Security Analysis of PKMS2

Pattern-Key-Multi-Segment-Multi-Standard

System and method for securing multiple data segments having different lengths using pattern keys having multiple different strengths

US Patent: 8744078 B2

Secure Channels Inc.

Security Analysis of PKMS2

Jonathan Katz

October 2, 2017

1 Overview

This document provides proofs of security and security analysis for a two-layer variant of PKMS2 that, from here on, we simply refer to as *the PKMS2 scheme*. At a high level, we show the following results:

1. Even if the cipher used in the first layer is insecure (e.g., has an arbitrary, unknown backdoor), the PKMS2 scheme remains secure as long as the cipher used in the *second layer* is secure.
2. Assuming the ciphers used in both layers are secure, the effective key length of the PKMS2 scheme is larger than the key length for either cipher.
3. Security against message-recovery attacks improves due to the use of segmentation.

We begin in Section 2 by specifying the details of the PKMS2 scheme considered here. Then, in Section 3, we formally define the notions of security corresponding to the results above. Proofs and analysis appear in the remaining sections.

2 Formal Specification of the PKMS2 Scheme

2.1 Background

We begin by defining standard CBC-mode encryption relative to a block cipher $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$. This algorithm takes as input a key $k \in \{0,1\}^\kappa$ and a message $M = m_1 \| \cdots \| m_\ell$ with $m_i \in \{0,1\}^n$. (We assume for simplicity that the message length is a multiple of the block length; padding can be used if this is not the case.) It outputs a ciphertext $c_0 \| c_1 \| \cdots \| c_\ell$ computed as follows:

1. Choose uniform $c_0 \in \{0,1\}^n$.
2. For $i = 1$ to ℓ do:
   - $c_i := F_k(c_{i-1} \oplus m_i)$.
3. Output the ciphertext $c_0 \| c_1 \| \cdots \| c_\ell$.

We denote the above by $C \leftarrow \mathsf{CBC}_k^F(M)$, where $C = c_0 \| \cdots \| c_\ell$.

It will also be convenient to define a cascaded version of CBC-mode encryption relative to two block ciphers F, G. This algorithm takes as input a pair of keys k_1, k_2 and a message $M = m_1 \| \cdots \| m_\ell$ as above, and outputs a ciphertext $c_{-1} \| c_0 \| c_1 \| \cdots \| c_\ell$ computed as follows:

1. $I \leftarrow \mathsf{CBC}^F(M)$.
2. $C \leftarrow \mathsf{CBC}^G(I)$.
3. Output C.

We denote the above by $C \leftarrow 2\mathsf{CBC}^{F,G}_{k_1,k_2}(M)$. Note that the ciphertext is 2 blocks longer than the plaintext.

2.2 The PKMS2 Scheme

The scheme described here corresponds to a two-layer variant of PKMS2, where segmentation is used in each layer.

Fix two block ciphers $F : \{0,1\}^{\kappa_1} \times \{0,1\}^n \to \{0,1\}^n$ and $G : \{0,1\}^{\kappa_2} \times \{0,1\}^n \to \{0,1\}^n$, where κ_1, κ_2 denote their respective key lengths[1] and n denotes their (common) block length.[2] We consider a version of the PKMS2 scheme in which two layers of "segmented" cascaded CBC-mode encryption are applied, with F used in the first layer and G used in the second layer.

The PKMS2 scheme is parameterized by a value s indicating the number of segments. The encryption algorithm takes as input a list of keys $(k_1^1, \ldots, k_1^s, k_2^1, \ldots, k_2^s)$, where $k_1^i \in \{0,1\}^{\kappa_1}$ and $k_2^i \in \{0,1\}^{\kappa_2}$ for all i, along with an ℓ-block message M. It generates a ciphertext as follows:

1. First, $\ell_1, \ldots, \ell_s$ are chosen such that $\ell_i \in \{\lfloor \ell/s \rfloor, \lceil \ell/s \rceil\}$ and $\sum_i \ell_i = \ell$. (We are agnostic as to precisely how this is done, and several options are possible. But we assume some deterministic, publicly known method is fixed in advance.)
2. Parse M as $M = M^1 \| \cdots \| M^s$ with $|M^i| = \ell_i \cdot n$.
3. For $i = 1, \ldots, s$ do:
   - $C^i \leftarrow 2\mathsf{CBC}^{F,G}_{k_1^i,k_2^i}(M^i)$.
4. Output the ciphertext $C^1 \| \cdots \| C^s$.

We remark that for short messages it is possible to set some of the ℓ_i to 0.

3 Security Definitions and Preliminaries

We consider two different settings: one where G is assumed to be secure but F is not, and another where both F and G are assumed to be secure. In the first setting we consider only the standard security notion of indistinguishability. In the second setting we consider two different security notions; in addition to indistinguishability, we also study a notion of security against message-recovery attacks.

To obtain concrete security bounds for an encryption scheme, it is necessary to model block ciphers as *ideal ciphers*; that is, to treat them as truly random, keyed permutations chosen as

[1] F and G can be viewed as allowing a choice among multiple ciphers (of the same block length) by increasing the key length accordingly.

[2] We assume the ciphers have the same block length for simplicity, and also because this is the most likely situation in practice.

part of the security experiment. By doing so, we ensure that the only way an attacker can learn information about them is via explicit queries to oracles for the algorithm, and so we can consider computationally unbounded attackers making a bounded number of oracle queries.

3.1 Fallback Security

Here we model G as an ideal cipher, and say that PKMS2 has *fallback security* if it is secure—in the sense of indistinguishability—regardless of F. In particular, F can be arbitrary (subject to being a keyed permutation), and we make no assumptions whatsoever about its security.

Formally, for an attacker A and an encryption scheme $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ based on G, define

$$\mathsf{Adv}^{\mathsf{RoR}}_{A,\Pi} = \left|\Pr_{k,G}[A^{\mathsf{Enc}_k(\cdot),G(\cdot,\cdot)} = 1] - \Pr_{G}[A^{\$(\cdot),G(\cdot,\cdot)} = 1]\right|,$$

where $\$(M)$ outputs a random string (independent of G) whose length is that of ciphertexts output by $\mathsf{Enc}_k(M)$, and we allow queries to both G and G^{-1}. We then define $\mathsf{Adv}^{\mathsf{RoR}}_{\Pi}(q_e, \ell, q_G) = \max_A\{\mathsf{Adv}^{\mathsf{RoR}}_{A,\Pi}\}$, where the maximum is taken over all attackers requesting encryption of at most q_e messages containing (in total) at most ℓ blocks of plaintext, and making at most q_G queries to G.

3.2 Indistinguishability

We can extend the above definition and model both F and G as (independent) ideal ciphers. Let A be an attacker and let $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ be an encryption scheme that depends on F, G. Define

$$\mathsf{Adv}^{\mathsf{RoR}}_{A,\Pi} = \left|\Pr_{k,F,G}[A^{\mathsf{Enc}_k(\cdot),F(\cdot,\cdot),G(\cdot,\cdot)} = 1] - \Pr_{F,G}[A^{\$(\cdot),F(\cdot,\cdot),G(\cdot,\cdot)} = 1]\right|,$$

where $\$(\cdot)$ is defined as in the previous section (and is independent of F, G), and we allow queries to both F and F^{-1} (and analogously for G). We then define $\mathsf{Adv}^{\mathsf{RoR}}_{\Pi}(q_e, \ell, q_F, q_G) = \max_A\{\mathsf{Adv}^{\mathsf{RoR}}_{A,\Pi}\}$, where the maximum is taken over all A making at most q_e queries (totalling at most ℓ plaintext blocks) to their first oracle, q_F queries to F, and q_G queries to G.

There is no fully satisfactory way to formally define the effective key length of an encryption scheme, and in fact it is probably best to avoid the term. Nevertheless, for the purposes of this document we informally say that an encryption scheme has *effective key length* κ if the best attack that "breaks" the scheme with overwhelming probability when $\ell \ll 2^n$ requires $q_F + q_G \approx 2^\kappa/\mathsf{poly}(n)$.

3.3 Message-Recovery Attacks

Indistinguishability is a rather strong notion, and it is interesting to also consider a weaker property, namely, the infeasibility of recovering the *entire* plaintext from the ciphertext. In general, a definition of security in this setting would consider an experiment in which a message M is sampled from some (known) distribution $\mathcal{M}$, and the attacker is then given an encryption of M; the attacker succeeds if it correctly determines M. Note that, to make the definition meaningful, the support of $\mathcal{M}$ should be small enough so that an attacker can (with overwhelming probability) determine M following an exhaustive key search.

In the context of the PKMS2 scheme we would like to exactly capture the intuition that each segment of the encryption hides the corresponding plaintext segment, even if the attacker has successfully found the key for some other segments. For that reason, we want to consider a distribution $\mathcal{M}$ in which all segments of M are chosen independently. (Our definition is thus somewhat specific to the PKMS2 scheme.)

Somewhat arbitrarily, we look at distributions $\mathcal{M}$ in which each segment M^i is chosen uniformly and independently from a (known) set $S^i \subset \{0,1\}^{2n}$ with $|S^i| = 2^n$. Let Enc denote the PKMS2 scheme based on F, G and using some number s of segments. Fix an attacker A and sets $S^1, \ldots, S^s$. Define

$$\mathsf{Adv}^{\mathsf{MR}}_A = \Pr_{k,F,G}\left[M^i \leftarrow S^i; C \leftarrow \mathsf{Enc}_k^{F,G}(M^1 \| \cdots \| M^s) : A^{F(\cdot,\cdot),G(\cdot,\cdot)}(C) = M^1 \| \cdots \| M^s\right],$$

where we allow queries to both F and F^{-1} (and analogously for G). We define $\mathsf{Adv}^{\mathsf{MR}}(q_F, q_G) = \max_A\{\mathsf{Adv}^{\mathsf{MR}}_A\}$, where the maximum is taken over all A making at most q_F queries to F and q_G queries to G.

3.4 The H-Coefficient Technique

We use the H-coefficient technique (see [1]), which we describe briefly. Fix an attacker A that makes queries to various oracles; without loss of generality we may assume A is deterministic. As described in the previous section, A may be interacting with oracles in two different experiments: one representing a "real world" and one representing an "ideal world." The execution of A in either world defines a *transcript* that contains A's oracle queries and corresponding answers; the output of A is a deterministic function of the transcript. If we let X denote the probability distribution on transcripts defined by the real world, and let Y denote the corresponding probability distribution defined by the ideal world, then we can bound A's distinguishing advantage by the statistical difference between X and Y, denoted $\Delta(X, Y)$.

A proof using the H-coefficient technique partitions the set of all possible transcripts $\mathcal{T}$ into a subset Bad of "bad" transcripts and a subset $\mathcal{T} \setminus \mathsf{Bad}$ of "good" transcripts. We then upper-bound the probability of a bad transcript in the ideal world (i.e., show that $\Pr[Y \in \mathsf{Bad}] \leq \epsilon$), and lower-bound the ratio of the probabilities of any good transcript in the real and ideal worlds (i.e., show that for any $\tau \notin \mathsf{Bad}$ we have $\Pr[X = \tau]/\Pr[Y = \tau] \geq 1 - \epsilon'$). Then $\Delta(X, Y) \leq \epsilon + \epsilon'$.

Letting Ω_X, Ω_Y denote the underlying probability spaces in the real and ideal worlds, respectively, one can show that for any transcript τ it holds that

$$\frac{\Pr[X = \tau]}{\Pr[Y = \tau]} = \frac{\Pr[\Omega_X \text{ is consistent with } \tau]}{\Pr[\Omega_Y \text{ is consistent with } \tau]}.$$

This means that the ratio can be bounded while ignoring the specific behavior of A.

4 Fallback Security of PKMS2

In this section we prove fallback security of PKMS2. This is captured by the following theorem, which bounds security of PKMS2 without any assumptions regarding the security of F:

Theorem 1. *Let $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ denote the PKMS2 scheme using s segments in each layer, and based on keyed permutations F, G with G (only) modeled as an ideal cipher. If $\ell < 2^n$ then*

$$\mathsf{Adv}^{\mathsf{RoR}}_{\Pi}(1, \ell, q_G) \leq \frac{\binom{s}{2}}{2^{\kappa_2}} + \frac{2 \cdot s \cdot \binom{\lceil \ell/s \rceil}{2}}{2^n} + \frac{2 \cdot \binom{s}{3} \cdot \lceil \ell/s \rceil^3}{2^{2n}} + \frac{2 \cdot q_G}{2^{\kappa_2}}.$$

We assume $q_e = 1$ for simplicity, and because this is the use-case envisioned for PKMS2. The proof can be easily extended to $q_e > 1$.

Proof. Since A can simulate the first layer of encryption on its own, it suffices to prove security for a single layer of CBC encryption based on an ideal cipher G (which will correspond to the second layer of the PKMS2 scheme).

A makes a single query $M = M^1 \| \cdots \| M^s$ to its left oracle, where the $\{M^i\}$ denote the different segments and we let $|M^i| = \ell_i$ and $M^i = m_1^i \| \cdots \| m_{\ell_i}^i$ with $m_j^i \in \{0,1\}^n$. (Note that A knows how segmentation is done, since segmentation depends only on s and the length of M.) In response to this query, A receives a ciphertext $C = C^1 \| \cdots \| C^s$ from its oracle, where $C^i = c_0^i \| \cdots \| c_{\ell_i}^i$ with $c_j^i \in \{0,1\}^n$. In the real world, this oracle corresponds to the encryption oracle for a single layer of segmented CBC-mode encryption; in the ideal world, this is the oracle that simply returns uniform output of the correct length.

The transcript of A's execution includes C and all G-queries made by A. Once A has finished its execution, we augment the transcript with keys $k^1, \ldots, k^s$; in the real world these are the actual keys used, while in the ideal world they are chosen uniformly and independently. We let X, Y denote the distribution of these (augmented) transcripts in the real and ideal worlds, respectively.

By way of notation, we define $x_j^i = m_j^i \oplus c_{j-1}^i$ for $i \in \{1, \ldots, s\}$ and $j \geq 1$. Let Bad be the set transcripts for which one of the following holds:

1. For some $i \neq j$ it holds that $k^i = k^j$.
2. For some segment i and positions $1 \leq j_1 < j_2 \leq \ell_i$, it holds that $c_{j_1}^i = c_{j_2}^i$, or $x_{j_1}^i = x_{j_2}^i$.
3. For some distinct $i_1, i_2, i_3 \in \{1, \ldots, s\}$ and arbitrary $j_1, j_2, j_3 \geq 1$, it holds that $c_{j_1}^{i_1} = c_{j_2}^{i_2} = c_{j_3}^{i_3}$ or $x_{j_1}^{i_1} = x_{j_2}^{i_2} = x_{j_3}^{i_3}$.
4. For any i and $j \geq 1$, there is a G-query $(k^i, x_j^i, \star)$ or $(k^i, \star, c_j^i)$.

Since $\ell_i \leq \lceil \ell / s \rceil$, is not difficult to see that

$$\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}] \quad \leq \quad \frac{\binom{s}{2}}{2^{\kappa_2}} + \frac{2 \cdot s \cdot \binom{\lceil \ell/s \rceil}{2}}{2^n} + \frac{2 \cdot \binom{s}{3} \cdot \lceil \ell/s \rceil^3}{2^{2n}} + \frac{2 \cdot q_G}{2^{\kappa_2}}\,.$$

The final term is due to the fact that—assuming the third bad event does not occur—for any G-query $G^{-1}(k, c)$ made by A, the value c appears in at most two segments (and analogously for a query $G(k, x)$).

Fixing some $\tau \notin \mathsf{Bad}$, we prove that $\Pr[X = \tau] / \Pr[Y = \tau] \geq 1$. In the real world the probability space of interest involves the choice of uniform and independent keys, uniform choice of the s initialization vectors used, and uniform choice of the ideal cipher G. Let G_k denote the reduced function $G(k, \cdot)$. For $k \in \{0,1\}^{\kappa_2}$, we say that G_k *is X-consistent with* τ if (1) for each G-query (k, x, y) in τ, it holds that $G_k(x) = y$, and (2) if $k = k^i$ for some i, then for all j it holds that $G_k(x_j^i) = c_j^i$. In the ideal world the probability space of interest involves the choice of uniform and independent keys, with uniform choice of the ideal cipher G, and uniform choice of all the $\{c_j^i\}$. We say that G_k *is Y-consistent with* τ if (1) for each G-query (k, x, y) in τ, it holds that $G_k(x) = y$, and (2) if $k = k^i$ for some i, then for all $j \geq 1$ it holds that c_j^i is equal to the corresponding value in τ. Clearly, we have

$$\Pr[X = \tau] = 2^{-s\kappa_2} \cdot 2^{-sn} \cdot \prod_{k \in \{0,1\}^{\kappa_2}} \Pr[G_k \text{ is } X\text{-consistent with } \tau]$$

(where the probability is over choice of G), and analogously for $\Pr[Y = \tau]$ (where the probability is over choice of G and the random ciphertext blocks). But it is easy to verify that for any k,

$$\Pr[G_k \text{ is } X\text{-consistent with } \tau] \geq \Pr[G_k \text{ is } Y\text{-consistent with } \tau]$$

(using the fact that G_k is a uniform permutation). This completes the proof. □

The effect of segmentation. For reasonable settings of the parameters, the security bound given by the theorem is dominated by $\frac{\ell^2}{s \cdot 2^n} + \frac{2 \cdot q_G}{2^{\kappa_2}}$. The first term is independent of the attacker's work, and represents the deviation of CBC-mode encryption (when done in s segments, each of length roughly ℓ/s) from random due to internal collisions. The second term roughly corresponds to what would be obtained by a brute-force search for the key used in one of the segments. (Note that finding the key for a single segment is enough to violate security in this context.) The result shows that segmentation does indeed provide a modest improvement in the overall concrete security bound in regimes where $\frac{\ell^2}{2^n} \approx \frac{q_G}{2^{\kappa_2}}$.

5 Indistinguishability of PKMS2

Here and in the following we treat both F and G as ideal ciphers, and are interested in the concrete security of PKMS2.

Note that the best possible security one can hope to achieve for *any* scheme built from block ciphers F, G, for any reasonable notion of security, is $n + \max\{\kappa_1, \kappa_2\} + 1$. This is because an attacker can always break a scheme by querying both F and G on all possible inputs.

5.1 Attacks on the Indistinguishability of PKMS2

For completeness, we mention two different attacks that provide upper bounds on the effective key length of PKMS2.

The first attack is a trivial brute-force search for the keys used to encrypt a particular segment. Specifically, an attacker can focus on, say, the first segment of the ciphertext, and perform a brute-force search for k_1^1 and k_2^1 that involves simply attempting to decrypt the initial segment of the ciphertext using all possible choices of those keys. This attack requires 2^{κ_2} evaluations of G and $2^{\kappa_1+\kappa_2}$ evaluations of F, or in other words requires about $2^{\kappa_1+\kappa_2}$ time. More generally, if the attacker tries q_G possibilities for k_2^1 and then, for each of those, tries q_F/q_G possibilities for k_1^1 then the attacker succeeds with probability roughly $\left(\frac{q_G}{2^{\kappa_2}}\right) \cdot \left(\frac{q_F}{q_G \cdot 2^{\kappa_1}}\right) = \frac{q_F}{2^{\kappa_1+\kappa_2}}$.

The second attack is a variant of the classical meet-in-the-middle attack. Applied to PKMS2, the attack involves inverting, say, the first segment of the second layer using all keys $k_2^1 \in \{0,1\}^{\kappa_2}$ and computing the first segment of the first layer (in the forward direction) for all possible keys $k_1^1 \in \{0,1\}^{\kappa_1}$ *as well as* all possible values for the random initialization vector used in the first layer; the attacker then looks for a "match" between the two intermediate results obtained. This attack succeeds with high probability using $\max\{2^{n+\kappa_1}, 2^{\kappa_2}\}$ work. In the general case, this attack succeeds with probability roughly $\left(\frac{q_G}{2^{\kappa_2}}\right) \cdot \left(\frac{q_F}{2^{\kappa_1+n}}\right) = \frac{q_G \cdot q_F}{2^{\kappa_1+\kappa_2+n}}$.

The above means that the best we can hope for is an effective key length of

$$\min\{\kappa_1 + \kappa_2,\ \max\{\kappa_1 + n, \kappa_2\}\}.$$

Put differently, if $\kappa_2 \geq \kappa_1 + n$ then the effective key length is at most κ_2, if $\kappa_2 < \kappa_1 + n$ (which we view as being more likely in practice) then the effective key length is at most $\kappa_1 + \min\{n, \kappa_2\}$. In Section 5.2 we show that this bound is essentially tight.

5.2 The Effective Key Length of PKMS2

Theorem 2. *Let* Π *denote the PKMS2 scheme using* s *segments in each layer, and based on ideal ciphers* F, G*. Then*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{RoR}}_{\Pi}(1,\ell,q_F,q_G) \;\leq\; & \frac{s^2}{2^{\kappa_1}} + \frac{s^2}{s^{\kappa_2}} + \frac{s\cdot(\lceil \ell/s\rceil+1)^2}{2^n} + \frac{2\cdot\binom{s}{3}\cdot(\lceil \ell/s\rceil+1)^3}{2^{2n}} \\ & + \min\left\{\frac{s\cdot(4n+2\log s)\cdot q_F}{2^{\kappa_1+\kappa_2}} + \frac{11s\cdot\lceil \ell/s\rceil\cdot q_F}{2^{\kappa_1+n}},\; \frac{2\cdot q_G}{2^{\kappa_2}}\right\}.\end{aligned}$$

We assume $q_e = 1$ for simplicity, and because this is the use-case envisioned for PKMS2. The proof can be easily extended to handle $q_e > 1$. We also note that it is possible to modify our proof appropriately for the case $F = G$.

The first four terms above are independent of the attacker's work, and represent the deviation of (segmented) CBC-mode encryption from random due to internal collisions. The remaining term essentially reflects the two attacks outlined in Section 5.1. In fact, the theorem above shows that—up to low-order terms—the bound from Section 5.1 is tight.[3]

Theorem 2 follows from Theorems 3 and 7 that we prove next.

Theorem 3. *Let* Π *denote the PKMS2 scheme using* s *segments in each layer, and based on ideal ciphers* F, G*. Then*

$$\mathsf{Adv}^{\mathsf{RoR}}_{\Pi}(1,\ell,q_F,q_G) \leq 2^{-n} + \frac{s^2}{2^{\kappa_1}} + \frac{s\cdot\lceil \ell/s\rceil^2}{2^n} + \frac{s\cdot(4n+2\log s)\cdot q_F}{2^{\kappa_1+\kappa_2}} + \frac{11s\cdot\lceil \ell/s\rceil\cdot q_F}{2^{\kappa_1+n}}.$$

Proof. A makes a single query $M = M^1\|\cdots\|M^s$ to its left oracle (its "encryption oracle"), where $|M^i| = m^i_1\|\cdots\|m^i_{\ell_i}$, and receives a response $C = C^1\|\cdots\|C^s$ where $C^i = c^i_{-1}\|c^i_0\|c^i_1\|\cdots\|c^i_{\ell_i}$. In the ideal world the $\{c^i_j\}$ are all uniform and independent n-bit strings, whereas in the real world they are output by the encryption scheme.

The transcript of A's execution includes C. In addition, the attacker may make F-queries and G-queries (whether to the functions or their inverses); these are stored in the transcript as $\{(F,k,x,y)\}$ and $\{(G,k,x,y)\}$, respectively, where, e.g., $F(k,x) = y$. After A's execution, we augment the transcript with all G-queries of the form $\{(G,k,J^i_j(k) \stackrel{\text{def}}{=} G^{-1}_k(c^i_j), c^i_j)\}$ that have not already been made, for $i \in \{1,\ldots,s\}$, $j \geq 0$, and all $k \in \{0,1\}^{\kappa_2}$. These, in turn, define for each i,j,k the values $I^i_j(k) \stackrel{\text{def}}{=} c^i_{j-1} \oplus J^i_j(k)$ and (for $j \geq 1$) $x^i_j(k) \stackrel{\text{def}}{=} I^i_{j-1}(k) \oplus m^i_j$. Finally, we augment the transcript by adding keys $k^1_1,\ldots,k^s_1,k^1_2,\ldots,k^s_2$; in the real world these are the actual keys used by the encryption scheme, whereas in the ideal world these keys are chosen uniformly and independently of the rest of the experiment. We let $\hat{J}^i_j, \hat{I}^i_j, \hat{x}^i_j$ denote the values determined by the second-layer keys added to the transcript; e.g., $\hat{J}^i_j = J^i_j(k^i_2)$, etc. It is worth remarking that each of these values is determined by a single key out of the vector of keys added to the transcript.

[3] In fact, the attacker's success probability in Theorem 2 is not quite tight, since it should actually have a term of the form $O\left(\frac{q_F}{2^{\kappa_1+n}}\cdot\frac{q_G}{2^{\kappa_2}}\right)$ rather than of the form $O\left(\min\{\frac{q_F}{2^{\kappa_1+n}},\frac{q_G}{2^{\kappa_2}}\}\right)$.

We let X, Y denote the distribution of these (augmented) transcripts in the real and ideal worlds, respectively.

We now define a set of bad transcripts Bad. This set contains transcripts where one of the following holds:

1. For some $i_1 \neq i_2$ it holds that $k_1^{i_1} = k_1^{i_2}$.
2. For some segment i and some distinct j_1, j_2, either $\hat{x}_{j_1}^i = \hat{x}_{j_2}^i$ or $\hat{I}_{j_1}^i = \hat{I}_{j_2}^i$.
3. For some i, j, the transcript contains an F-query of the form $(F, k_1^i, \hat{x}_j^i, \star)$ or $(F, k_1^i, \star, \hat{I}_j^i)$.

We refer to the set of transcripts satisfying the ith condition as Bad_i; thus $\mathsf{Bad} = \mathsf{Bad}_1 \cup \mathsf{Bad}_2 \cup \mathsf{Bad}_3$. It is immediate that $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_1] < \frac{s^2}{2^{\kappa_1}}$. The following lemmas upper bound the probability that an ideal-world transcript τ lies in Bad_2 or Bad_3.

Lemma 4. $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_2] \leq \frac{2 \cdot s \cdot \binom{\lceil \ell/s \rceil}{2}}{2^n}$.

Proof. The $\{c_j^i\}$ are uniform and independent, and therefore so are the $\{\hat{J}_j^i\}$, $\{\hat{I}_j^i\}$, and $\{\hat{x}_j^i\}$. Thus, for any i and distinct j_1, j_2, the probability with which $\hat{x}_{j_1}^i = \hat{x}_{j_2}^i$ or $\hat{I}_{j_1}^i = \hat{I}_{j_2}^i$ is at most $2/2^n$. A union bound gives the claimed result. □

Lemma 5. $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_3] \leq 2^{-n} + \frac{s \cdot (4n + 2\log s) \cdot q_F}{2^{\kappa_1 + \kappa_2}} + \frac{11 s \cdot \lceil \ell/s \rceil \cdot q_F}{2^{\kappa_1 + n}}$.

Proof. We first upper bound the probability that τ contains an F-query of the form $(F, k_1^i, \hat{x}_j^i, \star)$ for some fixed i; we refer to the set of transcripts for which this occurs as Bad_3'. Consider a transcript τ' *before* it is augmented with the keys to give a transcript τ; we call such a τ' a *partial transcript*. We view the partial transcript τ' as inducing a bipartite graph H with vertex sets $\{0,1\}^{\kappa_1}$ and $\{0,1\}^{\kappa_2}$, where there is an edge between vertices $k_1 \in \{0,1\}^{\kappa_1}$ and $k_2 \in \{0,1\}^{\kappa_2}$ if and only if for some j the partial transcript τ' contains an F-query $(F, k_1, \hat{x}_j^i, \star)$ (where $\hat{x}_j^i$ is induced by k_2).

The main observation is that when τ' is augmented with keys $k_1^1, \ldots, k_1^s, k_2^1, \ldots, k_2^s$ to give a (full) transcript τ, then $\tau \in \mathsf{Bad}_3'$ iff there is an edge between k_1^i and k_2^i in H. Thus, we have $\Pr[\tau \in \mathsf{Bad}_3'] = |H|/2^{\kappa_1 + \kappa_2}$, where $|H|$ denotes the number of edges in H. We can therefore obtain a bound on $\Pr[\tau \in \mathsf{Bad}_3']$ by bounding the number of edges in H. We compute such a bound by arguing that with high probability, each F-query does not induce "too many" edges in H.

For $k \in \{0,1\}^{\kappa_2}$, define $X^{(k)} = \{x_j^i(k)\}_{j=1}^{\ell_i}$. For any fixed $x \in \{0,1\}^n$ and $k \in \{0,1\}^{\kappa_2}$, the probability that $x \in X^{(k)}$ is at most $\ell_i/2^n$. Thus, the probability that any fixed x lies in B or more such sets (where B is a parameter we fix later) is at most $\binom{2^{\kappa_2}}{B} \cdot \left(\frac{\ell_i}{2^n}\right)^B$, and the probability that there exists some $x \in \{0,1\}^n$ contained in B or more sets is at most $2^n \cdot \binom{2^{\kappa_2}}{B} \cdot \left(\frac{\ell_i}{2^n}\right)^B$. Let Heavy denote this event. Setting $B = 2e2^{\kappa_2}\ell_i/2^n + 2n + \log s + 1$, the probability of Heavy is at most

$$2^n \cdot \binom{2^{\kappa_2}}{B} \cdot \left(\frac{\ell_i}{2^n}\right)^B \leq 2^n \cdot \left(\frac{e \cdot 2^{\kappa_2}}{B}\right)^B \cdot \left(\frac{\ell_i}{2^n}\right)^B = 2^n \cdot \left(\frac{e \cdot 2^{\kappa_2} \cdot \ell_i}{B \cdot 2^n}\right)^B \leq 2^n \cdot 2^{-B} \leq \frac{2^{-n-1}}{s}.$$

Assuming Heavy does not occur, each F-query by A induces at most $B-1$ edges in the graph H and so H has at most $(B-1)\cdot q_F$ edges in total. Overall, then, we have

$$\begin{aligned}\Pr[\tau \in \mathsf{Bad}_2'] &\leq \Pr[\mathsf{Heavy}] + \Pr\left[\tau \in \mathsf{Bad}_2' \mid \overline{\mathsf{Heavy}}\right] \\ &\leq 2^{-n-1-\log s} + (B-1)\cdot q_F/2^{\kappa_1+\kappa_2} \\ &= 2^{-n-1-\log s} + 2e\ell_i \cdot q_F/2^{\kappa_1+n} + (2n+\log s)\cdot q_F/2^{\kappa_1+\kappa_2}.\end{aligned}$$

A symmetric argument gives the same bound for the probability with which Bad_2 occurs because τ contains an F-query of the form $(F, k_1^i, \star, \hat{I}_j^i)$. Taking a union bound over all s segments yields the lemma. □

Taken together, the above lemmas imply that

$$\Pr[Y \in \mathsf{Bad}] \leq 2^{-n} + \frac{s^2}{2^{\kappa_1}} + \frac{s\cdot\lceil \ell/s\rceil^2}{2^n} + \frac{s\cdot(4n+2\log s)\cdot q_F}{2^{\kappa_1+\kappa_2}} + \frac{11s\cdot\lceil \ell/s\rceil\cdot q_F}{2^{\kappa_1+n}}.$$

To complete the proof, we bound the ratio $\Pr[X=\tau]/\Pr[Y=\tau]$ for $\tau \notin \mathsf{Bad}$.

Lemma 6. *For any $\tau \notin \mathsf{Bad}$ we have $\Pr[X=\tau]/\Pr[Y=\tau] \geq 1$.*

Proof. Fix any $\tau \notin \mathsf{Bad}$. The ratio of interest is equal to

$$\frac{\Pr[\Omega_X \text{ is consistent with } \tau]}{\Pr[\Omega_Y \text{ is consistent with } \tau]}.$$

Probability space Ω_Y is defined by choosing for each $k \in \{0,1\}^{\kappa_1}$ an independent, uniform permutation $F(k,\cdot) : \{0,1\}^n \to \{0,1\}^n$; for each $k \in \{0,1\}^{\kappa_2}$ an independent, uniform permutation $G(k,\cdot) : \{0,1\}^n \to \{0,1\}^n$; independent, uniform values $c_j^i \in \{0,1\}^n$, and independent, uniform keys $k_1^1,\ldots,k_1^s \in \{0,1\}^{\kappa_1}$ and $k_2^1,\ldots,k_2^s \in \{0,1\}^{\kappa_2}$; thus,

$$|\Omega_Y| = (2^n!)^{2^{\kappa_1+2^{\kappa_2}}} \cdot (2^n)^{\ell+2s} \cdot 2^{s\cdot(\kappa_1+\kappa_2)}.$$

Probability space Ω_X is defined by choosing F and G in the same way, and then choosing independent, uniform keys $k_1^1,\ldots,k_1^s \in \{0,1\}^{\kappa_1}$ and $k_2^1,\ldots,k_2^s \in \{0,1\}^{\kappa_2}$, and $2s$ independent, uniform initialization vectors that we denote by $\{c_{-1}^i, I_0^i\}$; thus,

$$|\Omega_X| = (2^n!)^{2^{\kappa_1+2^{\kappa_2}}} \cdot 2^{s\cdot(\kappa_1+\kappa_2)} \cdot (2^n)^{2s}.$$

Let $\mathsf{Con}_\tau^Y \subseteq \Omega_Y$ be the set of $\omega \in \Omega_Y$ consistent with τ, and define Con_τ^X analogously. Then

$$\frac{\Pr[\Omega_X \text{ is consistent with } \tau]}{\Pr[\Omega_Y \text{ is consistent with } \tau]} = \frac{|\mathsf{Con}_\tau^X|/|\Omega_X|}{|\mathsf{Con}_\tau^Y|/|\Omega_Y|}.$$

Now, a particular $\omega = (F, G, \{c_j^i\}, k_1^1,\ldots,k_1^s, k_2^1,\ldots,k_2^s) \in \Omega_Y$ is consistent with τ if and only if:

- The values $\{c_j^i\}$, $k_1^1,\ldots,k_1^s$, and $k_2^1,\ldots,k_2^s$ equal the corresponding values in τ.
- G is consistent with all G-queries in τ.
- F is consistent with all F-queries in τ.

Let $\mathsf{Con}_\tau(G)$ denote the set of ciphers $G : \{0,1\}^{\kappa_2} \times \{0,1\}^n \to \{0,1\}^n$ consistent with the G-queries in τ. For any $k \in \{0,1\}^{\kappa_1}$, let $\mathsf{Con}_\tau(F_k)$ denote the set of permutations $F_k : \{0,1\}^n \to \{0,1\}^n$ consistent with the F-queries in τ involving key k. Let q_F^i denote the number of F-queries in τ that involve the key k_1^i included in τ. Then

$$\begin{aligned}
|\mathsf{Con}_\tau^Y| &= |\mathsf{Con}_\tau(G)| \cdot \prod_{k \in \{0,1\}^{\kappa_1}} |\mathsf{Con}_\tau(F_k)| \\
&= |\mathsf{Con}_\tau(G)| \cdot \prod_i |\mathsf{Con}_\tau(F_{k_1^i})| \cdot \prod_{k \in \{0,1\}^{\kappa_1} \setminus \{k_1^1, \ldots, k_1^s\}} |\mathsf{Con}_\tau(F_k)| \\
&= |\mathsf{Con}_\tau(G)| \cdot \prod_i (2^n - q_F^i)! \cdot \prod_{k \in \{0,1\}^{\kappa_1} \setminus \{k_1^1, \ldots, k_1^s\}} |\mathsf{Con}_\tau(F_k)|.
\end{aligned}$$

In contrast, $\omega = (F, G, k_1^1, \ldots, k_1^s, k_2^1, \ldots, k_2^s, \{c_{-1}^i, I_0^i\}) \in \Omega_X$ is consistent with τ if and only if:

- The values $\{c_{-1}^i\}$, $k_1^1, \ldots, k_1^s$, and $k_2^1, \ldots, k_2^s$ equal the corresponding values in τ, and $I_0^i = \hat{I}_0^i$ for all i.
- G is consistent with all G-queries in τ.
- For all $k \notin \{k_1^1, \ldots, k_1^s\}$, the permutation $F(k, \cdot)$ is consistent with all F-queries in τ involving key k.
- For all i, the permutation $F(k_1^i, \cdot)$ is consistent with all F-queries in τ involving key k_1^i, and in addition $F(k_1^i, \hat{x}_j^i) = \hat{I}_j^i$ for $1 \le j \le \ell_i$.

For any $i \in \{1, \ldots, s\}$, let $\mathsf{Con}'_\tau(F_{k_1^i})$ denote the set of permutations $F_{k_1^i} : \{0,1\}^n \to \{0,1\}^n$ consistent with the F-queries in τ involving key k_1^i and also having $F(k_1^i, \hat{x}_j^i) = \hat{I}_j^i$ for $1 \le j \le \ell_i$. Because $\tau \notin \mathsf{Bad}$, the constraints $\left\{F(k_1^i, \hat{x}_j^i) = \hat{I}_j^i\right\}$ do not conflict with each other (since the $\{x_j^i\}$ and $\{I_j^i\}$ are each distinct), nor do they conflict with any F-queries in τ (since there is no F-query in τ with input equal to any of the $\{\hat{x}_j^i\}$ or output equal to any of the $\{\hat{I}_j^i\}$). Thus, $\mathsf{Con}'_\tau(F_{k_1^i}) = (2^n - q_F^i - \ell_i)!$ and so

$$\begin{aligned}
|\mathsf{Con}_\tau^X| &= |\mathsf{Con}_\tau(G)| \cdot \prod_i |\mathsf{Con}'_\tau(F_{k_1^i})| \cdot \prod_{k \in \{0,1\}^{\kappa_1} \setminus \{k_1^1, \ldots, k_1^s\}} |\mathsf{Con}_\tau(F_k)| \\
&= |\mathsf{Con}_\tau(G)| \cdot \prod_i (2^n - q_F^i - \ell_i)! \cdot \prod_{k \in \{0,1\}^{\kappa_1} \setminus \{k_1^1, \ldots, k_1^s\}} |\mathsf{Con}_\tau(F_k)|.
\end{aligned}$$

Putting everything together, we have

$$\begin{aligned}
\frac{\Pr[X = \tau]}{\Pr[Y = \tau]} = \frac{|\mathsf{Con}_\tau^X|/|\Omega_X|}{|\mathsf{Con}_\tau^Y|/|\Omega_Y|} &= \frac{|\mathsf{Con}_\tau^X|}{|\mathsf{Con}_\tau^Y|} \cdot \frac{|\Omega_Y|}{|\Omega_X|} \\
&= \frac{\prod_i (2^n - q_F^i - \ell_i)!}{\prod_i (2^n - q_F^i)!} \cdot (2^n)^\ell \\
&= \frac{(2^n)^\ell}{\prod_i \prod_{j=0}^{\ell_i - 1} (2^n - q_F^i - j)} \ \ge\ 1,
\end{aligned}$$

completing the proof. □

This completes the proof of Theorem 3. □

Theorem 7. *Let* Π *denote the PKMS2 scheme using* s *segments in each layer, and based on ideal ciphers* F, G*. Then*

$$\mathsf{Adv}^{\mathsf{RoR}}_{\Pi}(1, \ell, q_F, q_G) \leq \frac{s^2}{s^{\kappa_2}} + \frac{2 \cdot s \cdot \binom{\lceil \ell/s \rceil + 1}{2}}{2^n} + \frac{2 \cdot \binom{s}{3} \cdot (\lceil \ell/s \rceil + 1)^3}{2^{2n}} + \frac{2 \cdot q_G}{2^{\kappa_2}}.$$

Proof. The proof here is simpler than the proof of Theorem 3. A makes a single query $M = M^1 \| \cdots \| M^s$ to its left oracle (its "encryption oracle"), where $|M^i| = m^i_1 \| \cdots \| m^i_{\ell_i}$, and receives a response $C = C^1 \| \cdots \| C^s$ where $C^i = c^i_{-1} \| c^i_0 \| c^i_1 \| \cdots \| c^i_{\ell_i}$. In the ideal world the $\{c^i_j\}$ are all uniform and independent n-bit strings, whereas in the real world they are output by the encryption scheme. In addition, the attacker makes F-queries and G-queries (whether to the functions or their inverses); these are stored in the transcript as $\{(F, k, x, y)\}$ and $\{(G, k, x, y)\}$, respectively, where, e.g., $F(k, x) = y$. After A's execution, we augment the transcript with keys $k^1_1, \ldots, k^s_1, k^1_2, \ldots, k^s_2$ and initialization vectors $\hat{I}^1_0, \ldots, I^s_0$; in the real world these are the actual values used by Enc, whereas in the ideal world these values are chosen uniformly and independently of the rest of the experiment.

Recursively define $\hat{x}^i_j \stackrel{\text{def}}{=} m^i_j \oplus \hat{I}^i_{j-1}$ and $\hat{I}^i_j \stackrel{\text{def}}{=} F_{k^i_1}(\hat{x}^i_j)$ for $i = 1, \ldots, s$ and $j \geq 1$. We also augment the transcript with all F-queries $\{(F, k^i_1, \hat{x}^i_j, \hat{I}^i_j)\}$ that were not already made by A. These values, in turn, define values $\hat{J}^i_j \stackrel{\text{def}}{=} \hat{I}^i_j \oplus c^i_{j-1}$ for $i = 1, \ldots, s$ and $j \geq 0$. We let X, Y denote the distribution of these (augmented) transcripts in the real and ideal worlds, respectively.

We now define a set of bad transcripts Bad. This set contains transcripts where one of the following holds:

1. For some distinct i_1, i_2 it holds that $k^{i_1}_2 = k^{i_2}_2$.
2. For some segment i and distinct $j_1, j_2 \geq 0$, either $c^i_{j_1} = c^i_{j_2}$ or $\hat{J}^i_{j_1} = \hat{J}^i_{j_2}$.
3. For some distinct $i_1, i_2, i_3 \in \{1, \ldots, s\}$ and arbitrary $j_1, j_2, j_3 \geq 1$, it holds that $c^{i_1}_{j_1} = c^{i_2}_{j_2} = c^{i_3}_{j_3}$ or $\hat{J}^{i_1}_{j_1} = \hat{J}^{i_2}_{j_2} = \hat{J}^{i_3}_{j_3}$.
4. For some i, j, the transcript contains a G-query of the form $(G, k^i_2, \hat{J}^i_j, \star)$ or $(G, k^i_2, \star, c^i_j)$.

We refer to the set of transcripts satisfying the ith condition as Bad_i; thus $\mathsf{Bad} = \mathsf{Bad}_1 \cup \mathsf{Bad}_2 \cup \mathsf{Bad}_3 \cup \mathsf{Bad}_4$. It is immediate that $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_1] < \frac{s^2}{2^{\kappa_2}}$. The following lemmas upper bound the probability that an ideal-world transcript τ lies in $\mathsf{Bad}_2, \mathsf{Bad}_3$, or Bad_4.

Lemma 8. $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_2] \leq \frac{2 \cdot s \cdot \binom{\lceil \ell/s \rceil + 1}{2}}{2^n}$ *and* $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_3] \leq \frac{2 \cdot \binom{s}{3} \cdot (\lceil \ell/s \rceil + 1)^3}{2^{2n}}$.

Proof. The $\{c^i_j\}$ are uniform and independent, and therefore so are the $\{\hat{J}^i_j\}$. Thus, for any i and distinct j_1, j_2, the probability with which $c^i_{j_1} = c^i_{j_2}$ or $\hat{J}^i_{j_1} = \hat{J}^i_{j_2}$ is at most $2/2^n$. A union bound gives the first result. A proof of the second result is analogous. □

Lemma 9. $\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}_4 \cap \overline{\mathsf{Bad}_3}] \leq \frac{2 \cdot q_G}{2^{\kappa_2}}$.

Proof. This follows from the observation that, as long as $\tau \notin \mathsf{Bad}_3$, for any G-query $G^{-1}(k, c)$ made by A the value c appears in at most two segments (and analogously for a query $G(k, J)$). □

The above discussion shows that

$$\Pr_{\tau \leftarrow Y}[\tau \in \mathsf{Bad}] \leq \frac{s^2}{s^{\kappa_2}} + \frac{2 \cdot s \cdot \binom{\lceil \ell/s \rceil + 1}{2}}{2^n} + \frac{2 \cdot \binom{s}{3} \cdot (\lceil \ell/s \rceil + 1)^3}{2^{2n}} + \frac{2 \cdot q_G}{2^{\kappa_2}}.$$

It is not hard to show, as in the proof of Lemma 6, that $\Pr[X = \tau]/\Pr[Y = \tau]$ for $\tau \notin \mathsf{Bad}$. This completes the proof. This concludes the proof of Theorem 7. □

6 Security of PKMS2 Against Message-Recovery Attacks

We are unable to give a (tight) proof of security for PKMS2 against message-recovery attacks. However, we can analyze a natural message-recovery attack against PKMS2 that we conjecture is essentially the best possible (up to lower-order corrections).

First consider message-recovery attacks against (non-segmented) cascaded CBC-mode encryption. Letting $\kappa \stackrel{\text{def}}{=} \max\{\kappa_2, \kappa_1 + \min\{n, \kappa_2\}\}$, Theorem 2 shows that (up to lower-order terms) an attacker making q queries to F and/or G can determine the underlying plaintext with probability at most $O\left(\frac{q}{2^\kappa}\right)$. This is also essentially tight, as there is a matching attack as discussed in Section 5.1.

We analyze the natural message-recovery attack against the PKMS2 scheme with s segments that simply carries out the above brute-force attack sequentially on each segment. The attack succeeds only if it manages to succeed on *every* segment. We are interested in determining the success probability for this attack when the attacker has a total budget of q^* queries.

Let Q_i be a random variable denoting the number of queries needed to succeed on the ith segment, and let $N \stackrel{\text{def}}{=} 2^\kappa$. We may model the $\{Q_i\}$ as independent random variables, each uniform in the interval $\{0, \ldots, N\}$ (allowing $Q_i = 0$ only helps the attacker); in that case, the attacker succeeds exactly if $\sum_i Q_i \leq q^*$. (Note that the attacker can tell when it has succeeded, and can then move on to the next segment.) We analyze the probability that this occurs using Hoeffding's inequality:

Lemma 10 (Hoeffding)**.** *Let $Q_1, \ldots, Q_s$ be independent random variables with $Q_i \in \{1, \ldots, N\}$ and $\mathbf{Exp}[Q_i] = \mu_i$. Then for any $t > 0$,*

$$\Pr\left[\left|\sum_i Q_i - \sum_i \mu_i\right| \geq t\right] \leq 2 \cdot \exp\left(-\frac{2t^2}{s \cdot N^2}\right).$$

Note that $\mu_i = \mathbf{Exp}[Q_i] = N/2$ and so $\sum_i \mu_i = sN/2$. Let $Q = \sum_i Q_i$ be the random variable denoting the total number of queries needed to succeed on all segments, and assume $q^* < sN/2$. We then have:

$$\begin{aligned}
\Pr[Q \leq q^*] &= \Pr\left[\frac{sN}{2} - Q \geq \frac{sN}{2} - q^*\right] \\
&\leq \Pr\left[\left|\frac{sN}{2} - Q\right| \geq \frac{sN}{2} - q^*\right] \\
&\leq 2 \cdot \exp\left(-\frac{2\left(\frac{sN}{2} - q^*\right)^2}{s \cdot N^2}\right).
\end{aligned}$$

Let $q^* = c \cdot \frac{sN}{2}$ for some $c < 1$. Then

$$\begin{aligned} \Pr[Q \leq q^*] &\leq 2 \cdot \exp\left(-\frac{2\left(\frac{sN}{2} - \frac{csN}{2}\right)^2}{s \cdot N^2}\right) \\ &= 2 \cdot \exp\left(-\frac{s \cdot (1-c)^2}{2}\right). \end{aligned}$$

References

[1] Shan Chen and John Steinberger. Tight security bounds for key-alternating ciphers. *Eurocrypt 2014*.

[2] Phil Rogaway. Evaluation of Some Blockcipher Modes of Operation. Manuscript, 2011. Available at `http://web.cs.ucdavis.edu/ rogaway/papers/modes.pdf`.