $i = 0, \ldots, \ell - 1$, where ℓ is the key length and $\varphi_i(K) = K \mathsf{\ AND\ } 1^{\ell-i}0^i$. That is, $\varphi_i(K)$ consists of the first $\ell - i$ bits of K padded with zeroes. Clearly, by getting one known plaintext/ciphertext pair per black-box, an adversary can recover all bits of K sequentially by exhaustive search with complexity $\mathcal{O}(\ell)$.

In 2003, Bellare and Kohno [3] proposed another similar attack in this model. Essentially, they use ℓ related keys again (ℓ being the key length). The permutation φ_i for $i > 0$ was defined as follows: if the ith bit of x is 1, then $\varphi_i(K)$ is obtained from K by flipping the least significant bit, otherwise $\varphi_i(K) = K$. (Assume that the least significant bit $\mathsf{lsb}(K)$ is 0.) Additionally, $\varphi_0(K) = K$. In a chosen plaintext attack, one could get $y_i = \mathsf{Enc}_{K_i}(x)$ for all i. If $y_i \neq y_0$, it means that the ith bit of K_0 is 1. Clearly, we recover again all bits in linear time.

Recently, Harris [22,23] proposed another attack which is similar to the Bellare-Kohno [3] attack. Here, $\varphi_{i,x}(K)$ is either K or $K \oplus e_i$ for $e_i = 0^{\ell-i-1}10^i$ (i.e., $K \oplus e_i$ is K with its ith bit flipped), depending on some condition related to the least significant bits (lsb) $y = \mathsf{lsb}(\mathsf{Enc}_K(x))$ and $y' = \mathsf{lsb}(\mathsf{Enc}_{K\oplus e_i}(x))$. Namely, if $y = y'$, then $\varphi_{i,x}(K) = K$. Otherwise, either y or y' is equal to the ith bit of K. If this is y then $\varphi_{i,x}(K) = K$. Otherwise, $\varphi_{i,x}(K) = K \oplus e_i$. It is not hard to realize that this defines a permutation $\varphi_{i,x}$. The nice property is that $\mathsf{lsb}(\mathsf{Enc}_{\varphi_{i,x}(K)}(x))$ equals the ith bit of K with probability $\frac{3}{4}$ over the random choice of x. So, by statistical analysis, we can infer every bit by using several related keys.

Even more recently, Bernstein [5] proposed a generic related-key distinguisher using a single related key. The proposed relation is $K' = \mathsf{Enc}_K(0)$. Although it is not a permutation, one can admit that it is still a one-way transformation for which finding collisions is hard. The attack consists of encrypting 0 with key K (say $y_0 = \mathsf{Enc}_K(0)$) and any plaintext x with key K' (say $y = \mathsf{Enc}_{K'}(x)$) then comparing $\mathsf{Enc}_{y_0}(x)$ with y. The distinguisher has essentially an advantage of 1. So far, it is not clear how this attack can be turned into a key recovery attack.

All these attacks could be seen as devastating in theory although they do not seem to mean any endemic weakness for any cipher. What is in common between all these attacks is that they are generic and they use some intricate relations. Consequently, these relations must be explicitly forbidden by ad hoc rules, i.e., arbiters should rule them over.

If the set of authorized permutations makes it possible to define r related keys, one could use a tradeoff attack as proposed by Biham [8]. Essentially, one could collect $y_i = \mathsf{Enc}_{K_i}(x)$ for all i then perform a multi-target exhaustive search to recover one key out of r. This works with complexity $\mathcal{O}(2^\ell/r)$. For $r = 2^{\frac{\ell}{2}}$, this is $\mathcal{O}(2^{\frac{\ell}{2}})$. For instance, if the transformations $\varphi(x) = x \oplus c$ are allowed for all c, we can mount a key recovery attack against any ℓ-bit key cipher with complexity $\mathcal{O}(2^{\frac{\ell}{2}})$. As another example, if only the transformation $\varphi(x) = x + 1 \bmod 2^\ell$ and its iterations are allowed, we obtain the same result. In general, allowing r permutations and related keys makes it possible to use the previous attack with space complexity $\mathcal{O}(r)$ and time complexity $\mathcal{O}(2^\ell/r)$. Fortunately, this attack has a super-polynomial complexity. So, in practice, we are not threatened by this attack.

A.2 Previous Approaches for Related-Key Security

Due to the existence of related-key attacks breaking all ciphers by using special relations (see Appendix A.1), sound security models for related-key security must rule over attackers using these relations. Bellare-Kohno [3] devised an exhaustive list of criteria including such allowed relations but his criteria only work in the ideal cipher model. These relations must be in a set D such that the following aspects are the case.

- **Output unpredictability**: For any subset P of a (polynomially) large set D and for any set X of keys of (polynomially) bounded size, the probability over $K \in_U \mathcal{K}$ that $\{\varphi(K); \varphi \in P\} \cap X \neq \emptyset$ is negligible.

– **Collision resistance**: For any subset P of a (polynomially) large set D, the probability over $K \in_U \mathcal{K}$ that $\#\{\varphi(K); \varphi \in P\} < \#P$ is negligible.

Output unpredictability rules over attackers working with functions which cancel too many bits. Since $\varphi(K) = K$ for many relations, collision resistance eliminates the Bellare-Kohno attack and the Harris attack (see Appendix A.1). So, these criteria eliminate *all threats* except the Bernstein [5] attack.[4] (See Appendix A.1.) Bellare and Kohno prove that these two criteria are sufficient to prove security in the $\mathsf{PRP\text{-}RKA}^D_{\mathsf{Enc}}$ game *in the ideal cipher model*. This model by itself discards Bernstein's attack, since relations cannot call the cipher itself. What is satisfactory about their approach is that all polynomial attacks mentioned in Appendix A.1 are eliminated. As per [3], therein we could indeed show that some secure block ciphers exist. What is not satisfactory about their approach is that all dedicated attacks in the literature are also eliminated because they attack a cipher which is not in the ideal cipher model.

In [23], Harris proposed to define related-key security in the standard model but for tweakable encryption, in which the adversary would have to commit to a set of allowed relations before he learns which tweak τ is being used. Of course, this set must be polynomially bounded. Otherwise, the adversary could decide to allow $\varphi_{i,x,\tau}$ for all τ. Still, the relevance of this model to practice is debatable.

In [18], relations can invoke Enc and Enc^{-1} but there is the extra condition, called *oracle independence*. It means that the adversary shall not produce (φ_1, x) and φ_2 such that $(\varphi_1(K), x)$ was queried to Enc or Enc^{-1} during the computation of $\varphi_2(K)$. This condition rules over the Bernstein attack (due to $\varphi_1(K) = K$, $x = 0$, and $\varphi_2(K) = \mathsf{Enc}_K(0)$). It also eliminates the improved Harris attack (due to $\varphi_1(K) = K \oplus e_i$ and $\varphi_2 = \varphi_{i,x}$). However, oracle independence inherently relies on the ideal cipher model: any instantiation may hide the fact that $(\varphi_1(K), x)$ is queried during the computation of $\varphi_2(K)$; this is done by not querying it but doing the computation locally instead.

In order to rule over the above improvement of the Harris attack in the standard model (i.e., with no cipher oracle), we shall use our security model based on rulers.

A.3 Using Rulers for Related-Key Security

The Bellare-Kohno [3] conditions for output unpredictability and collision resistance could also cast as a class Jury of rulers. Indeed, we make Jury contain two types of rulers, as follows.

– For output unpredictability:
For each $k \in \mathcal{K}$, and for each integers d and i, we define $\mathcal{R}_{k,d,i}$ as follows. Let

[4] What is in common between the Harris attack [22,23] and the Bernstein one [5] is that the relation is defined using the encryption itself. We could consider a related-key attack in the ideal cipher model (a.k.a. the Shannon model), where the encryption/decryption would be given as an oracle-access. Having encryption/decryption circumvented in an oracle makes it possible to prevent from using it in the definition of elements of D.

φ_i be the relation in the ith query by $\mathcal{A}$. The ruler ought to make statistics to estimate whether $\Pr[\varphi_i(K) = k] \geq \lambda^{-d}$ (taken over random K), and it ought to reject if this holds. (If there is no ith query, just output 0.)

– For collision resistance:
  For each d, i, j such that $i < j$, we define $\mathcal{R}_{d,i,j}$ as follows. Let φ_i be the relation in the ith query of $\mathcal{A}$ and φ_j be the relation in jth query of $\mathcal{A}$. The ruler ought to make statistics to estimate whether $\Pr[\varphi_i(K) = \varphi_j(K)] \geq \lambda^{-d}$ (taken over random K) and it ought to reject, if this holds. (If there is no ith query, just output 0.)

Namely, for output unpredictability, using $n = \lambda^{2d+1}$ samples $K_1, \ldots, K_n$, the ruler $\mathcal{R}_{k,d,i}$ aborts if the number of js such that $\varphi_i(K_j) = k$ goes beyond $\frac{n}{2}\lambda^{-d}$. By using the Hoeffding bound [24], we obtain that —if $\Pr[\varphi_i(K) = k] \geq \lambda^{-d}$— then, with probability at most $e^{-\frac{\lambda}{2}}$, the ruler $\mathcal{R}_{k,d,i}$ does not abort. If $\Pr[\varphi_i(K) = k] \leq \frac{1}{4}\lambda^{-d}$, the ruler aborts with probability at most $e^{-\frac{\lambda}{8}}$. If the set of relations satisfies output unpredictability, then we know that $\Pr[\varphi_i(K) = k] = \mathcal{O}(\lambda^{-d-1})$. So, there is a λ_0 such that for all $\lambda > \lambda_0$ and we have $\Pr[\varphi_i(K) = k] < \frac{1}{4}\lambda^{-d}$. Therefore, $\mathcal{R}_{k,d,i}$ aborts with negligible probability. This holds for all k, d, i. Conversely, if the set does not satisfy output unpredictability, there must be some k, d, and i such that $\Pr[\varphi_i(K) = k] \geq \lambda^{-d}$ for infinitely many λ's. So, $\mathcal{R}_{k,d,i}$ aborts with a probability which is not negligible.

The same arguments hold for collision resistance.

We note that all these rulers are polynomially bounded and permissive. In this fashion, we rule over most of polynomial attacks from Appendix A.1 except the Bernstein one. To rule over the Bernstein attack, we can use a ruler $\mathcal{R}$ who looks whether there exists an (i, j)-pair such that $y_i = \mathcal{A}_1(c_j, K; \rho)$ (i.e., one encryption-result equals one related key). This ruler rejects if this is the case. Again, this is also polynomially bounded and permissive. So, we can rule over all polynomial attacks from Appendix A.1 without using the Shannon model.

B Related-Key Model versus Key-Dependent Input Model

As we can see, the problem with the Harris attack [22,23] lies in the way the adversary makes the relation depends on the message to encrypt. Somehow, this is a reminiscent of the key-dependent input (KDI) model. In the KDI model, a query φ returns $\mathsf{Enc}_K(\varphi(K))$.

Indeed, the Harris attack translates to our model in a straightforward way. We define $\varphi_i(K) = x$ as the smallest number such that $\mathsf{lsb}(\mathsf{Enc}_K(x))$ is the ith bit of K. By making the φ_i query to a KDI challenger, we obtain a ciphertext whose least significant bit is equal to the ith bit of K. This was already noticed in Black-Rogaway-Shrimpton [9].

This could be even worse: when the key is smaller than the message block, the $\varphi(K) = \mathsf{Enc}_K^{-1}(K)$ query would yield K, as noticed by Halevi-Krawczyk [21]. They further observed that no deterministic encryption can be KDI-secure with

respect to every set of allowed relations of cardinality 1. This is essentially due to the Bernstein attack: setting $\varphi(K) = \mathsf{Enc}_K(0)$, we can query φ, 0 and its result and check consistency of the outputs to mount a distinguisher.

Halevi-Krawczyk [21] then showed that for any well-spread function φ (i.e. preimages are not too big), we can construct a deterministic encryption which is KDI-secure with respect to the class $\{\varphi\}$. We note that the well-spread condition reminds our previous condition on colliding relations.

Halevi-Krawczyk [21] observed that we can achieve KDI-secure deterministic encryption in the ideal cipher model by preventing key-dependent input functions to depend on the ideal cipher. This is the same situation as in the related-key model in Bellare-Kohno [3].

As we can see, the related-key attacks and key-dependent input attacks share similar properties. Of course, we could combine them and propose a more general framework. In this paper, we were rather inspired by the results on KDI-security and want to see how to address related-key attacks.

In a recent result, Haitner-Holenstein [20] proved that if relations are treated as black-boxes, there is no KDI-secure encryption based on a one-way permutation. We took this approach and look at what happens if related-key permutations were treated like black-boxes.